

# ۶۰۰ برنامه C++ با حل کامل آنها

(حل مسائل C++ – مرجع کامل)



مولفین: دکتر رمضان عباس نژادورزی  
مهندس نیلوفر شعیبی

## برخی از عناوین مهم

آشنایی با C++ (دستورات ورودی و خروجی)  
ساختار تصمیم و حلقه تکرار  
توابع در C++  
آرایه‌ها و رشته‌ها  
گرافیک  
ساختمان، کلاس‌ها و وراثت  
فایل‌های ورودی و خروجی

---

---

# ۶۰۰ برنامه C++ با حل آن‌ها

## (حل مسائل C++ - مرجع کامل)

---

---

تألیف:

دکتر رمضان عباس نژادورزی  
مهندس نیلوفر شعبی



فن آوری نوین

---

---

|                                                                                                 |                     |
|-------------------------------------------------------------------------------------------------|---------------------|
| عباس نژاد ورزی، رمضان، ۱۳۴۸ -                                                                   | سرشناسه             |
| ۶۰۰ برنامه C++ با حل آن‌ها (حل مسائل C++- مرجع کامل) / تألیف رمضان عباس نژاد ورزی، نیلوفر شعبی. | عنوان و نام پدیدآور |
| بابل: فناوری نوین، ۱۳۹۷.                                                                        | مشخصات نشر          |
| ۴۹۶ ص.                                                                                          | مشخصات ظاهری        |
| ۴۰۰۰۰ ریال: ۴-۱۱-۷۲۷۲-۶۰۰-۹۷۸                                                                   | شابک                |
| فیپا                                                                                            | وضعیت فهرست نویسی   |
| ششصد برنامه C++ با حل آن‌ها (حل مسائل C++- مرجع کامل).                                          | موضوع               |
| سی ++ (زبان برنامه نویسی کامپیوتر)                                                              | موضوع               |
| C++ (Computer program language)                                                                 | موضوع               |
| سی ++ (زبان برنامه نویسی کامپیوتر) -- آزمون‌ها و تمرین‌ها (عالی)                                | موضوع               |
| C++ (Computer program language) -- Exercises, questions, etc. (Higher)                          | موضوع               |
| شعبی، نیلوفر، ۱۳۷۰ -                                                                            | شناسه افزوده        |
| ۱۳۹۷ ۲۶ ع ۹۳ / ۷۶ / ۷۳ QA                                                                       | رده بندی کنگره      |
| ۰۰۵                                                                                             | رده بندی دیویی      |
| ۵۱۴۶۳۳۶                                                                                         | شماره کتابشناسی ملی |



فن آوری نوین

[www.fanavarienovin.net](http://www.fanavarienovin.net)

بابل، کد پستی ۴۷۱۶۷-۷۳۴۴۸

تلفن: ۰۱۱-۳۲۲۵۶۶۸۷

۶۰۰ برنامه C++ با حل آن‌ها (حل مسائل C++- مرجع کامل)

تألیف: رمضان عباس نژاد ورزی، نیلوفر شعبی

نوبت چاپ: چاپ اول

سال چاپ: بهار ۹۷

شمارگان: ۱۰۰۰

قیمت: ۴۰۰۰۰ تومان

نام چاپخانه و صحافی:

شابک: ۴-۱۱-۷۲۷۲-۶۰۰-۹۷۸

نشانی ناشر: بابل، چهارراه نواب، کاظم بیگی، جنب مسجد منصور کاظم بیگی، طبقه اول

طراح جلد: کانون آگهی و تبلیغات آبان (احمد فرجی)

تهران، خ اردیبهشت، نش و حید نظری، پلاک ۱۴۲ تلفکس: ۶۶۴۰۰۱۴۴-۶۶۴۰۰۲۲۰

## فهرست مطالب

|                                                      |     |
|------------------------------------------------------|-----|
| فصل اول: آشنایی با C++ و دستورات ورودی و خروجی ..... | ۵   |
| فصل دوم: ساختار تصمیم و حلقه تکرار .....             | ۳۹  |
| فصل سوم: توابع در C++ .....                          | ۱۵۱ |
| فصل چهارم: آرایه ها و رشته ها .....                  | ۲۴۳ |
| فصل پنجم: <b>گرافیک در C++</b> .....                 | ۳۷۲ |
| فصل ششم: <b>ساختمان و کلاس ها</b> .....              | ۳۹۳ |
| فصل هفتم: <b>ورودی و خروجی فایل</b> .....            | ۴۷۴ |
| منابع: .....                                         | ۴۹۳ |

## مقدمه

زبان C++ پایه زبان‌های نظیر C#، جاوا، ویزوال C++ و غیره می‌باشد. این زبان امروزه به‌عنوان یکی از زبان‌های برنامه‌سازی در رشته‌های کامپیوتر (نرم‌افزار، سخت‌افزار و IT)، رشته‌های مهندسی از قبیل عمران، برق، الکترونیک، ...، رشته‌های علوم پایه مانند ریاضی، فیزیک، علوم کامپیوتر و غیره تدریس می‌شود. لذا، یادگیری این زبان برای دانشجویان مهندسی و علوم پایه اجباری می‌باشد. به همین دلیل، این زبان به‌عنوان بااهمیت‌ترین زبان‌های برنامه‌نویسی می‌باشد. یکی از راه‌های آموزش زبان برنامه‌نویسی و الگوریتم، حل مسائل متعدد و متنوع در آن زبان می‌باشد. کتاب حاضر شامل ۷ فصل است.

فصل اول، شامل ۷۵ برنامه می‌باشد که مسائلی از قبیل تعریف متغیرها دستورات cin، cout و عملگرها را در C++ آموزش می‌دهد.

فصل دوم، شامل ۲۳۵ برنامه است که بخشی از آن در کتاب و بخش دیگر آن در پیوست الکترونیکی حل گردیده است. در این فصل مباحثه تصمیم‌گیری (دستورات if، switch و حلقه‌های تکرار (دستورات for، while، do while، continue و break) با مثال‌های متعدد شرح داده شده است.

فصل سوم، شامل ۱۰۷ مثال برنامه‌نویسی است که بخشی از آن در پیوست الکترونیکی آمده است. این فصل تعریف توابع و فراخوانی آن‌ها را با مثال‌های مختلف آموزش می‌دهد.

فصل چهارم، شامل ۱۳۵ مثال برنامه‌نویسی است. این فصل مباحثی مانند آرایه‌های یک‌بعدی، دوبعدی و رشته‌ها را در C++ آموزش می‌دهد.

فصل پنجم، شامل ۳۹ برنامه است که گرافیک در C++ را آموزش می‌دهد.

فصل ششم، شامل ۳۴ برنامه است که تعریف کلاس‌ها، وراثت، چندریختی و پیاده‌سازی مجدد عملگرها را آموزش می‌دهد.

و بالاخره فصل هفتم شامل ۱۶ برنامه است که خواندن و نوشتن در فایل‌های متنی و باینری را آموزش می‌دهد.

از تمامی اساتید و دانشجویان عزیز تقاضا داریم، هرگونه اشکال، ابهام در متن کتاب، پیشنهاد و انتقادات را به آدرس پست الکترونیک [fanavarienovin@gmail.com](mailto:fanavarienovin@gmail.com) ارسال نمایند.

در پایان امیدوارم این اثر مورد توجه جامعه انفورماتیک کشور، اساتید و دانشجویان عزیز قرار گیرد.

مؤلفین

[fanavarienovin@gmail.com](mailto:fanavarienovin@gmail.com)

| نام کتاب                                                       | لینک خرید چاپی                                                                                | لینک خرید الکترونیکی                                        | لینک فایل نمونه                                                             |
|----------------------------------------------------------------|-----------------------------------------------------------------------------------------------|-------------------------------------------------------------|-----------------------------------------------------------------------------|
| مبانی رایانه و برنامه‌نویسی به زبان ++C                        | <a href="http://daneshnegar.com/book/380238.html">http://daneshnegar.com/book/380238.html</a> | <a href="http://ktbr.ir/b/30588">http://ktbr.ir/b/30588</a> | <a href="http://ketabesabz.com/dl/52319">http://ketabesabz.com/dl/52319</a> |
| آشنایی با مبانی امنیت شبکه (امنیت اطلاعات)                     | <a href="http://daneshnegar.com/book/371137.html">http://daneshnegar.com/book/371137.html</a> | <a href="http://ktbr.ir/b/30327">http://ktbr.ir/b/30327</a> |                                                                             |
| اصول طراحی پایگاه داده‌ها                                      | <a href="http://daneshnegar.com/book/371655.html">http://daneshnegar.com/book/371655.html</a> | <a href="http://ktbr.ir/b/29943">http://ktbr.ir/b/29943</a> | <a href="http://ketabesabz.com/dl/52155">http://ketabesabz.com/dl/52155</a> |
| آموزش گام‌به‌گام برنامه‌نویسی پایتون                           | <a href="http://daneshnegar.com/book/392582.html">http://daneshnegar.com/book/392582.html</a> | <a href="http://ktbr.ir/b/29984">http://ktbr.ir/b/29984</a> | <a href="http://ketabesabz.com/dl/52181">http://ketabesabz.com/dl/52181</a> |
| حل مسائل ++C (آزمایشگاه کامپیوتر مرجع کامل)                    | <a href="http://daneshnegar.com/book/392262.html">http://daneshnegar.com/book/392262.html</a> | <a href="http://ktbr.ir/b/29982">http://ktbr.ir/b/29982</a> | <a href="http://ketabesabz.com/dl/53546">http://ketabesabz.com/dl/53546</a> |
| آموزش گام‌به‌گام LINQ با C#                                    | <a href="http://daneshnegar.com/book/369388.html">http://daneshnegar.com/book/369388.html</a> | <a href="http://ktbr.ir/b/28451">http://ktbr.ir/b/28451</a> |                                                                             |
| ساختمان داده‌ها با ++C                                         | <a href="http://daneshnegar.com/book/379094.html">http://daneshnegar.com/book/379094.html</a> | <a href="http://ktbr.ir/b/29676">http://ktbr.ir/b/29676</a> | <a href="http://ketabesabz.com/dl/52156">http://ketabesabz.com/dl/52156</a> |
| طراحی سیستم‌های شی‌گرا با زبان C#                              | <a href="http://daneshnegar.com/book/374658.html">http://daneshnegar.com/book/374658.html</a> | <a href="http://ktbr.ir/b/29621">http://ktbr.ir/b/29621</a> | <a href="http://ketabesabz.com/dl/52126">http://ketabesabz.com/dl/52126</a> |
| مدیریت استراتژیک فناوری اطلاعات                                | <a href="http://daneshnegar.com/book/374659.html">http://daneshnegar.com/book/374659.html</a> | <a href="http://ktbr.ir/b/29779">http://ktbr.ir/b/29779</a> | <a href="http://ketabesabz.com/dl/52125">http://ketabesabz.com/dl/52125</a> |
| گرافیک رایانه‌ای با زبان برنامه‌نویسی C#                       | <a href="http://daneshnegar.com/book/376021.html">http://daneshnegar.com/book/376021.html</a> | <a href="http://ktbr.ir/b/29674">http://ktbr.ir/b/29674</a> | <a href="http://ketabesabz.com/dl/51049">http://ketabesabz.com/dl/51049</a> |
| درس و کنکور پایگاه داده پیشرفته                                | <a href="http://daneshnegar.com/book/392578.html">http://daneshnegar.com/book/392578.html</a> | <a href="http://ktbr.ir/b/29644">http://ktbr.ir/b/29644</a> | <a href="http://ketabesabz.com/dl/52102">http://ketabesabz.com/dl/52102</a> |
| فیزیک الکترونیسته                                              | <a href="http://daneshnegar.com/book/379161.html">http://daneshnegar.com/book/379161.html</a> | <a href="http://ktbr.ir/b/29680">http://ktbr.ir/b/29680</a> |                                                                             |
| تجارت الکترونیکی                                               | <a href="http://daneshnegar.com/book/379188.html">http://daneshnegar.com/book/379188.html</a> | <a href="http://ktbr.ir/b/29619">http://ktbr.ir/b/29619</a> | <a href="http://ketabesabz.com/dl/51013">http://ketabesabz.com/dl/51013</a> |
| راهنمای کاربردی کاربری OPNET برای شبکه‌های شبیه‌سازی کامپیوتر  | <a href="http://daneshnegar.com/book/392583.html">http://daneshnegar.com/book/392583.html</a> | <a href="http://ktbr.ir/b/28504">http://ktbr.ir/b/28504</a> |                                                                             |
| درس و کنکور سیستم عامل پیشرفته                                 | <a href="http://daneshnegar.com/book/392580.html">http://daneshnegar.com/book/392580.html</a> | <a href="http://ktbr.ir/b/28505">http://ktbr.ir/b/28505</a> | <a href="http://ketabesabz.com/dl/52180">http://ketabesabz.com/dl/52180</a> |
| شبکه‌های کامپیوتری با رویکرد کاربردی، آزمایشگاه شبیه‌سازی شبکه | <a href="http://daneshnegar.com/book/392254.html">http://daneshnegar.com/book/392254.html</a> | <a href="http://ktbr.ir/b/28528">http://ktbr.ir/b/28528</a> |                                                                             |
| آزمایشگاه پایگاه داده با SQL Server ۲۰۱۲                       | <a href="http://daneshnegar.com/book/377301.html">http://daneshnegar.com/book/377301.html</a> | <a href="http://ktbr.ir/b/28503">http://ktbr.ir/b/28503</a> | <a href="http://ketabesabz.com/dl/53447">http://ketabesabz.com/dl/53447</a> |

۶۰۰ برنامه C++ با حل آن‌ها ۷

|                                                                             |                                                           |                                                                                               |                                                                |
|-----------------------------------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------------------------------------------|----------------------------------------------------------------|
| <a href="http://ketabesabz.com/dl/۵۳۴۴۶">http://ketabesabz.com/dl/۵۳۴۴۶</a> | <a href="http://ktbr.ir/b۲۸۴۵۰">http://ktbr.ir/b۲۸۴۵۰</a> | <a href="http://daneshnegar.com/book_۳۷۵۸۹۲.html">http://daneshnegar.com/book_۳۷۵۸۹۲.html</a> | کاربرد رایانه در مدیریت و حسابداری                             |
| <a href="http://ketabesabz.com/dl/۵۳۴۸۸">http://ketabesabz.com/dl/۵۳۴۸۸</a> | <a href="http://ktbr.ir/b۲۸۴۴۹">http://ktbr.ir/b۲۸۴۴۹</a> | <a href="http://daneshnegar.com/book_۳۶۸۹۲۹.html">http://daneshnegar.com/book_۳۶۸۹۲۹.html</a> | آموزش گام به گام برنامه نویسی بانک اطلاعاتی با ویژوال بیسیک نت |
|                                                                             | <a href="http://ktbr.ir/b۲۸۴۵۲">http://ktbr.ir/b۲۸۴۵۲</a> | <a href="http://daneshnegar.com/book_۳۸۰۲۳۸.html">http://daneshnegar.com/book_۳۸۰۲۳۸.html</a> | آموزش گام به گام برنامه نویسی به زبان C++                      |
| <a href="http://ketabesabz.com/dl/۵۱۰۴۷">http://ketabesabz.com/dl/۵۱۰۴۷</a> | <a href="http://ktbr.ir/b۲۸۴۴۸">http://ktbr.ir/b۲۸۴۴۸</a> | <a href="http://daneshnegar.com/book_۳۶۸۴۸۶.html">http://daneshnegar.com/book_۳۶۸۴۸۶.html</a> | دانلود کتاب آموزش گام به گام برنامه نویسی بانک اطلاعاتی با C#  |
|                                                                             | <a href="http://ktbr.ir/b۲۸۳۹۸">http://ktbr.ir/b۲۸۳۹۸</a> |                                                                                               | حل مسائل پاسکال                                                |
| <a href="http://ketabesabz.com/dl/۵۱۰۴۸">http://ketabesabz.com/dl/۵۱۰۴۸</a> | <a href="http://ktbr.ir/b۲۸۴۴۱">http://ktbr.ir/b۲۸۴۴۱</a> | <a href="http://daneshnegar.com/book_۳۹۲۲۶۲.html">http://daneshnegar.com/book_۳۹۲۲۶۲.html</a> | حل مسائل C++                                                   |
| <a href="http://ketabesabz.com/dl/۵۱۰۱۱">http://ketabesabz.com/dl/۵۱۰۱۱</a> | <a href="http://ktbr.ir/b۲۸۳۹۹">http://ktbr.ir/b۲۸۳۹۹</a> | <a href="http://daneshnegar.com/book_۳۷۴۶۵۷.html">http://daneshnegar.com/book_۳۷۴۶۵۷.html</a> | دانلود کتاب حل مسائل C#                                        |
|                                                                             | <a href="http://ktbr.ir/b۳۲۶۷۲">http://ktbr.ir/b۳۲۶۷۲</a> | <a href="http://daneshnegar.com/book_۳۹۲۸۱۸.html">http://daneshnegar.com/book_۳۹۲۸۱۸.html</a> | ۶۰۰ برنامه C++ با حل کامل آنها (حل مسائلی - C++ مرجع کامل)     |
|                                                                             | <a href="http://ktbr.ir/b۳۲۶۷۱">http://ktbr.ir/b۳۲۶۷۱</a> | <a href="http://daneshnegar.com/book_۳۹۲۸۱۷.html">http://daneshnegar.com/book_۳۹۲۸۱۷.html</a> | حل مسائل جاوا (حل ۶۰۰ برنامه - مرجع کامل)                      |
|                                                                             |                                                           | <a href="http://daneshnegar.com/book_۳۹۲۸۱۵.html">http://daneshnegar.com/book_۳۹۲۸۱۵.html</a> | ۶۵۰ برنامه # C با حل کامل آنها (حل مسائل #C - مرجع).           |
|                                                                             |                                                           | <a href="http://daneshnegar.com/book_۳۹۲۸۱۹.html">http://daneshnegar.com/book_۳۹۲۸۱۹.html</a> | حل مسائل پایتون (حل ۶۵۰ برنامه - مرجع کامل)                    |
|                                                                             |                                                           | <a href="http://daneshnegar.com/book_۳۹۲۸۱۴.html">http://daneshnegar.com/book_۳۹۲۸۱۴.html</a> | دانشی ثروتمند شدن                                              |
|                                                                             |                                                           | <a href="http://daneshnegar.com/book_۳۹۲۸۱۶.html">http://daneshnegar.com/book_۳۹۲۸۱۶.html</a> | برنامه سازی پیشرفته به زبان C#                                 |
|                                                                             |                                                           | <a href="http://daneshnegar.com/book_۳۹۲۸۱۳.html">http://daneshnegar.com/book_۳۹۲۸۱۳.html</a> | الگوریتم ها و محاسبات موازی                                    |
|                                                                             | <a href="http://ktbr.ir/b۲۸۳۹۷">http://ktbr.ir/b۲۸۳۹۷</a> |                                                                                               | دانلود کتاب حل مسائل C                                         |

## آشنایی با C# (دستورات ورودی و خروجی)

۱. دستوراتی که متغیرهای  $m$  و  $n$  را با نوع `int`،  $ch$  و  $ch2$  را با نوع `char`،  $d1$  و  $d2$  را با نوع `double` و ثابت `pl` را با نوع `float` با مقدار ۳,۱۴ تعریف می کنند.

۲. دستورات زیر چند بایت از حافظه را اشغال می کنند؟

```
long double x, y;
char ch;
float f1, f2;
```

این دستورات، ۲۹ بایت از حافظه را اشغال می کنند. زیرا هر یک از متغیرهای  $x$  و  $y$ ، ۱۰ بایت (نوع `long double`)،  $ch$  یک بایت (نوع `char`) و  $f1$  و  $f2$  هر یک ۴ بایت (نوع `float`) را اشغال می کنند. لذا، این دستورات مجموعاً ۲۹ (۸ + ۱ + ۲۰) بایت را اشغال می کنند.

۳. عبارت زیر را به ازای  $x = 5$  و  $y = 4$  ارزیابی کنید:

```
m = x + (y * 2) / 6 + y / 8;
```

مقادیر را در عبارت قرار می دهیم:

$$m = 5 + (4 * 2) / 6 + 4 / 8$$

$$m = 5 + 8 / 6 + 4 / 8$$

$$m = 5 + 1 + 4 / 8$$

$$m = 5 + 1 + 0 = 6$$

۴. عبارت زیر را به ازای  $x = 4$  و  $y = 3$  ارزیابی کنید:

```
m = x > y ? x * 3 : y * 3;
```

ابتدا مقادیر را در  $x$  و  $y$  قرار می دهیم:

$$m = 4 > 3 ? 3 * 3 : 4 * 3;$$

عبارت  $4 > 3$  ارزیابی می شود. چون این شرط ارزش درستی دارد،  $3 * 3$  (۹) در  $m$  قرار می گیرد.

۵. پس از اجرای دستورات زیر، چه مقداری در  $x$  قرار می گیرد؟

```
int x;
long double y;
x = sizeof y;
```

دستور اول، متغیر  $x$  را با نوع `int` تعریف می کند، دستور دوم، متغیر  $y$  را با نوع `long double` تعریف می کند و دستور سوم، طول نوع متغیر  $y$  را محاسبه کرده، در متغیر  $x$  قرار می دهد. بنابراین، ۱۰ در  $x$  قرار می گیرد (نوع `long double`، ۱۰ بایت را اشغال می کند).

۶. عبارت زیر را بدون استفاده از عملگر `*` انجام دهید:

$$y = 255 * x - 18 * x + 15;$$

ابتدا  $x * 255$  را محاسبه می کنیم. برای انجام این کار،  $x$  را ۸ بار به سمت چپ شیفت داده، از  $x$  کم می کنیم. زیرا، با هر بار شیفت، مقدار  $x$  در ۲ ضرب می شود. پس مقدار  $x$  در  $2^8$  (۲۵۶) ضرب می گردد.

## ۶۰۰ برنامه C++ با حل آن ها ۹

سپس، مقدار  $x \times 18 -$  را محاسبه کرده، ( $x$  را ۴ بار به سمت چپ شیفت داده دو بار با  $x$  جمع کرده، علامت آن را به منفی تبدیل می کنیم) در پایان، این مقادیر را با ۱۵ جمع می کنیم. یعنی، عبارت زیر را داریم:

```
y = x << 8;
y1 = y - x;
y2 = x << 4;
y2 += (x + x);
y = y1 - y2 + 15;
```

۷. پس از اجرای عبارت زیر مقدار متغیر  $m$  چه خواهد شد؟

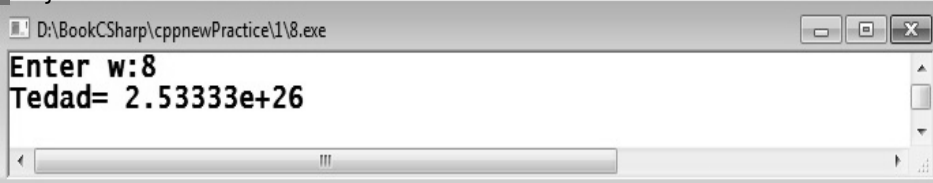
```
m = (y = 0, x = ++y, m = y, y + 2);
```

ابتدا  $y$  برابر با صفر می شود و در عبارت دوم به  $y$  یک واحد اضافه شده، در  $x$  قرار می گیرد. پس اکنون مقدار  $x$  و  $y$  برابر یک است. در عبارت بعدی،  $y$  در  $m$  قرار می گیرد و در نتیجه،  $m$  نیز برابر با یک می شود. حاصل عبارت  $y + 2$  که برابر با ۳ است در  $m$  قرار می گیرد.

۸. وزن یک مولکول آب  $2.3 \times 10^{-23}$  گرم و وزن یک لیتر آب در حدود ۹۵۰ گرم است. برنامه ای که وزن آب را بر حسب لیتر از ورودی خوانده، تعداد مولکول های آن را محاسبه می کند.

```
#include <iostream>
using namespace std;
int main()
{
    double m=3.0e-23, l=950, tedad;
    long int w;
    cout << "Enter w:";
    cin >> w;
    tedad =(w * l) / m;
    cout << "Tedad= " << tedad;
    return 0;
}
```

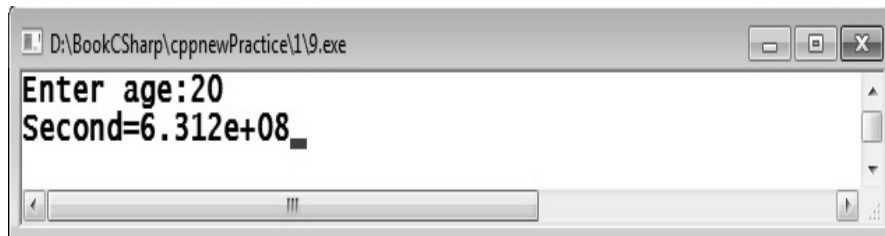
| متغیر | هدف                                    |
|-------|----------------------------------------|
| m     | وزن یک مولکول                          |
| l     | وزن یک لیتر آب                         |
| w     | وزن آب به لیتر از ورودی خوانده می شود. |
| tedad | تعداد مولکول های آب                    |



۹. هر سال برابر با  $3.156 \times 10^7$  ثانیه است. برنامه ای که سن شما به سال را دریافت کرده، به ثانیه تبدیل می کند.

```
#include <iostream>
using namespace std;
int main()
{
    double ss = 3.156e7, second;
    int age;
    cout << "Enter age:";
    cin >> age;
    second = age * ss;
    cout << "Second=" << second;
    return 0;
}
```

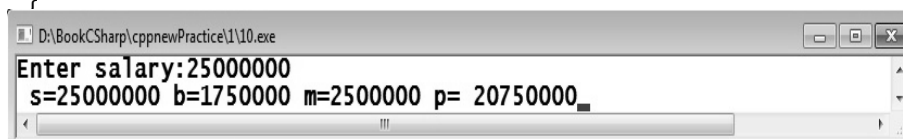
| متغیر  | هدف                 |
|--------|---------------------|
| ss     | هر سال بر حسب ثانیه |
| age    | سن به سال           |
| second | سن بر حسب ثانیه     |



۱۰. برنامه‌ای که حقوق کارمندی را خوانده، بیمه، مالیات و دریافتی او را محاسبه کرده، نمایش می‌دهد. بیمه و مالیات به ترتیب ۷ و ۱۰ درصد حقوق می‌باشند.

```
#include <iostream>
using namespace std;
int main()
{
    long int p, s, m, b;
    cout << "Enter salary:";
    cin >> s;
    b = s * 0.07;
    m = s * 0.1;
    p = s - b - m;
    cout << " s=" << s << " b=" << b << " m=" << m << " p=" << p;
    return 0;
}
```

| متغیر | هدف     |
|-------|---------|
| s     | حقوق    |
| b     | بیمه    |
| m     | مالیات  |
| p     | دریافتی |



ب

➤ برنامه‌ای که مقدار x را از ورودی خوانده، حاصل عبارت مقابل را محاسبه می‌کند:

$$y = \frac{1}{x^2 + x + 1}$$

➤ برنامه‌ای که دو عدد را از ورودی می‌خواند (عدد اول مقاومت یک مدار الکتریکی و عدد دوم جریان آن را نمایش می‌دهد). با فرمول مقابل ولتاژ را محاسبه کرده، نمایش می‌دهد:

$$\text{جریان} * \text{مقاومت} = \text{ولتاژ}$$

➤ مجموع مقاومت‌های R<sub>۱</sub>، R<sub>۲</sub> و R<sub>۳</sub> که به‌طور موازی به هم متصل‌اند، با فرمول زیر محاسبه می‌شود:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} = \frac{1}{R} = \frac{R_2 * R_3 + R_1 * R_3 + R_1 * R_2}{R_1 * R_2 * R_3}$$

برنامه‌ای که سه مقاومت را از ورودی خوانده، مقاومت کل را محاسبه می‌کند.

➤ برنامه‌ای که وزن کالایی را برحسب کیلوگرم دریافت می‌نماید و وزن آن را برحسب گرم نمایش می‌دهد. هر کیلوگرم برابر ۱۰۰۰ گرم است.

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۱

- اگر کارمندی به اندازه ۱۵ درصد حقوق ماه گذشته خود پاداش بگیرد، برنامه‌ای که حقوق کارمند را خوانده، پاداش او را محاسبه نموده، نمایش می‌دهد.
- برنامه‌ای که ارتفاع و قاعده مثلث را از ورودی خوانده، مساحت آن را محاسبه کرده، به خروجی می‌برد.  
$$\text{قاعده} * \text{ارتفاع} \times 0,5 = \text{مساحت مثلث}$$
- در شرکتی، سالانه ۱۵۰ خودکار، ۵۰ بسته کاغذ A۴ مصرف می‌شود. در پایان سال این شرکت می‌خواهد بداند در سال آینده چقدر برای این بخش از تجهیزات اداری، افزایش هزینه دارد. برنامه‌ای که قیمت این اقلام را در امسال از ورودی دریافت می‌نماید، نرخ تورم در سال آینده را نیز می‌گیرد، هزینه اضافه شده شرکت را در این بخش محاسبه می‌کند و به خروجی می‌برد. تورم به صورت درصد وارد می‌گردد که برنامه باید آن را به مقدار اعشاری تبدیل کند. به عنوان مثال، اگر تورم را ۵,۶ وارد کنیم، برنامه باید از ۰/۰۵۶ استفاده کند.
- یک دوچرخه‌سوار با سرعت  $x$  کیلومتر بر ساعت شروع به حرکت می‌کند و پس از  $n$  دقیقه سرعت آن به  $k$  کیلومتر در ساعت می‌رسد. برنامه‌ای که با استفاده از فرمول زیر شتاب او را محاسبه کرده، نمایش می‌دهد:  
$$\text{شتاب} = \frac{60 * (\text{سرعت اولیه } x) + (\text{سرعت نهایی } k)}{n(\text{زمان})}$$
- برنامه‌ای که مصرف (کیلومتر به لیتر) و گنجایش باک بنزین (لیتر) اتومبیلی را از ورودی می‌خواند و تعیین می‌کند که با یک باک بنزین اتومبیل چه مسافتی را می‌تواند پیماید.
- برنامه‌ای که دو عدد را از کاربر دریافت می‌کند و حاصل جمع، ضرب، تقسیم، تقریق و باقی‌مانده تقسیم صحیح آن‌ها را چاپ می‌کند.
- برنامه‌ای که دو عدد صحیح را از کاربر دریافت کرده، عدد بزرگ‌تر و سپس عبارت `is larger than` و عدد کوچک‌تر را چاپ می‌کند. اگر این دو عدد برابر باشند، عبارت `"These number are equal"` را چاپ می‌کند.
- برنامه‌ای که سه عدد را از صفحه کلید می‌گیرد و حاصل جمع، میانگین، حاصل ضرب، بزرگ‌ترین و کوچک‌ترین عدد را چاپ می‌کند.

- برنامه‌ای که دو عدد را از صفحه کلید می‌گیرد و تعیین می‌کند که آیا عدد اول بر عدد دوم بخش پذیر است یا نه؟
  - برنامه‌ای که دو عدد را خوانده، تعیین می‌کند آیا عدد اول مضربی از عدد دوم است یا خیر.
  - برنامه‌ای که پنج عدد صحیح را خوانده، بزرگ‌ترین و کوچک‌ترین عدد را نمایش می‌دهد.
  - با استفاده از امکاناتی که در این فصل دیدید برنامه‌ای که توان ۲، توان ۳، و عدد ۰ تا ۱۰ را به صورت زیر جدول بندی می‌نماید؟ (توضیح: همان‌طور که مشاهده نمودید، تعداد خطوط این برنامه زیاد شده است که با بیان حلقه‌های تکرار در فصل بعد، تعداد دستورات این قبیل برنامه‌ها کاهش خواهد یافت).
  - برنامه‌ای که یک عدد صحیح پنج‌رقمی را خوانده، ارقام آن را جدا می‌کند و هر یک از ارقام را با سه فاصله بین آن‌ها چاپ می‌کند. به عنوان مثال، اگر کاربر عدد ۴۲۳۳۹ را وارد کند، خروجی به صورت زیر باشد:
- Result is    ۴    ۲    ۳    ۳    ۹
- هر تن ۳۵۲۷۳،۹۲ اونس است. برنامه‌ای که وزن یک محموله را به اونس گرفته، به تن تبدیل می‌کند. برنامه به کاربر اجازه می‌دهد تا هر زمان می‌خواهد این محاسبه را تکرار کند.
  - هر لیتر معادل ۰،۲۶۴۱۷۹ گالن است. برنامه‌ای که میزان بنزین مصرفی اتومبیل کاربر را برحسب لیتر و مسافت طی شده آن را به مایل گرفته، سپس مصرف به ازای هر مایل به گالن را نمایش می‌دهد.
  - برنامه‌ای که عددی را خوانده، مربع و مکعب آن را نمایش می‌دهد.
  - برنامه‌ای که دو ضلع موازی و ارتفاع (a، b و h) ذوزنقه را دریافت کرده، مساحت آن را با فرمول زیر محاسبه می‌کند:
- $$s = h * (a + b) / 2$$
- برنامه‌ای که شعاع دایره را خوانده، محیط، مساحت و قطر آن را نمایش می‌دهد (محیط، مساحت و قطر دایره به صورت زیر حساب می‌شوند):
- $$\pi * \text{شعاع} * 2 = \text{محیط دایره}$$
- $$\pi * (\text{شعاع})^2 = \text{مساحت دایره}$$
- $$\text{شعاع} * 2 = \text{قطر دایره}$$

### ۶۰۰ برنامه C++ با حل آن‌ها ۱۳

➤ برنامه‌ای که عددی را به سال خوانده، تعیین می‌کند چند ماه، چند روز و چند ثانیه است.

$$\begin{aligned}\text{سال} &= 365,25 * \text{روز} \\ 60 * 24 * \text{روز} &= \text{ثانیه} / 30 \\ \text{ماه} &= \end{aligned}$$

➤ برنامه‌ای که  $m$  و  $n$  را خوانده، حاصل عبارات زیر را محاسبه کند:

$$a = m^2 - n^2 \qquad b = 2 * m * n \qquad c = m^2 + n^2$$

➤ برنامه‌ای که دو عدد را خوانده، در متغیرهای  $a$  و  $b$  قرار می‌دهد، سپس حاصل جمع، تفریق، ضرب، تقسیم،  $a^2 + b^2$  و  $a^3 + b^3$  را نمایش می‌دهد.

➤ برنامه‌ای که یک عدد سه‌رقمی را خوانده، اگر مجموع رقم‌های اول و سوم برابر رقم دوم باشد، "Yes"، وگرنه "No" را چاپ می‌کند.

➤ فرض کنید هر صفحه کتاب ۳۰ سطر (خط) است و هر خط ۸۰ بایت حافظه اشغال می‌کند. برنامه‌ای که تعداد صفحات یک کتاب و ظرفیت یک فلش را به گیگابایت خوانده، تعیین می‌کند که این فلش چند تا از این کتاب‌ها را می‌تواند در خودش ذخیره کند (گیگابایت برابر با  $(1024)^3$  بایت است).

➤ برنامه‌ای که یک عدد چهاررقمی را خوانده، اگر حاصل ضرب رقم‌های اول و چهارم، برابر حاصل جمع ارقام دوم و سوم باشد، "Yes"، وگرنه، "No" را نمایش می‌دهد.

➤ برنامه‌ای که تعداد کالا و قیمت هر کالا را خوانده، مبلغ فروش را نمایش می‌دهد (مبلغ فروش برابر با تعداد کالا \* قیمت کالا است).

➤ برنامه‌ای که سن‌تان را به سال خوانده، مشخص کند چند دقیقه زندگی کرده‌اید (هر سال ۳۶۵,۲۵ روز است و هر روز  $(60)^2 \times 24$  دقیقه می‌باشد).

➤ برنامه‌ای که مقداری را به مایل و فوت دریافت کرده، به متر و کیلومتر تبدیل می‌کند و نمایش می‌دهد (هر فوت ۰,۳۰۴۸ متر و هر مایل ۱۶۰۹,۳۵ متر است).

➤ برنامه‌ای که  $x$  و  $y$  را از ورودی خوانده، حاصل عبارت مقابل را نمایش می‌دهد:

$$Z = x++ * y / x++;$$

➤ برنامه‌ای که یک حرف انگلیسی کوچک را خوانده، به حرف بزرگ تبدیل کرده و نمایش می‌دهد (فاصله بین حرف بزرگ و کوچک ۳۲ کاراکتر است. یعنی، چنانچه از یک حرف کوچک ۳۲ تا کم کنیم به حرف بزرگ تبدیل می‌شود).

➤ برنامه‌ای که یک کاراکتر را خوانده و تعیین می‌کند که بین ۰ تا ۹ است یا خیر؟

➤ برنامه‌ای که x و y را خوانده، حاصل عبارت مقابل را نمایش می‌دهد:

$$z = x^3 + 2x^2 + 3y - 5$$

➤ برنامه‌ای که نمره‌ای دانشجویی را دریافت کرده، اگر نمره کم‌تر از ۱۰ بود، کلمه "Failed"، وگرنه کلمه "Passed" را نمایش می‌دهد (در این برنامه متغیر grade، نمره دانشجو است).

➤ برنامه‌ای که دو عدد را خوانده، بدون استفاده از متغیر کمکی محتوی آن‌ها را تعویض می‌کند.

➤ برنامه‌ای که یک حرف کوچک را خوانده، با استفاده از عملگر بیتی آن را به حرف بزرگ تبدیل می‌کند (کد اسکی حروف کوچک ۳۲ واحد از حروف بزرگ بیش‌تر است). برای این منظور، اگر کاراکتر حرف کوچک باشد، آن را با مقدار (۲۵۵ - ۳۲) بیت به بیت و (&) بیتی می‌نماییم تا تبدیل به حرف بزرگ شود.

➤ برنامه‌ای که قاعده و ارتفاع مثلث را خوانده، مساحت مثلث را محاسبه می‌کند (مساحت مثلث برابر با ارتفاع ضرب‌در نصف قاعده است).

➤ پروژه برنامه‌نویسی ۱: برنامه‌ای که ۵ عدد را خوانده، بزرگ‌ترین و کوچک‌ترین عدد را نمایش می‌دهد.

➤ پروژه برنامه‌نویسی ۲: برنامه‌ای که یک عدد ۵ رقمی را خوانده، ارقام عدد را بافاصله نمایش می‌دهد.

## ➤ ساختار تصمیم و حلقه تکرار

۱. برنامه‌ای که شماره دانشجویی و معدل تعداد  $n$  دانشجو را از ورودی خوانده، دانشجویی که دومین معدل را از نظر بزرگی دارد، پیدا می‌کند و به خروجی می‌برد (توضیح: چون هیچ‌یک از دانشجویان معدل‌شان کوچک‌تر از صفر نمی‌باشد، لذا اولین بار معدل دو دانشجویی که بالاترین معدل را دارند، صفر در نظر گرفتیم).

```
#include <iostream>
using namespace std;
int main()
{
    int id1=-1, id2, i, n, id;
    float aver, max1=-1, max2;
    cout << "Enter n:";
    cin >> n;
    if (n < 2)
    {
        cout << "Please enter a number greater than 1";
        return 0;
    }
    for (i=1; i <= n; i++)
    {
        cout << "Enter id, average(" << i << "):";
        cin >> id >> aver;
        if (aver > max1)
        {
            id2 = id1;
            max2 = max1;
            max1 = aver;
            id1 = id;
        }
        else if(aver > max2)
        {
            max2 = aver;
            id2 = id;
        }
    }
    cout.setf(ios::fixed);
    cout.setf(ios::showpoint);
    cout.precision(2);
    cout << "\nMax2 = " << max2 << "\tId2 = " << id2;
    return 0;
}
```

| متغیر | هدف                                   |
|-------|---------------------------------------|
| n     | تعداد دانشجویان                       |
| max1  | اولین معدل                            |
| max2  | دومین معدل                            |
| i     | شمارنده                               |
| id1   | شماره دانشجویی که اولین معدل را دارد. |
| id2   | شماره دانشجویی که دومین معدل را دارد. |
| aver  | معدل دانشجو                           |
| id    | شماره دانشجو                          |

```
D:\BOOKCSHARP\CPPNEWPRACTICE\2\1.exe
Enter n:4
Enter id, average(1):1 10
Enter id, average(2):2 14
Enter id, average(3):3 11
Enter id, average(4):4 1.5

Max2 = 11.00    Id2 =3
```

۲. برنامه‌ای که تعدادی عدد را از ورودی خوانده تشخیص می‌دهد که آیا اعداد موردنظر کامل هستند یا خیر. عددی کامل است که مجموع مقسوم‌علیه‌های آن (به‌جز خودش) برابر با آن عدد باشد، پس از بررسی هر عدد از کاربر سؤال کند که می‌خواهد به کارش ادامه دهد یا خیر.

```
#include <iostream>
using namespace std;
int main()
{
    int sum, num, i;
    char yes;
    do
    {
        cout << "Enter a number:";
        cin >> num;
        sum = 0;
        for (i=1; i < num; i++)
            if (num % i == 0) sum += i;
        if (sum == num)
            cout << "\t Perfected";
        else
            cout << "\t Not perfect";
        cout << "\nContinue ?";
        cin >> yes;
        cout << endl;
        yes -= 32;
    }
    while(yes == 'Y');
    return 0;
}
```

| متغیر | هدف                 |
|-------|---------------------|
| num   | اعداد موردنظر       |
| i     | شمارنده از ۱ تا num |
| sum   | مجموع مقسوم‌علیه‌ها |
| yes   | تعیین ادامه کار     |

```
D:\BOOKCSHARP\CPPNEWPRACTICE\2\2.exe
Enter a number:6
Perfected
Continue ?y
Enter a number:10
Not perfect
Continue ?n
```

۳. برنامه‌ای که تعداد n جمله از سری فیبوناچی را تولید می‌کند.

سری فیبوناچی      ۱      ۱      ۲      ۳      ۵      ۸      ۱۳      ...

```
#include <iostream>
using namespace std;
int main()
{
    int n, f1 = 1, f2 = 1, f3, i;
    cout << "Enter a number:";
    cin >> n;
    if (n == 1)
    {
        cout << f1;
```

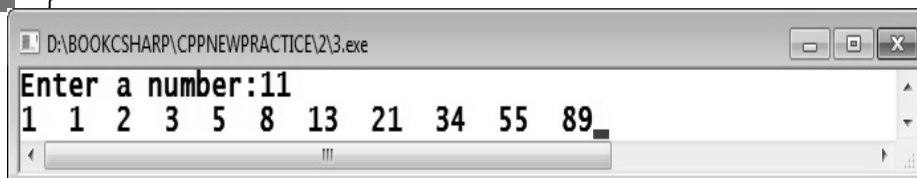
| متغیر | هدف            |
|-------|----------------|
| i     | شمارنده        |
| f1    | دو عدد قبل سری |
| f2    | عدد قبل سری    |
| f3    | عدد فعلی سری   |
| n     | تعداد جملات    |

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۷

```

    return 0;
}
else if (n==2)
{
    cout << f1 << " " << f2;
    return 0;
}
cout << f1 << " " << f2;
for(i=3 ; i <= n; i++)
{
    f3 = f1 + f2;
    cout << " " << f3;
    f1 = f2;
    f2 = f3;
}
return 0;
}

```



۴. برنامه‌ای که شماره کارمندی و حقوق تعدادی از کارکنان موسسه را دریافت کرده، بر اساس تعریف زیر، مالیات حقوق را محاسبه می‌کند و به خروجی می‌برد. سپس مشخص می‌کند بیش‌ترین دریافتی مربوط به کدام کارمند است.

مالیات از معاف  $\leq 400000$  حقوق

۱۰ درصد به مازاد بر  $400000$   $400000 < \text{حقوق} \leq 500000$

۱۵ درصد مازاد  $500000 < \text{حقوق} \leq 700000$

۱۷ درصد مازاد  $700000 < \text{حقوق}$

| متغیر  | هدف                                       |
|--------|-------------------------------------------|
| id     | شماره کارمندی                             |
| salary | حقوق                                      |
| tax    | مالیات                                    |
| pay    | دریافتی                                   |
| i      | شمارنده                                   |
| id1    | شماره کارمندی که بیش‌ترین دریافتی را دارد |
| max    | بیش‌ترین دریافتی                          |
| n      | تعداد کارمندان                            |

```

#include <iostream>
using namespace std;
int main()
{
    long int i, salary, tax, pay, max=0, id1=0, id, n;
    cout << "Enter a number:";
    cin >> n;
    for (i=1; i <= n; i++) {
        cout << "Please enter id, salary:";
        cin >> id >> salary;
        if (salary <= 400000)
            tax = 0;
        else if (salary >= 400001 && salary <= 500000)
            tax = (salary - 400000) * 0.10;
    }
}

```

```

else if(salary >= 500001 && salary <= 700000)
    tax = (500000 - 400001) * 0.10 + (salary - 500001) * 0.15;
else if(salary >= 700001)
    tax = (500000 - 400001) * 0.10 + (salary - 500001) * 0.15 +
        (salary - 700001) * 0.17;
pay = salary - tax;
if (pay > max)
{
    max = pay;
    id1 = id;
}
cout<<"\n Salary="<<salary<<"Tax="<<tax<<"Pay="<< pay << endl;
}
cout << "\n Max=" << max << "    Id=" << id1;
return 0;
}

```

The screenshot shows a console window titled "D:\BookCSharp\cppnewPractice\2\4.exe". It displays the following output:

```

Enter a number:4
Please enter id, salary:1 400000

Salary=400000 Tax=0 Pay=400000
Please enter id, salary:2 450000

Salary=450000 Tax=5000 Pay=445000
Please enter id, salary:3 1000000

Salary=1000000 Tax=135999 Pay=864001
Please enter id, salary:4 650000

Salary=650000 Tax=32499 Pay=617501

Max=864001 Id=3_

```

۵. برنامه‌ای که سال تولد کاربر و سال فعلی را از ورودی خوانده، مشخص می‌کند چند سال، چند ماه، چند روز، چند ساعت، چند دقیقه و چند ثانیه عمر کرده است.

```

#include <iostream>
using namespace std;
int main()
{
    int byy, bdd, bmm, cyy, cdd,
    cmm, year, month, day;
    double hh, ss, mm, days;
    cout << "\n Enter birth
date:";
    cin >> byy >> bmm >> bdd;
    cout << "\n Enter current
date:";
    cin >> cyy >> cmm >> cdd;
    if (cdd < bdd)
    {
        cmm --;
        cdd += 30;
    }
    day = cdd-bdd;
    if (cmm < bmm)
    {
        cyy --;
        cmm += 12;
    }
    month = cmm - bmm;
    year = cyy - byy;
}

```

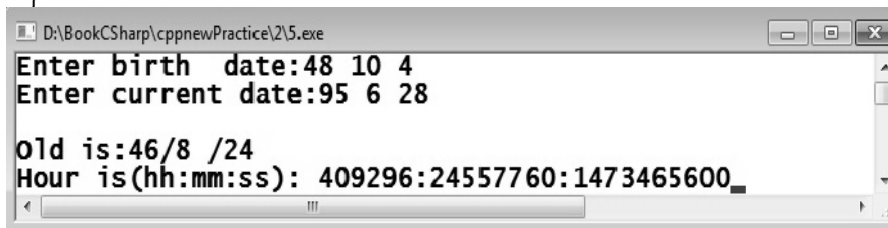
| متغیر | هدف                            |
|-------|--------------------------------|
| byy   | سال تولد                       |
| bmm   | ماه تولد                       |
| bdd   | روز تولد                       |
| cyy   | سال فعلی                       |
| cmm   | ماه فعلی                       |
| cdd   | روز فعلی                       |
| year  | سال‌هایی که زندگی کرده         |
| month | ماه‌هایی که زندگی کرده         |
| day   | روزهایی که زندگی کرده          |
| hh    | تعداد ساعاتی که زندگی کرده     |
| mm    | تعداد دقیقاتی که زندگی کرده    |
| ss    | تعداد ثانیه‌هایی که زندگی کرده |
| days  | تعداد کل روزهایی که زندگی کرده |

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۹

```

days = day + month * 30 + year * 365;
hh = days * 24;
mm = hh * 60;
ss = mm * 60;
cout << "\nOld is:" << year << "/" << month << "/" << day;
cout << "\nHour is(hh:mm:ss): " << (long int) hh << ":" << (long
int)mm << ":" << (long int)ss;
return 0;
}

```



➤ برنامه‌ای که سه مقدار را از ورودی خوانده، آن‌ها را به ترتیب صعودی در خروجی چاپ می‌کند.

➤ برنامه‌ای که شماره کارمندی، ساعت کارکرد و دستمزد ساعتی کارکنان موسسه‌ای را خوانده، حقوق آن‌ها را محاسبه می‌کند. اگر کارمندی بیش از ۴۰ ساعت کار کرده باشد، اضافه‌کار به او تعلق می‌گیرد. به ازای هر ساعت کاری،  $\frac{3}{2}$  دستمزد ساعتی به‌عنوان اضافه‌کاری پرداخت می‌شود.

➤ یکی از مسائل مهم شرکت‌ها و ادارات پیش‌بینی بودجه چند سال بعد است. برنامه‌ای که قیمت کالا (ارزش پروژه) و نرخ تورم کالا را گرفته، قیمت کالا را برای چند سال بعد پیش‌بینی می‌نماید (برنامه تعداد سال‌هایی که قیمت کالا باید پیش‌بینی شود، از کاربر دریافت می‌نماید).

➤ برنامه‌ای که مجموع ده جمله اول سری زیر را محاسبه و چاپ می‌کند.

$$s = \frac{1}{x} - \frac{1}{x+2x^2} + \frac{1}{x+2x^2+3x^3} - \frac{1}{x+2x^2+3x^3+4x^4} + \dots$$

➤ برنامه‌ای که مجموعه‌ای از اعداد دوتایی صحیح را خوانده، حاصل ضرب آن‌ها را بدون استفاده از عملگر ضرب نمایش می‌دهد. چنانچه کاربر به‌جای دو عدد، صفر وارد کند، برنامه خاتمه یابد.

➤ برنامه‌ای که عددی را خوانده، مشخص می‌کند آیا عدد متقارن است یا خیر. نمونه‌های از اعداد متقارن عبارت‌اند از: ۱۲۴۲۱ و ۳۲۴۲۳. (توضیح: چنانچه عددی برابر با مغلوبش باشد، متقارن است. پس، ابتدا مغلوب عدد را محاسبه کرده، با خودش مقایسه می‌کنیم.)

➤ برنامه‌ای که  $m$  و  $n$  دو عدد صحیح و مثبت را خوانده، با استفاده از عملگر  $+$ ،  $m^n$  را محاسبه و چاپ می‌نماید.

➤ برنامه‌ای که تعدادی عدد را خوانده، هر یک از اعداد را که بر ۹ بخش پذیر باشد، در خروجی چاپ می‌کند (از روش مجموع ارقام استفاده کنید).

➤ برنامه‌ای که اطلاعات تعدادی کارمند از قبیل شماره کارمندی، سال استخدام، کد جنسیت، (۰: زن، ۱: مرد) پایه استخدام (۹-۱) و حقوق را خوانده، موارد زیر را در خروجی چاپ می‌کند.

الف. تعداد کارمندان مرد      ب. تعداد کارکنان زن      ج. میانگین حقوق به تفکیک زن و مرد

➤ فروشگاه‌های را در نظر بگیرید که درصدی از فروش به فروشنده اختصاص می‌دهد. میزان درصد به صورت زیر محاسبه می‌گردد:

کالای درجه ۱: اگر میزان فروش کم‌تر یا مساوی ۱۰۰۰۰۰۰ باشد، ۶ درصد به فروشنده تخصیص می‌یابد.

اگر میزان فروش ۱۰۰۰۰۰۰ تا ۲۰۰۰۰۰۰ باشد ۷٪ و گرنه ۱۰٪ به فروشنده اختصاص می‌یابد.

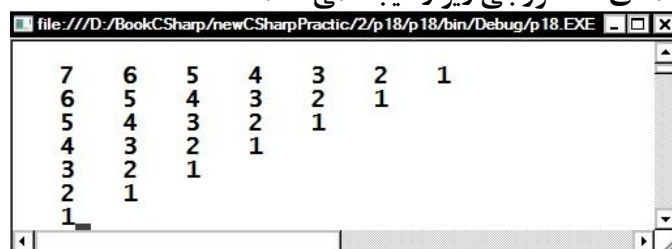
کالای درجه ۲: اگر میزان فروش کم‌تر از ۱۰۰۰۰۰ باشد، ۴٪ و گرنه ۶٪ به فروشنده اختصاص می‌یابد.

کالای درجه ۳: در این نوع کالا، ۴۱٪ میزان فروش به فروشنده تخصیص می‌یابد.

کالای درجه ۴: در این نوع کالا، ۵٪ میزان فروش به فروشنده اختصاص می‌یابد.

➤ برنامه‌ای که تعداد اقلام فروش را دریافت می‌کند. سپس به تعداد خوانده شده درجه کالا، قیمت واحد کالا، تعداد فروش را می‌خواند و مجموع فروش و مبلغی را چاپ می‌نماید که باید به فروشنده تخصیص یابد.

➤ برنامه‌ای که خروجی زیر را ایجاد می‌کند.



|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| 6 | 5 | 4 | 3 | 2 | 1 |   |
| 5 | 4 | 3 | 2 | 1 |   |   |
| 4 | 3 | 2 | 1 |   |   |   |
| 3 | 2 | 1 |   |   |   |   |
| 2 | 1 |   |   |   |   |   |
| 1 |   |   |   |   |   |   |

➤ بازی سنگ، کاغذ و قیچی دو بازیکن دارد. بازیکن اول می‌تواند قیچی (مقدار ۱)، سنگ (مقدار ۲) و یا کاغذ (مقدار ۳) را انتخاب کند و بازیکن دوم همین‌طور. اگر انتخاب دو بازیکن یکسان باشد، مساوی‌اند. اگر متفاوت باشد، یک نفر برنده است و برنده یک امتیاز را کسب می‌کند. برای مثال:

اگر بازیکن اول قیچی و بازیکن دوم کاغذ را انتخاب کند، بازیکن اول برنده است. چون قیچی کاغذ را می‌برد.

## ۶۰۰ برنامه C++ با حل آن‌ها ۲۱

اگر بازیکن اول قیچی و بازیکن دوم سنگ را انتخاب کند، بازیکن دوم برنده است. زیرا سنگ قیچی را می‌شکند.

اگر بازیکن اول سنگ و بازیکن دوم کاغذ را انتخاب کند، بازیکن دوم برنده است. زیرا کاغذ دور سنگ می‌پیچد.

برنامه‌ای که این مسابقه را ۵ دور بین دو بازیکن انجام می‌دهد و برنده را نمایش می‌دهد (بازیکن‌ها از انتخاب هم دیگر خبر ندارند).

➤ برنامه‌ای که تمام اعداد بین ۱۰۰۰ تا ۱۱۰۰ که بر ۹ بخش پذیرند را نمایش می‌دهد (عددی بر ۹ بخش پذیر است که مجموع ارقام آن بر ۹ قابل قسمت باشد).

➤ برنامه‌ای که کاراکتری را که نشان‌دهنده رنگی می‌باشد، از ورودی خوانده، می‌گوید چه رنگی را انتخاب کردید. مثلاً اگر 'r' یا R را وارد کردید، برنامه به شما می‌گوید دوست دارید رنگ قرمز (Red) را انتخاب کنید. برای خروج کاربر باید e یا E را وارد کنید.

➤ برنامه‌ای که دو عدد مثبت را از ورودی خوانده، آنگاه عدد بزرگ‌تر را به روش تفریق بر عدد کوچک‌تر تقسیم می‌کند.

➤ برنامه‌ای که n را می‌خواند و حاصل عبارت زیر را محاسبه می‌کند:

$$\frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}$$

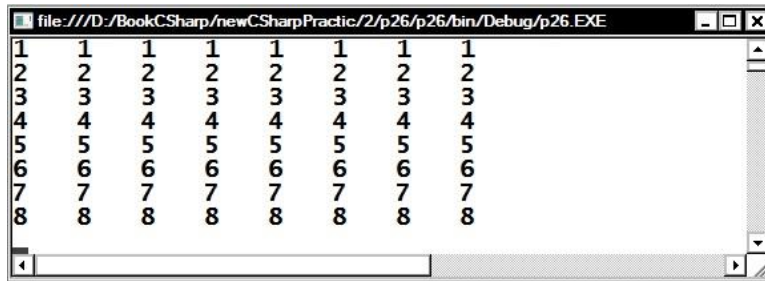
➤ برنامه‌ای که با استفاده از حلقه تودرتو خروجی زیر را ایجاد می‌کند:

```
file:///D:/BookCSharp/newCSharpPractic/2/p24/p24/bin/Debug/p24.EXE
$$$$$$
$$$$$$
$$$$$$
$$$$$$
```

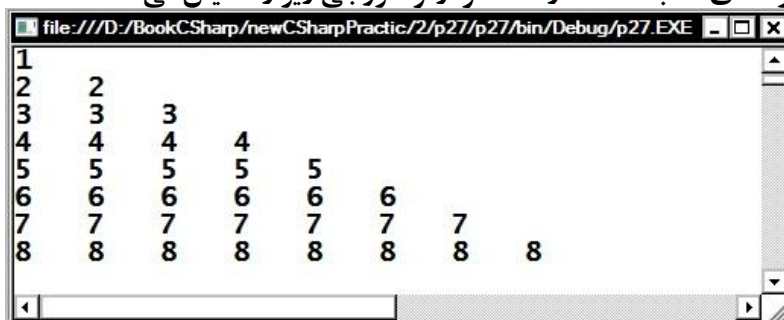
➤ برنامه‌ای که با استفاده از حلقه تودرتو خروجی زیر را چاپ می‌کند:

```
file:///D:/BookCSharp/newCSharpPractic/2/p25/p25/bin/Debug/p25.EXE
1 1 1 1 1 1 1 1
2 2 2 2 2 2
3 3 3 3 3
4 4 4 4
5 5 5
6 6
7
8
```

➤ برنامه‌ای که با استفاده از حلقه تودرتو خروجی زیر را چاپ می‌کند:



➤ برنامه‌ای که با استفاده از حلقه تودرتو خروجی زیر را نمایش می‌دهد:



➤ برنامه‌ای که القاب افراد را از ورودی به صورت یک کاراکتر خوانده، سپس لقب آن را به صورت کامل در خروجی چاپ می‌کند (بر اساس جدول زیر):

| علامت ورودی | لقب     | لقب لاتین  |
|-------------|---------|------------|
| b, B        | بانو    | lady       |
| d, D        | دوشیزه  | miss       |
| p, P        | پروفسور | professor  |
| a, A        | آقا     | mr         |
| j, J        | جناب    | excellency |
| m, M        | خانم    | wife       |

➤ قیمت اتومبیل در انتهای هر سال ۲۰٪ کاهش می‌یابد (استهلاک اتومبیل ۲۰٪ است).  
برنامه‌ای که قیمت یک اتومبیل را از ورودی خوانده، تا ده سال آینده قیمت آن را در پایان هر سال نمایش می‌دهد.

➤ برنامه‌ای که عددی را از ورودی خوانده، تشخیص می‌دهد که این عدد جزء سری فیبوناچی است یا خیر؟

➤ برنامه‌ای که تمام حالت‌هایی که یک اسکناس ۱۰۰۰ ریالی را با ۵۰۰، ۲۰۰، ۱۰۰ و ۵۰ ریالی می‌توان خرد نمود می‌نویسد.

- برای رانندگان میزان مصرف سوخت خودرویشان از اهمیت ویژه‌ای برخوردار است. راننده‌ای چند بار مقدار بنزین باک خودروی خود را به همراه مقدار مسافتی که رانندگی کرده، ثبت نموده است. برنامه‌ای که مسافت را به مایل و مصرف بنزین را به گالن از کاربر می‌گیرد و نسبت مایل به گالن را برای هر بار محاسبه می‌کند و نمایش می‌دهد. برنامه پس از پردازش همه اطلاعات نسبت مایل به گالن کل را محاسبه نموده، چاپ می‌کند. برای خروج از برنامه به جای گالن مقدار ۰ را وارد کنید.

- برنامه‌ای که عدد N را از ورودی خوانده، خروجی زیر را چاپ می‌کند:

| I  | 10*I | 100*I | 1000*I |
|----|------|-------|--------|
| 1  | 10   | 100   | 1000   |
| 2  | 20   | 200   | 2000   |
| 3  | 30   | 300   | 3000   |
| 4  | 40   | 400   | 4000   |
| 5  | 50   | 500   | 5000   |
| 6  | 60   | 600   | 6000   |
| 7  | 70   | 700   | 7000   |
| 8  | 80   | 800   | 8000   |
| 9  | 90   | 900   | 9000   |
| 10 | 100  | 1000  | 10000  |

- برنامه‌ای که دنباله‌ای از اعداد صحیح را باهم جمع می‌کند. فرض کنید که اولین عدد صحیحی که خوانده می‌شود، تعداد اعدادی را که باید وارد شوند، مشخص می‌کند. به عنوان مثال، ورودی به صورت زیر می‌باشد:

➤ ۸۰۰ ۲۷۰ ۲۰۰ ۱۷۰ ۱۵۰ ۱۰۰ ۶

- برنامه میانگین و مجموع اعداد ۱۰۰، ۱۵۰، ۱۷۰، ۲۰۰، ۲۷۰ و ۸۰۰ را نمایش می‌دهد.

- برنامه‌ای که مقدار سری  $\pi$  (سری زیر) را حداکثر تا  $n$  جمله محاسبه می‌کند و نمایش می‌دهد (n را از ورودی می‌خواند):

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \dots$$

- برنامه جدولی چاپ می‌کند که حاصل مقدار  $\pi$  را با تقریب یک جمله، دو جمله، سه جمله، ...،  $n$  جمله نمایش می‌دهد.

- برنامه‌ای که سه ضلع (عدد صحیح غیر صفر) را می‌خواند و تشخیص می‌دهد آیا سه ضلع اضلاع یک مثلث هستند یا خیر. (توضیح: سه ضلع زمانی تشکیل مثلث را می‌دهند که مجموع هر دو ضلع بیش‌تر از ضلع سوم باشد.)

- برنامه‌ای که سه عدد صحیح غیر صفر را گرفته، تشخیص دهد این سه عدد تشکیل مثلث قائم‌الزاویه را می‌دهند یا خیر؟ زمانی که توان ۲ یکی از اضلاع برابر مجموع توان ۲ اضلاع دیگر باشد مثلث قائم‌الزاویه است. (توضیح: هدف از این برنامه کاربرد عملگر || (یا منطقی) می‌باشد.)

➤ نمره دانشجویان یک کلاس به صورت زیر محاسبه می شود:

در طول ترم دو کوئیز برگزار می گردد که هر کدام ۱۰ نمره دارند.

امتحان پایان ترم ۵۰ درصد نمره کل، امتحان میان ترم ۲۵ درصد نمره کل و ۲۵ درصد دیگر مربوط به کوئیزها است.

برنامه ای که نمرات کوئیز میان ترم و پایان ترم یک دانشجو را گرفته، نمره کل او را محاسبه می کند و به صورت حروفی نمایش می دهد (برای نمرات بزرگ تر یا مساوی ۹۰، حروف A، نمرات بزرگ تر یا مساوی ۸۰ و کم تر از ۹۰ حرف B، نمرات بزرگ تر یا مساوی ۷۰ و کم تر از ۸۰ حرف C، نمرات بزرگ تر یا مساوی ۶۰ و کم تر از ۶۰ حرف D و نمرات زیر ۶۰ حرف E را نمایش دهد). (توضیح: در این برنامه، ابتدا نمرات کوئیز را به ۱۰۰ تبدیل می کنیم. یعنی، نمره کوئیز اول + نمره کوئیز دوم را در ۵ ضرب می کنیم. سپس برای محاسبه نمره کل، نمره به دست آمده از کوئیز را در ۰٫۲۵ ضرب کرده، با ۰٫۲۵ نمره میان ترم جمع نمود. حاصل را با ۰٫۵ پایان ترم جمع می نماییم تا نمره کل به ۱۰۰ به دست آید).

➤ برنامه ای که تعیین می کند در یک فروشگاه آیا میزان خرید مشتری از سقف اعتبار حسابش بیش تر شده است یا خیر. برای هر مشتری اطلاعات زیر موجود می باشد:

۱. شماره حساب (عدد صحیح)
۲. تراز اول ماه
۳. مجموع همه اقلامی که در یک ماه توسط این مشتری خریداری شده است.
۴. مجموع همه اعتباراتی که در یک ماه به حساب مشتری وارد شده است.
۵. سقف اعتبار مجاز

این برنامه باید هر یک از اطلاعات ۱ تا ۵ را بگیرد و تراز جدید (= اعتبارات - هزینه ها + تراز اولیه) را محاسبه و تعیین کند که آیا تراز جدید از سقف اعتباراتشان بیش تر شده است. برنامه باید شماره حساب، سقف اعتبارات و تراز مشتری را به همراه پیام 'Credit Limited' 'Exceeded' (از سقف اعتبارات تجاوز کرده است) چاپ کند.

➤ برنامه ای که n را خوانده، علاوه بر نمایش عبارت زیر، مجموع عبارت را نیز نمایش می دهد:

$$مجموع = 1 + 2 + 3 + \dots + n$$

➤ برنامه ای که n را خوانده، علاوه بر نمایش عبارت زیر، حاصل عبارت را نیز نمایش می دهد:

$$حاصل عبارت = 1 - 2 + 3 - 4 + \dots \pm n$$

➤ برنامه ای که n را خوانده، علاوه بر نمایش عبارت زیر، حاصل عبارت را نیز نمایش می دهد:

$$حاصل عبارت = -1 + 2 - 3 + 4 + \dots \pm n$$

## ۶۰۰ برنامه C++ با حل آن‌ها ۲۵

- برنامه‌ای که n را خوانده، علاوه بر نمایش عبارت زیر، حاصل عبارت را نیز نمایش می‌دهد:

$$\text{حاصل عبارت} = 1 - \frac{2}{2!} + \frac{3}{3!} - \frac{4}{4!} + \dots \pm \frac{n}{n!}$$

- برنامه‌ای که x و n را خوانده، علاوه بر نمایش عبارت زیر، حاصل عبارت را نیز نمایش می‌دهد:

$$\text{حاصل عبارت} = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \frac{x^4}{4!} + \dots \pm \frac{x^n}{n!}$$

- برنامه‌ای که x و n را خوانده، علاوه بر نمایش عبارت زیر، حاصل عبارت را نیز نمایش می‌دهد:

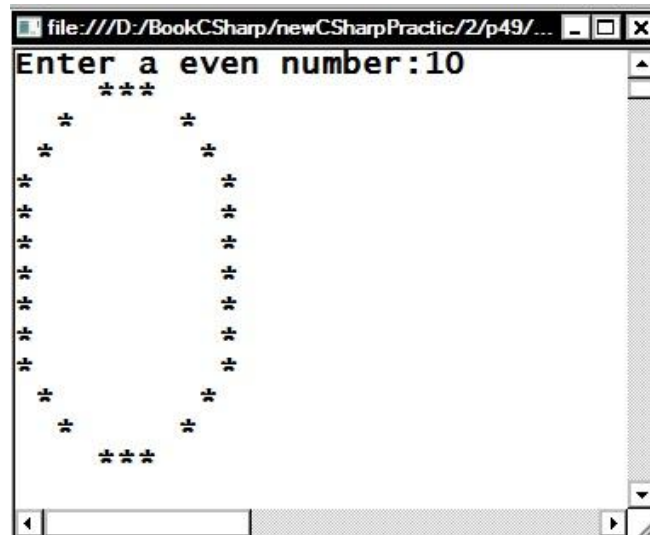
$$\text{حاصل عبارت} = \frac{x^2}{2!} - \frac{x^4}{4!} + \frac{x^6}{6!} - \dots \pm \frac{x^n}{n!}$$

- برنامه‌ای که یک عدد، یک کاراکتر و سپس عدد دیگری را خوانده، اعمال زیر را انجام دهد:

اگر کاراکتر ورودی "A"، "a" یا "+" باشد، حاصل جمع دو عدد را نمایش دهد.  
اگر کاراکتر ورودی "S"، "s" یا "-" باشد، حاصل تفریق دو عدد را نمایش دهد.  
اگر کاراکتر ورودی "X"، "x" یا "\*" باشد، حاصل ضرب دو عدد را نمایش دهد.  
اگر کاراکتر ورودی "D"، "d" یا "/" باشد، حاصل تقسیم دو عدد را نمایش دهد.  
اگر کاراکتر ورودی "M"، "m" یا "%" باشد، باقی‌مانده تقسیم صحیح عدد اول بر عدد دوم را نمایش دهد.  
در غیر این صورت، پیغام "Invalid operator" را نمایش دهد.

- برنامه قبلی را با دستور switch پیاده‌سازی می‌کند.
- برنامه‌ای که دو عدد را خوانده، اگر باقی‌مانده تقسیم هر عدد بر عدد دیگر صفر شود، "Yes"، وگرنه، "No" را نمایش دهد.

- برنامه‌ای که عدد فرد n را خوانده، خروجی زیر را نمایش می‌دهد:



➤ برنامه‌ای که کاراکتر به همراه کد اسکی تمام کاراکترهای رایانه را نمایش می‌دهد (متغیر i شمارنده‌ای که صفر تا ۲۵۵ را می‌شمارد. کد اسکی کاراکترهای رایانه از صفر تا ۲۵۵ می‌باشد).

➤ برنامه‌ای که خروجی زیر را تولید کرده، نمایش می‌دهد.

|    |    |    |    |    |    |    |    |    |     |
|----|----|----|----|----|----|----|----|----|-----|
| 1  |    |    |    |    |    |    |    |    |     |
| 2  | 4  |    |    |    |    |    |    |    |     |
| 3  | 6  | 9  |    |    |    |    |    |    |     |
| 4  | 8  | 12 | 16 |    |    |    |    |    |     |
| 5  | 10 | 15 | 20 | 25 |    |    |    |    |     |
| 6  | 12 | 18 | 24 | 30 | 36 |    |    |    |     |
| 7  | 14 | 21 | 28 | 35 | 42 | 49 |    |    |     |
| 8  | 16 | 24 | 32 | 40 | 48 | 56 | 64 |    |     |
| 9  | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 |     |
| 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 |

➤ برنامه‌ای که تعدادی عددی را دریافت کرده، معادل ماه آن‌ها را نمایش می‌دهد(اگر عدد وارد شده خارج از بازه ۱ تا ۱۲ باشد، برنامه خاتمه می‌یابد). در این برنامه month عددی است که ماه معادل آن باید تعیین شود.

➤ برنامه‌ای که رشته‌ای را دریافت کرده، تعداد کاراکترهای رشته را می‌شمارد (رشته به Enter ختم می‌شود).

➤ برنامه‌ای که حقوق تعدادی کارمند را خوانده، اگر کارمند بیش از ۴۰ ساعت کار کرده باشد، به اندازه ۱,۵ برابر هر ساعت به او اضافه کاری داده می‌شود. در پایان، کل حقوق دریافتی را نمایش می‌دهد(برای خروج از برنامه از کاربر سؤال می‌کند که آیا ادامه می‌دهد یا خیر؟ اگر کاربر صفر را وارد کند، برنامه خاتمه می‌یابد).

## ۶۰۰ برنامه C++ با حل آن ها ۲۷

➤ برنامه‌ای که نتایج عملگرهای (AND) &&, (OR) ||, (NOT) ! را به عدد تبدیل کرده، نمایش دهد.

➤ برنامه‌ای که عددی را خوانده، در هر مرحله عدد موردنظر، رقم یکان و حاصل ضرب ارقام فرد آن را نمایش می‌دهد. به عنوان مثال، اگر کاربر عدد ۳۲۷۶ را وارد نماید خروجی به صورت زیر نمایش داده می‌شود:

| Number      | Digit | Multiply |
|-------------|-------|----------|
| 3276        | 6     | 1        |
| 327         | 7     | 7        |
| 32          | 2     | 7        |
| 3           | 3     | 21       |
| Multiply is |       | 21       |

➤ برنامه‌ای که عددی را خوانده، میانگین ارقام فرد و زوج آن را نمایش می‌دهد. به عنوان مثال، اگر عدد ۲۵۶۹ باشد، میانگین ارقام فرد آن، یعنی  $(۵+۹)/۲$  (۷) است. میانگین ارقام زوج آن  $(۲+۶)/۲$  (۴) می‌باشد.

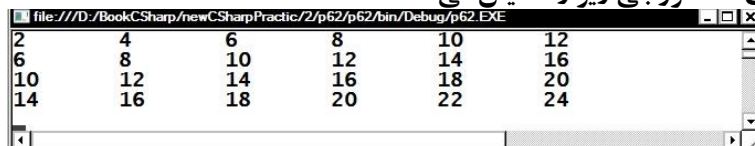
➤ برنامه‌ای که کلیه اعداد سه رقمی را نمایش می‌دهد که با ارقام ۱، ۲ و ۳ تولید می‌شوند.

➤ برنامه‌ای که کلیه اعداد سه رقمی را نشان می‌دهد که در ارقام آن ۱، ۴ یا ۸ باشد.

➤ برنامه‌ای که کلیه اعداد چهاررقمی که فقط از ارقام ۱، ۲، ۴ و ۹ تشکیل شده باشند و هر رقم یک بار تکرار شده باشد را نمایش می‌دهد.

➤ برنامه‌ای که یک جمله را خوانده، تعداد ارقام ۰ تا ۹ آن را می‌شمارد (جمله با Ctrl + z یعنی EOF خاتمه می‌یابد).

➤ برنامه‌ای که خروجی زیر را نمایش می‌دهد:



|    |    |    |    |    |    |
|----|----|----|----|----|----|
| 2  | 4  | 6  | 8  | 10 | 12 |
| 6  | 8  | 10 | 12 | 14 | 16 |
| 10 | 12 | 14 | 16 | 18 | 20 |
| 14 | 16 | 18 | 20 | 22 | 24 |

➤ برنامه‌ای که خروجی زیر را نمایش می‌دهد:



➤ برنامه‌ای که x و n را خوانده، حاصل عبارت  $\sum_{i=1}^n x^n$  را نمایش می‌دهد (یعنی، حاصل عبارت زیر را نمایش می‌دهد):

$$x^1 + x^2 + x^3 + \dots + x^n$$

➤ برنامه‌ای که  $x$ ،  $n$  و  $k$  را خوانده، حاصل عبارت  $F(x, n, k) = \frac{\sum_{j=n}^k x^j}{\sum_{i=1}^k i!}$  را نمایش می‌دهد.

➤ برنامه‌ای که  $n$  (عددی بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```

file:///D:/BookCSharp/newCSharpPractic/2/p66/p66/bin/Debug/p66.EXE
Enter n:17
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****

```

➤ برنامه‌ای که  $n$  (عددی بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```

file:///D:/BookCSharp/newCSharpPractic/2/p67/p67/bin/Debug/p67.EXE
Enter n:8
*****
*****
*****
*****
*****
*****
*****
*****

```

➤ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی مانند زیر را نمایش می‌دهد.

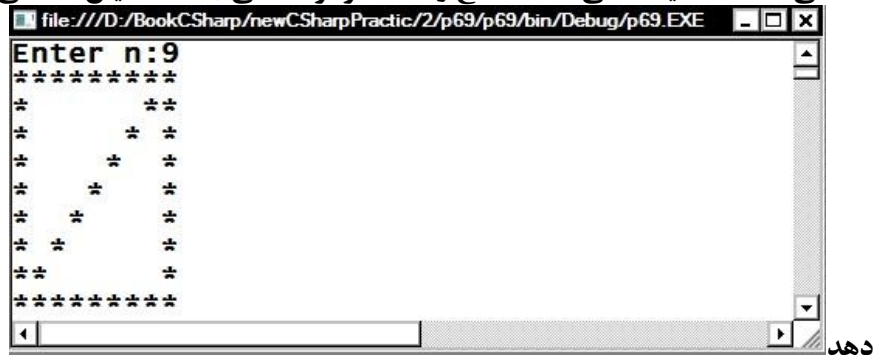
```

file:///D:/BookCSharp/newCSharpPractic/2/p68/p68/bin/Debug/p68.EXE
Enter n:8
*****
*****
*****
*****
*****
*****
*****
*****

```

## ۶۰۰ برنامه C++ با حل آن‌ها ۲۹

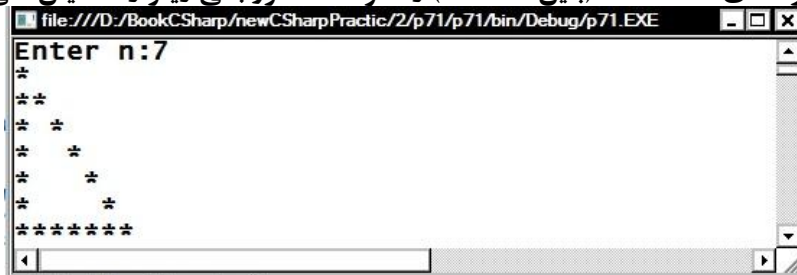
➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را از ورودی خوانده، خروجی مثل زیر را نمایش می‌دهد. یعنی، اضلاع و قطر فرعی را نمایش می‌دهد.



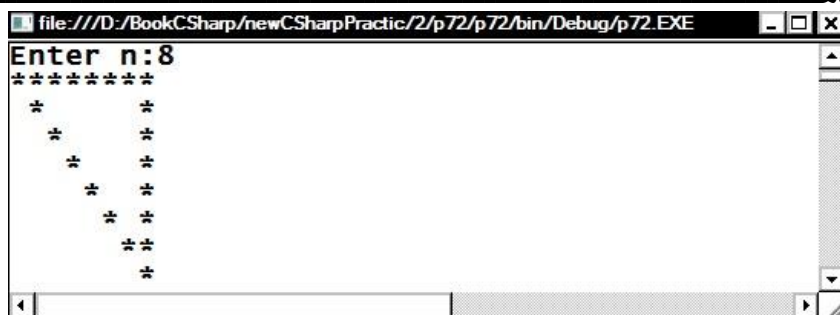
➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد.



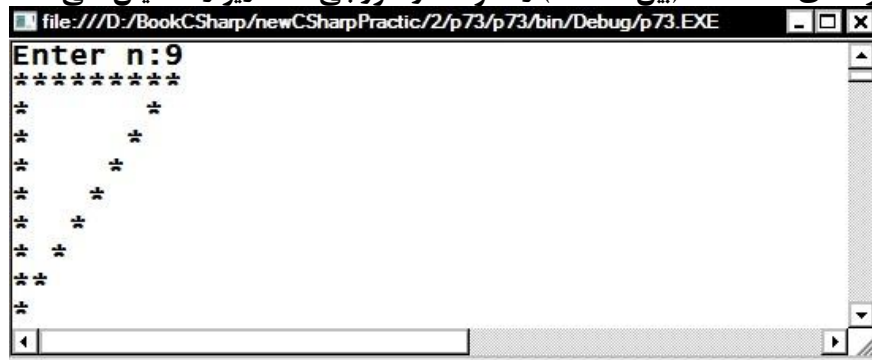
➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد.



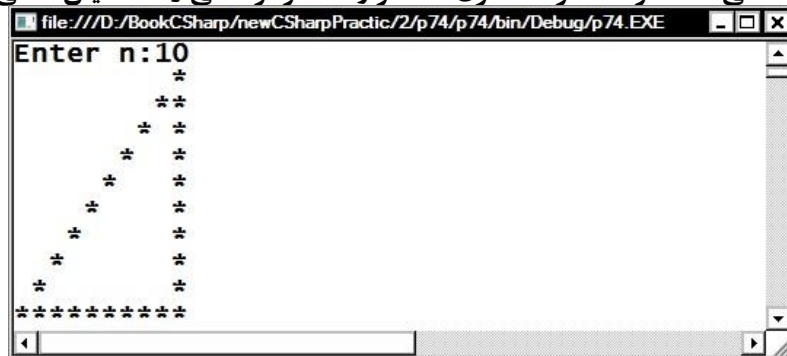
➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مانند شکل زیر را نمایش می‌دهد.



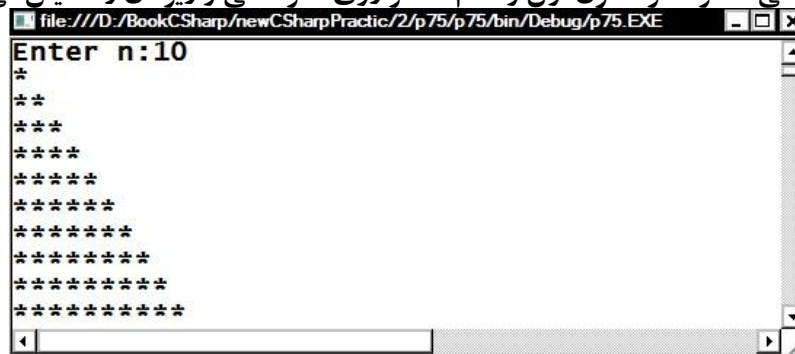
➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده و خروجی مانند زیر را نمایش می‌دهد.



➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. یعنی، سطر آخر، ستون آخر و قطر فرعی را نمایش می‌دهد.

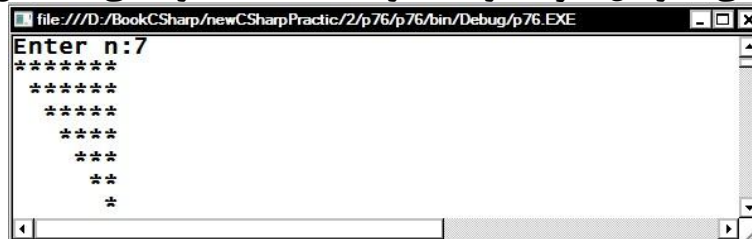


➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. یعنی، سطر آخر، ستون اول و تمام عناصر روی قطر اصلی و زیر آن را نمایش می‌دهد.



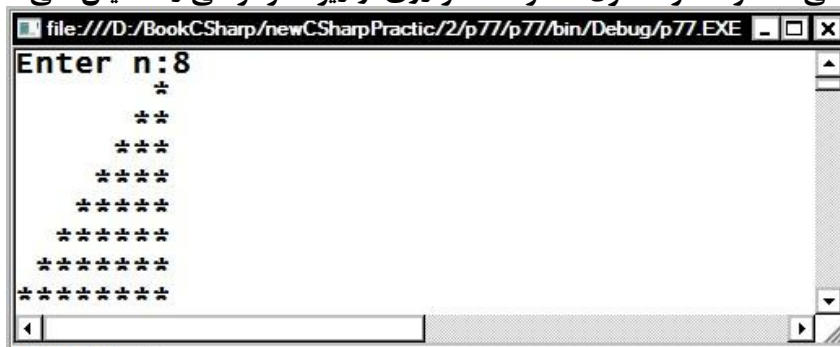
### ۶۰۰ برنامه C++ با حل آن‌ها ۳۱

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. یعنی، سطر اول، ستون آخر و عناصر روی و بالای قطر اصلی را نمایش می‌دهد.



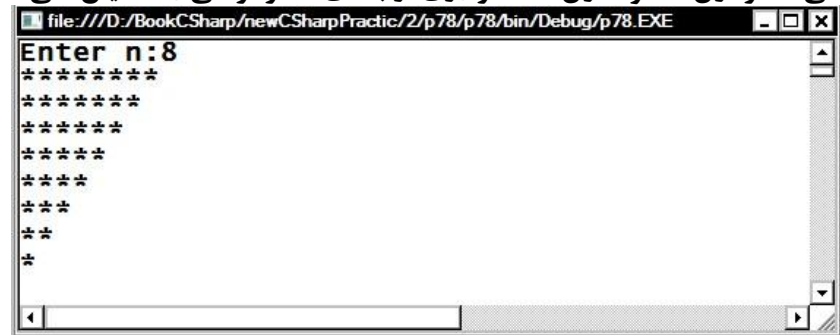
```
file:///D:/BookCSharp/newCSharpPractic/2/p76/p76/bin/Debug/p76.EXE
Enter n:7
*****
*****
*****
*****
*****
*****
*****
```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. یعنی، سطر آخر، ستون آخر، عناصر روی و زیر قطر فرعی را نمایش می‌دهد.



```
file:///D:/BookCSharp/newCSharpPractic/2/p77/p77/bin/Debug/p77.EXE
Enter n:8
*
**
***
****
*****
*****
*****
*****
```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. یعنی، سطر اول، ستون اول، عناصر روی و بالای قطر فرعی را نمایش می‌دهد.



```
file:///D:/BookCSharp/newCSharpPractic/2/p78/p78/bin/Debug/p78.EXE
Enter n:8
*****
*****
*****
*****
*****
*****
*****
*****
```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد.

```

Enter n:8
*
**
***
****
***
**
*

```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```

Enter n:9
*****
*
*
*
*
*
*
*
*

```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```

Enter n:10
*
**
***
****
*****
*****
****
***
**
*

```

### ۶۰۰ برنامه C++ با حل آن ها ۳۳

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد.

```
file:///D:/BookCSharp/newCSharpPractic/2/p82/p82/bin/Debug/p82.EXE
Enter n:10
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```
file:///D:/BookCSharp/newCSharpPractic/2/p83/p83/bin/Debug/p83.EXE
Enter n:8
*
**
***
****
*****
*****
*****
*****
*
```

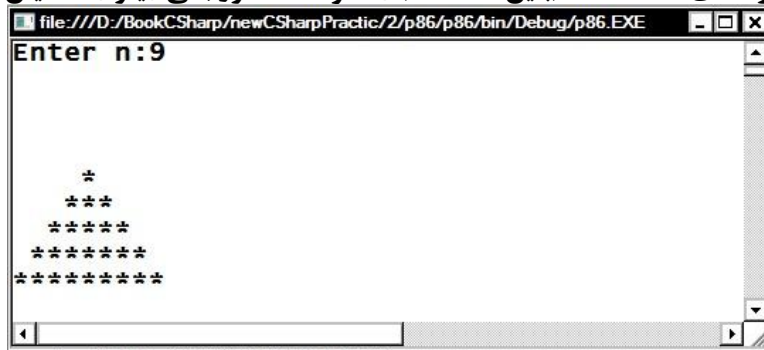
➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```
file:///D:/BookCSharp/newCSharpPractic/2/p84/p84/bin/Debug/p84.EXE
Enter n:10
*
**
***
****
*****
*****
*****
*****
*****
*
```

➤ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.

```
file:///D:/BookCSharp/newCSharpPractic/2/p85/p85/bin/Debug/p85.EXE
Enter n:9
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

- برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد.



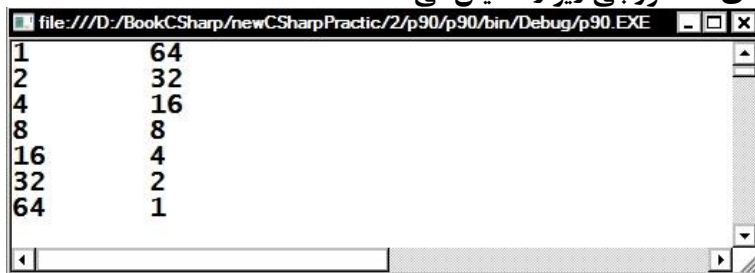
- برنامه‌ای که حقوق کارمند، نوع کسور بیمه و درصد مالیات را خوانده، حقوق خالص کارمند را حساب می‌نماید. برای محاسبه حقوق خالص به صورت زیر عمل می‌گردد (برای حل این تمرین از عملگر ؟ استفاده کنید):

اگر نوع کسور بیمه ۱ باشد،  $۷ / ۱۰۰ * \text{حقوق} = \text{بیمه}$   
 اگر نوع کسور بیمه ۲ باشد،  $۸,۵ / ۱۰۰ * \text{حقوق} = \text{بیمه}$   
 اگر نوع کسور بیمه ۳ باشد،  $۱۰ / ۱۰۰ * \text{حقوق} = \text{بیمه}$   
 $۱۰۰ / \text{درصد مالیات} * \text{حقوق} = \text{مالیات}$   
 $\text{مالیات} - \text{بیمه} - \text{حقوق} = \text{حقوق خالص}$

- برنامه‌ای که دو عدد صحیح x و y را خوانده، با استفاده از عملگر ضرب،  $x^y$  را محاسبه می‌کند و نمایش می‌دهد:

- برنامه‌ای که تعداد عدد را خوانده و تشخیص می‌دهد هر عدد خوانده شده دارای رقم صفر است یا خیر. برای خروج از برنامه کاربر باید ۱- را وارد کند.

- برنامه‌ای که خروجی زیر را نمایش می‌دهد:



- برنامه‌ای که شماره خانوار، هزینه روزانه آن‌ها را به روز می‌خواند، هزینه ماهیانه و سالیانه آن‌ها را نمایش می‌دهد. سپس تعیین می‌کند کدام خانوار کم‌ترین هزینه و کدام خانوار بیش‌ترین هزینه را دارد. برای خروج از برنامه به جای شماره خانوار عدد ۹۹- وارد می‌شود.

- برنامه‌ای که عدد  $n$  را خوانده ( $n$  تعداد کارمندان است). سپس ساعات اضافه کار  $n$  کارمند را خوانده و سه کارمند که کم‌ترین اضافه کار را دارند، نمایش می‌دهد.
- برنامه‌ای که سن و جنسیت تعدادی افراد را خوانده، تعداد و درصد مردانی که سن آن‌ها بین ۱۸ تا ۲۰ است را محاسبه می‌کند و نمایش می‌دهد. اگر جنسیت  $M$  یا  $m$  وارد گردید، این فرد، مرد است (برای خاتمه برنامه به جای سن ۱- وارد می‌شود).
- یک شرکت بیمه به بیمه‌گذاران خود سود سالانه می‌دهد. برنامه‌ای که میزان سود سالانه (درصد سود سالانه) و میزان سرمایه‌گذاری یک بیمه‌گذار و تعداد سال‌های سرمایه‌گذاری را دریافت می‌کند و سود بیمه‌گذار را برای  $n$  سال مختلف حساب می‌نماید (بدون پرداخت سود به بیمه‌گذار یعنی سود به سرمایه سالانه اضافه گردد). سرمایه بیمه‌گذار پس از  $n$  سال چقدر است.
- برنامه‌ای که تمام اعداد چهاررقمی را نمایش می‌دهد که حاصل ضرب ارقام آن‌ها عدد تام (کامل) است. عددی تام است که مجموع مضرب‌های اعداد کوچک‌تر از خودش برابر خودش باشد. به‌عنوان مثال، دو عدد زیر تام هستند:  

$$۶ = ۱ + ۲ + ۳$$

$$۲۸ = ۱ + ۲ + ۴ + ۷ + ۱۴$$
- برنامه‌ای که تعداد رقم‌های صفر اعداد چهاررقمی را می‌شمارد.
- برنامه‌ای که سه عدد را خوانده، اعداد بین اعداد اول و دوم که مجموع ارقام آن‌ها برابر عدد سوم باشد را نمایش می‌دهد. به‌عنوان مثال، اگر کاربر ۵، ۱۰۰۰ و ۱۵ را وارد کند. برنامه باید اعداد از ۵ تا ۱۰۰۰ که مجموع ارقام آن‌ها برابر ۱۵ باشد را نمایش دهد.
- برنامه‌ای که دو عدد را خوانده، اعداد اولی که بین این دو عدد قرار دارند را نمایش می‌دهد (عددی اول است که بر هیچ عدد کوچک‌تر از نصف خودش به جزء یک بخش پذیر نباشد).
- برنامه‌ای که یک عدد صحیح را خوانده و ارقامی از آن عدد که بر ۲ یا ۳ بخش پذیر هستند را نمایش می‌دهد.
- هنگام خرید خانه عوامل مختلفی را باید در نظر گرفت. در هنگام خرید خانه، میزان هزینه تقریبی سوختی که برای یک سال مصرف می‌شود و نرخ مالیات سالانه داده‌شده‌اند. برنامه‌ای که داده‌هایی از قبیل قیمت خانه، هزینه سوخت سالانه و مالیات

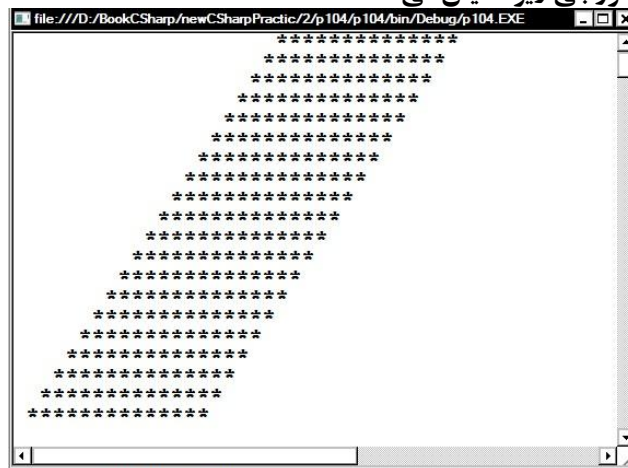
سالانه را دریافت می کند. این برنامه هزینه ۱۰ ساله خانه را محاسبه می نماید. برنامه تا زمانی که کاربر به جای قیمت خانه ۹۹۹- وارد کند، ادامه می یابد.

➤ برنامه ای که عدد  $n$  را خوانده، مقدار  $\sqrt{n + \sqrt{n + \sqrt{n + \dots}}}$  را تا دقت  $10^{-10}$  محاسبه می کند و نمایش می دهد.

➤ در یک بازی دونفره  $n$  چوب کبریت وجود دارد. هر یک از بازیکنان می توانند در نوبت خود، ۱، ۲ یا حداکثر ۳ چوب کبریت را بردارند. بازیکنی که آخرین چوب کبریت را برمی دارد، بازنده است. برنامه ای که این بازی را بین دو بازیکن اجرا می کند. این برنامه، ابتدا تعداد چوب کبریت ها را خوانده و در هر مرحله تعداد چوب کبریت های باقی مانده را نمایش می دهد.

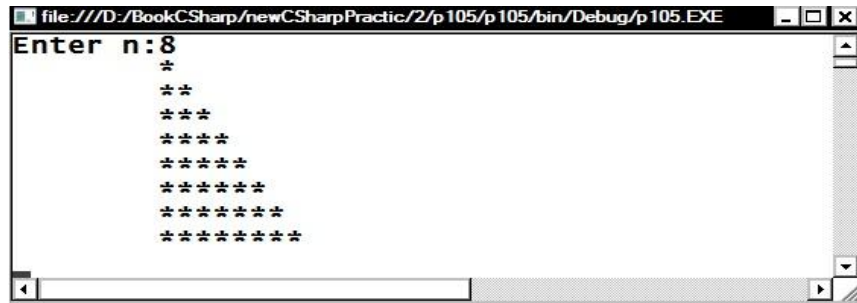
➤ برنامه ای که اگر کاربر هر یک از حروف A, B, C و D را وارد کند، پیام Big Character! را نمایش می دهد. اگر کاربر یکی از حروف a, b, c یا d را وارد کند، Litter character! را نمایش خواهد داد. ولی اگر کاربر یکی از کاراکترهای ۱، ۲، ۳ یا ۴ را وارد نماید، پیام A Figurer! را نمایش می دهد. وگرنه پیام Unknown Character! را نمایش می دهد (هدف این برنامه or کردن مقادیر در دستور switch است).

➤ برنامه ای که خروجی زیر نمایش می دهد:



➤ برنامه ای که  $n$  را خوانده، خروجی مانند زیر نمایش را می دهد:

## ۶۰۰ برنامه C++ با حل آن‌ها ۳۷



➤ ۱۰۶. برنامه‌ای که سه ضلع مثلث را خوانده و مشخص می‌کند که آیا مثلث متساوی‌الاضلاع P، متساوی‌الساقین است یا مختلف‌الاضلاع می‌باشد (در این برنامه متغیرهای a، b و c اضلاع مثلث را نگهداری می‌کنند).

➤ ۱۰۷. برنامه‌ای که سه ضلع مثلث را دریافت کرده، اگر مثلث قائم‌الزاویه بود، مساحت آن، وگرنه محیط آن را محاسبه کرده و نمایش می‌دهد. در این برنامه متغیرهای a، b و c اضلاع مثلث را نگهداری می‌کنند، اما متغیر temp کمکی است که محیط یا مساحت را ذخیره می‌نماید.

➤ ۱۰۸. برنامه‌ای که کاراکتری را خوانده و مشخص می‌کند که در کدام یک از دسته‌های زیر قرار دارد؟  
☐ حروف بی‌صدا    ☐ حروف صدادار    ☐ ارقام    ☐ سایر کاراکترها

➤ ۱۰۹. برنامه‌ای که دو عدد صحیح را خوانده و مقسوم‌علیه مشترکشان را نمایش می‌دهد. در این برنامه x و y دو عدد هستند و i شمارنده از ۱ تا x می‌باشد.

➤ ۱۱۰. برنامه‌ای که ابتدا n را خوانده، سپس n معدل را برای n دانشجو می‌خواند و بزرگ‌ترین معدل مشروط را نمایش می‌دهد. در این برنامه n، تعداد دانشجویان، grade، معدل هر دانشجو و max بزرگ‌ترین معدل مشروط شده است.

➤ ۱۱۱. برنامه‌ای که معدل n دانشجو را خوانده، میانگین معدل‌های بالاتر از ۱۵ را نمایش می‌دهد. در این برنامه n (تعداد دانشجویان)، grade (معدل هر دانشجو)، sum (مجموع معدل‌های بالای ۱۵)، count (تعداد معدل‌های بالای ۱۵) و i (شمارنده) است.

➤ ۱۱۲. برنامه‌ای که حاصل سری زیر را تا ۴ رقم اعشار محاسبه کرده، نمایش می‌دهد:

$$+\frac{1}{2}-\left(\frac{1}{2}\times\frac{1}{3}\right)+\left(\frac{1}{2}\times\frac{1}{3}\times\frac{1}{4}\right)-\dots$$

➤ ۱۱۳. برنامه‌ای که ابتدا تعداد دانشجویان ( $n$ ) را خوانده، سپس  $n$  معدل را می‌خواند و تعداد دانشجویایی که معدل آن‌ها بین ۱۰ تا ۱۵ است را شمارش می‌کند و نمایش می‌دهد. در این برنامه  $n$  (تعداد دانشجویان)،  $grade$  (معدل هر دانشجو)،  $i$  (شمارنده) و  $count$  (تعداد دانشجویانی که معدل آن‌ها بین ۱۰ تا ۱۵ است).

➤ ۱۱۴. برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  جمله سری زیر را نمایش می‌دهد. در این سری زیر جمله‌ی سوم به بعد هر جمله میانگین دو جمله قبلی می‌باشد.  
 $1, 5, 3, 4, 3.5, \dots$

➤ ۱۱۵. برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  عدد را می‌خواند و واریانس آن را نمایش می‌دهد.

➤ پروژه برنامه‌نویسی ۱: برنامه‌ای که روزی از سال را خوانده، مشخص می‌کند که چه فصلی، چه ماه و چه روزی از سال است. به عنوان مثال، روز ۳۵ سال، فصل بهار، ماه اردیبهشت و روز چهارم این ماه است.

➤ پروژه برنامه‌نویسی ۲: برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را محاسبه می‌کند.

$$\frac{x^n n^x (x+n)^{n+x}}{n! x! (n+x)!} \sum_{i=1}^n \frac{x^i i^x}{n! i! (n+i)!}$$

➤ پروژه برنامه‌نویسی ۳: شرکت مخابرات ایران برای مکالمات راه دور از نرخ‌های زیر استفاده می‌کند:

تمام مکالماتی که بین ساعات ۲۳ شب تا ۸ صبح انجام می‌شود، نرخ مکالمات ۵۰٪ محاسبه می‌گردد.

برای مکالماتی که در روزهای تعطیل (روز ۷) انجام می‌شود، نرخ مکالمه ۷۵ درصد محاسبه می‌شود.

برای بقیه ساعات روزهای دیگر مکالمه به طور کامل حساب می‌شود.

نرخ هر پالس ۴۶ ریال می‌باشد.

به کلیه مکالمات ۴ درصد مالیات تعلق می‌گیرد.

برنامه‌ای که برای تعدادی مشترک، ساعت شروع مکالمه (از ساعت صفر تا ۲۳)، تعداد پالس-های مکالمه، روز انجام مکالمه را دریافت می‌کند، مبلغ ناخالص مکالمه، مالیات، مبلغ خالص، مجموع مالیات و مجموع مبلغ خاص را محاسبه و نمایش می‌دهد (برای خاتمه کاربر به جای ساعت شروع ۹۹- را وارد نماید).

➤ پروژه برنامه‌نویسی ۴: نرخ هر مترمکعب آب با توجه به نرخ مصرف از قبیل مصارف خانگی، تجاری یا صنعتی فرق می‌کند. برنامه‌ای که میزان مصرف آب به مترمکعب و نوع مصرف تعدادی مشترک را خوانده، مبلغ صورت حساب آب آن‌ها را محاسبه و چاپ می‌کند. برای محاسبه مصرف صورت حساب آب به روش زیر عمل می‌نماید:

اگر نوع مصرف حرف H یا h باشد (مصرف خانگی)، به ازای هر ۱۰۰ مترمکعب مصرف ۵۰۰ ریال دریافت می‌شود.

اگر نوع مصرف حرف I یا i وارد شود (مصرف صنعتی تا چهار میلیون مترمکعب)، به ازای هر ۱۰۰۰ مترمکعب آب ۷۵۰ ریال و به ازای هر مترمکعب بیش از آن ۰/۰۰۰۲۵ به مبلغ قبض اضافه می‌گردد.

اگر نوع مصرف حرف E یا e وارد شود (مصرف تجاری)، در صورتی که مصرف آب تا دو میلیون مترمکعب باشد، به ازای هر ۱۵۰۰ مترمکعب مصرف ۶۰۰ ریال و به ازای هر مترمکعب بیش از دو میلیون ۰/۰۰۰۰۴ به مبلغ صورت حساب اضافه خواهد شد.

برای خاتمه برنامه کاربر به جای مصرف آب، عدد منفی وارد می‌نماید.

## فصل ۳

### توابع در C++

۱. برنامه‌ای که  $n$  جمله سری فیبوناچی را به روش بازگشتی تولید می‌کند.

وظایف توابع:

تابع `main`: این تابع متغیرهای برنامه را تعریف کرده، تابع `fibonacci` را فراخوانی می‌کند.

تابع `fibonacci`: این تابع آمارین عدد فیبوناچی را تولید می‌کند.  $i$  را به عنوان پارامتر می‌پذیرد.

```
#include <iostream>
using namespace std;
int fibonacci(int);
int main()
{
    int n,vi;
    cout << "Enter n:";
    cin >> n;
    cout << "Result is:";
    for(i=1; i <= n; i++)
        cout << "    " << fibonacci(i);
    return 0;
}
int fibonacci(int n)
{
    if (n==1)
        return(1);
    if (n==2)
        return(1);
    return(fibonacci(n-2)+fibonacci(n-1));
}
```

| تابع | متغیر | هدف                  |
|------|-------|----------------------|
| main | n     | تعداد اعداد فیبوناچی |
|      | i     | شمارنده حلقه         |

```
D:\BookCSharp\cppnewPractice\3\1.exe
Enter n:10
Result is: 1 1 2 3 5 8 13 21 34 55
```

۲. برنامه‌ای که فاصله‌ای را بر حسب فوت و اینچ دریافت کرده، معادل آن را بر حسب متر و سانتی‌متر بیان می‌کند. هر فوت ۰/۳۰۴۸ متر، یک متر ۱۰۰ سانتی‌متر و هر فوت ۱۲ اینچ است. حداقل از سه تابع استفاده کنید. یکی برای ورودی، یکی برای انجام محاسبات و یکی برای خروجی.

وظایف توابع:

تابع `main`: این تابع، توابع `read`، `write` و `calculate` را فراخوانی می‌کند.

تابع `read`: این تابع فاصله را به فوت و اینچ می‌خواند.

تابع `calculate`: این تابع، هر یک از فواصل بر حسب فوت و اینچ را به متر و سانتی‌متر تبدیل می‌کند.

تابع `write`: این تابع برای نمایش فواصل تبدیل شده به کار می‌رود.

| تابع | متغیر         | هدف           |
|------|---------------|---------------|
|      | foot          | ورودی به فوت  |
|      | Inch          | ورودی به اینچ |
|      | foot_to_meter | فوت به متر    |
|      | inch_to_meter | اینچ به متر   |

|                  |                    |       |
|------------------|--------------------|-------|
| سانتیمتر         | centi_meter        | عمومی |
| فوت به سانتیمتر  | foot_to_centimeter |       |
| اینچ به سانتیمتر | inch_to_centimeter |       |

```
#include <iostream>
using namespace std;
void read();
void calculate();
void write();
float foot, inch, foot_to_meter, inch_to_meter;
float centi_meter, foot_to_centimeter, inch_to_centimeter;
int main()
{
    read();
    calculate();
    write();
    return 0;
}
void read()
{
    cout<< "Foot?";
    cin >> foot;
    cout<< "Inch?";
    cin >> inch;
}
void calculate()
{
    foot_to_meter=0.3048*foot;
    foot_to_centimeter=100*foot_to_meter;
    inch_to_meter=(float)(1.0/12)*0.3048*inch;
    inch_to_centimeter=100*inch_to_meter;
}
void write()
{
    cout.setf(ios::fixed);
    cout.setf(ios::showpoint);
    cout.precision(3);
    cout << "The " << foot << " foot is "
        << foot_to_meter << " meter";
    cout << "\nThe " << foot << " foot is "
        << foot_to_centimeter << " centiMeter";
    cout << "\nThe " << inch << " inch is "
        << inch_to_meter << " meter";
    cout << "\nThe " << inch << " inch is"
        << inch_to_centimeter << " centiMeter";
    return 0;
}
```

```
D:\BookCSharp\cppnewPractice\3\2.exe
Foot?10
Inch?12
The 10.000 foot is 3.048 meter
The 10.000 foot is 304.800 centiMeter
The 12.000 inch is 0.305 meter
The 12.000 inch is30.480 centiMeter
```

۳. نیروی جاذبه بین دو جسم به صورت زیر محاسبه می گردد:

$$f = \frac{g * m1 * m2}{d^2}$$

$m_1$  جرم جسم اول،  $m_2$  جرم جسم دوم،  $d$  فاصله بین دو جسم و  $g$  ثابت جهانی جاذبه با مقدار  $۶,۶۹۳ \times ۱۰^{-۸}$  cm(g,sec) است. برنامه ای که با استفاده از دو تابع مقدار نیروی جاذبه را محاسبه می کند.

```
#include <iostream>
using namespace std;
double calc_f(double,double,double);
double pow(double);
const double g = 6.673e-8;
int main()
{
    double m1 ,m2, d;
    cout << "Enter m1, m2 and d:";
    cin >> m1 >> m2 >> d;
    cout.setf(ios::fixed);
    cout.setf(ios::showpoint);
    cout.precision(20);
    cout << endl << " F = " << calc_f(m1,m2,d);
    return 0;
}
double calc_f(double m1, double m2, double d)
{
    return ( g*m1*m2/pow(d));
}
double pow(double d)
{
    return ( d*d);
}
```

| تابع   | متغیر | هدف               |
|--------|-------|-------------------|
| سراسری | g     | نیروی جاذبه جهانی |
| Main   | m1    | جرم جسم اول       |
|        | m2    | جرم جسم دوم       |
|        | d     | فاصله بین دو جسم  |

```
D:\BookCSharp\cppnewPractice\3\3.exe
Enter m1, m2 and d:700000000 80000000 5000000
F = 0.00014947520000000000
```

۴. یکی از مشکلات هواشناسی محاسبه شاخص سردی باد است. برای این منظور، فرمول زیر وجود دارد:

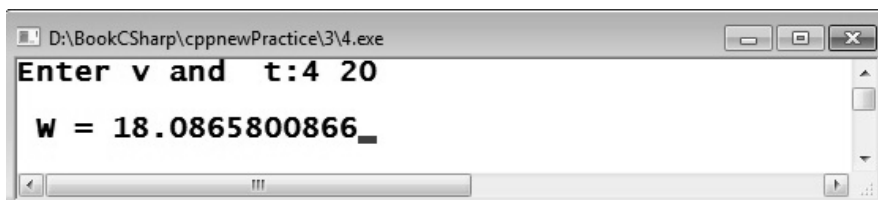
$$w = 33 - \frac{(10\sqrt{V} - V + 10.5)(33 - t)}{23 - 1}$$

$V$ ، سرعت باید بر حسب متر بر ثانیه،  $t$  دما بر حسب درجه سانتی گراد، ( $t \leq 10$ ) و  $W$  شاخص سردی باد است. برنامه ای که با استفاده از یک تابع، سرعت باد و دما را می خواند، شاخص سردی باد را نمایش می دهد.

## ۶۰۰ برنامه C++ با حل آن‌ها ۴۳

```
#include <iostream>
using namespace std;
#include <math.h>
double calc_cold(double ,double);
int main()
{
    double v, t;
    cout << "Enter v and t:";
    cin >> v >> t;
    cout.setf(ios::fixed);
    cout.setf(ios::showpoint);
    cout.precision(10);
    cout << endl << " W = " << calc_cold(v,t);
    return 0;
}
double calc_cold(double v,double t)
{
    return (33-((10*sqrt(v)-v+10.5)*(33-t))/23.1);
}
```

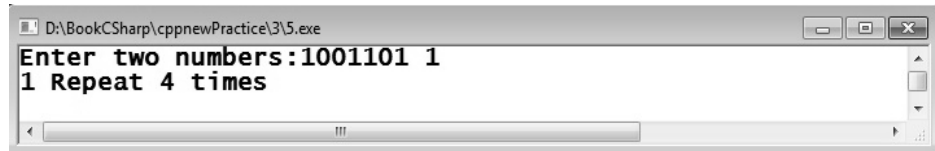
| تابع | متغیر | هدف      |
|------|-------|----------|
| main | v     | سرعت باد |
|      | t     | دما      |



۵. برنامه‌ای که یک عدد و یک رقم را خوانده، با استفاده از تابعی تعداد تکرار رقم در عدد را برمی‌گرداند و چاپ می‌کند.

```
#include <iostream>
using namespace std;
#include <math.h>
int repeat(long int ,int);
int main()
{
    double n, m;
    cout << "Enter two numbers:";
    cin >> n >> m;
    cout << +m << " Repeat " << repeat(n,m) << " times";
    return 0;
}
int repeat(long int n, int m)
{
    int count=0;
    while (n>0)
    {
        if ((n % 10 ) == m) count++;
        n/=10;
    }
    return(count);
}
```

| تابع   | متغیر | هدف                            |
|--------|-------|--------------------------------|
| main   | n     | عدد موردنظر                    |
|        | m     | رقمی که باید در عدد جستجو گردد |
| repeat | count | تعداد تکرار رقم در عدد         |



۶. برنامه‌ای که  $x$  و  $n$  را خوانده، مقدار سینوس زاویه  $x$  را محاسبه می‌کند.

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \dots + \frac{x^n}{n!}$$

وظایف توابع:

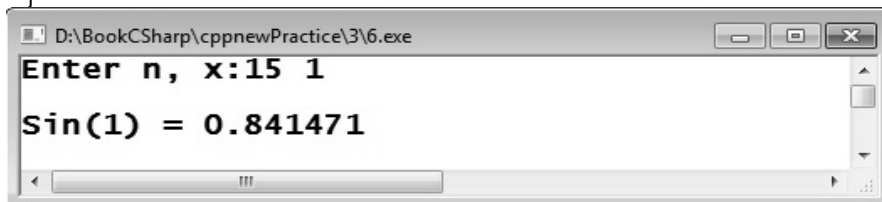
تابع **main**: این تابع  $n$  و  $x$  را خوانده، با فراخوانی توابع **fact** و **pow** حاصل عبارت  $\sin(x)$  را چاپ می‌کند.

تابع **fact**: این فاکتوریل عددی را برمی‌گرداند.

تابع **pow**: این تابع توان  $n$  عددی را برمی‌گرداند.

```
#include <iostream>
using namespace std;
double fact(int);
double pow(float,int);
int main()
{
    int n, i, sign = 1;
    float x, sum = 0;
    cout<< "Enter n, x:";
    cin >> n >> x;
    for (i=1; i<=n; i+=2)
    {
        sum+=sign*pow(x,i)/fact(i);
        sign=-sign;
    }
    cout << endl << "Sin(" << x << ") = " << sum;
    return 0;
}
double fact(int n)
{
    if (n==1)
        return 1.0;
    else
        return(n*fact(n-1));
}
double pow(float x, int n)
{
    if (n==0)
        return 1.0;
    else
        return(x*pow(x,n-1));
}
```

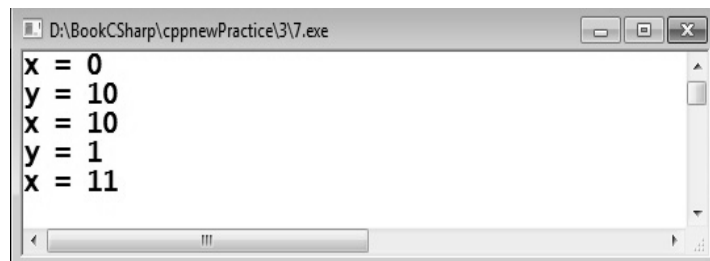
| تابع | متغیر | هدف                                    |
|------|-------|----------------------------------------|
| main | x     | ورودی                                  |
|      | n     | ورودی                                  |
|      | sum   | مجموع                                  |
|      | i     | شمارنده                                |
|      | sign  | علامت (یکی در میان مثبت و منفی می‌شود) |



## ۶۰۰ برنامه C++ با حل آن‌ها ۴۵

۷. خروجی قطعه برنامه زیر چیست؟ (توضیح برنامه: دو متغیر  $x$  و  $y$  به صورت عمومی تعریف می‌شوند و مقدار آن‌ها صفر است (متغیرهای  $x$  و  $y$  قبل از تابع `main` تعریف شده‌اند). با اجرای تابع `cout` مقدار  $x = 0$  چاپ می‌گردد. سپس تابع `f1` فراخوانی می‌گردد. با فراخوانی تابع `f1` متغیر  $y$  به صورت محلی تعریف گردیده، مقدار اولیه ۱۰ به آن تخصیص می‌یابد. سپس با اجرای `cout`، مقدار  $x = 10$  چاپ خواهد شد و مقدار متغیر عمومی  $x$  به ۱۰ تغییر می‌یابد ( $x$   $+= 10$ ). با اجرای `cout`، مقدار  $y = 10$  چاپ خواهد شد و سپس مقدار ۱۱ در  $x$  قرار می‌گیرد ( $x++$ ). پس از برگشت از تابع `f1` با اجرای `cout`، ابتدا یک واحد به متغیر عمومی  $y$  اضافه می‌گردد، مقدار ۱ در  $y$  قرار می‌گیرد و مقدار  $y = 1$  چاپ می‌شود. با اجرای آخرین `cout` مقدار  $x = 11$  چاپ می‌شود. سپس، یک واحد به متغیر  $x$  اضافه می‌گردد.

```
#include <iostream>
using namespace std;
int x, y;
void f1();
int main()
{
    cout << "x = " << x;
    f1();
    cout << "\ny = " << ++y;
    cout << "\nx = " << x++;
    return 0;
}
void f1()
{
    int y = 10;
    cout << "\ny = " << y;
    x += 10;
    cout << "\nx = " << x++;
}
```



➤ برنامه‌ای که اعداد سه‌رقمی را چاپ می‌کند که مجموع فاکتوریل ارقام آن اعداد برابر با خود عدد باشد. به عنوان مثال، اگر  $n_1$ ،  $n_2$  و  $n_3$  به ترتیب رقم یکان، دهگان و صدگان باشند، فرمول زیر برقرار باشد:

$$n_3 n_2 n_1 = n_1! + n_2! + n_3!$$

➤ برنامه‌ای که کلیه اعداد چهاررقمی چاپ را می‌کند که مجموع رقم اول به توان ۱ و رقم چهارم به توان ۴ برابر با مجموع رقم دوم به توان ۲ و رقم سوم به توان ۳ باشد. به عنوان مثال، داریم:

$$2141 = 2^4 + 1^1 = 1^2 + 4^2 = 16 + 1 = 1 + 16 = 17$$

- برنامه‌ای که دو عدد را خوانده، به متدی ارسال کند. متد، تفاضل حاصل ضرب و حاصل تقسیم آن‌ها را محاسبه کرده، به برنامه ارسال کند. متدها را به صورت قالب‌های متد پیاده‌سازی کنید و دو بار آن را برای اعداد صحیح و اعشاری فراخوانی نمایید. این متد با استفاده از Template پیاده‌سازی شود.
- برنامه‌ای که عدد صحیح مثبتی را خوانده، مجموع ارقام آن را محاسبه می‌کند. اگر حاصل مجموع ارقام، عدد یک‌رقمی نبود، این روند را ادامه داده تا نهایتاً عدد یک‌رقمی شود (از یک متد بازگشتی استفاده می‌کند).
- برنامه‌ای که اضلاع مثلثی را خوانده، مساحت آن را با متدی محاسبه می‌کند و به خروجی می‌برد. اگر اضلاع مثلثی  $a$ ،  $b$  و  $c$  باشد، مساحت آن با فرمول مقابل محاسبه می‌شود.
 
$$p = (a + b + c) / 2$$

$$\sqrt{p(p-a)(p-b)(p-c)} = \text{مساحت مثلث}$$
- برنامه‌ای که یک عدد بزرگ (حتی ۱۰۰۰ رقمی) را گرفته، تعیین می‌کند بر ۱۵ بخش پذیر است یا خیر. عددی بر ۱۵ بخش پذیر است که بر ۳ و ۵ بخش پذیر باشد. عددی بر ۳ بخش پذیر است که مجموع ارقام آن بر ۳ بخش پذیر باشد و عددی بر ۵ بخش پذیر است که رقم یکان آن صفر یا ۵ باشد (رقم یکان بر ۵ بخش پذیر است).
- برنامه‌ای که یک رقم را از ورودی خوانده، تمام اعداد صحیح بین ۱ و ۱۰۰ را چاپ می‌کند، به طوری که رقم دریافتی در اعداد ۱ تا ۱۰۰، مجذور و مکعب این اعداد وجود داشته باشد. به عنوان مثال، رقم (۱) را از ورودی بخوانیم، عدد ۱۳ یکی از اعدادی است که رقم یک در آن شرط صدق می‌کند، زیرا، در ۱۳، ۱۶۹ و ۲۱۹۷ وجود دارد (برای تشخیص وجود رقم در عدد، مجذور و مکعب عدد از یک متد استفاده کنید).
- برنامه‌ای که سه عدد را از ورودی می‌خواند و توسط متدی میانگین آن‌ها را محاسبه می‌کند و نمایش می‌دهد.
- برنامه‌ای که یک عدد را از ورودی می‌خواند و توسط یک متد بازگشتی، شمارش معکوس از آن عدد به یک را انجام دهد (شماره‌ها را در خروجی چاپ می‌کند).
- متدی که دو آرگومان را می‌پذیرد و آرگومان اول را به توان آرگومان دوم می‌رساند. آرگومان اول یک مقدار double و آرگومان دوم یک مقدار صحیح مثبت یا منفی است. برنامه‌ای که از این متد استفاده می‌کند.

## ۶۰۰ برنامه C++ با حل آن‌ها ۴۷

- برنامه‌ای که نمایش زمان را از حالت ۲۴ ساعت به ۱۲ ساعت تبدیل می‌کند. مثلاً باید ۱۴:۲۵ را به صورت ۲:۲۵pm تبدیل نماید. این برنامه طوری طراحی شده است که دارای سه متد زیر می‌باشد.
- قیمت سهام معمولاً به صورت کسری بیان می‌شود، مثلاً  $\frac{7}{8}$ , ۲ یا  $\frac{1}{2}$ , ۸. برنامه‌ای که ارزش سهام را به صورت دو عدد صحیح و کسری از کاربر می‌گیرد. قسمت کسری را نیز به صورت دو عدد صحیح (عدد اول صورت و عدد دوم مخرج) دریافت می‌نماید (بر اساس تبدیل سهام از یک متد استفاده می‌کند که سه پارامتر را گرفته، ارزش سهام را به صورت double برمی‌گرداند).
- پارکینگی برای توقف تا سه ساعت، حداقل ۲ دلار و برای هر ساعت اضافه یا بخشی از ساعت اضافه بر ۳ ساعت ۰/۵ دلار می‌گیرد. حداکثر مبلغ قابل پرداخت برای هر دوره ۲۴ ساعتی ۱۰ دلار است. فرض کنید که هیچ خودرویی بیش از ۲۴ ساعت توقف نمی‌کند. برنامه‌ای که زمان استفاده از پارکینگ را برای مشتریان دریافت می‌کند و مبلغ پرداختی مشتری، مبلغ کل پرداختی را چاپ نماید. کاربر برای هر مشتری ساعات و دقیقه توقف را وارد می‌کند. متدی مبلغ پرداختی را محاسبه و برمی‌گرداند.
- برنامه‌ای که با استفاده از چند متد اعمال زیر را انجام دهید:
  - تمام اعداد دورقمی که رقم اول و دوم آن‌ها برابر باشند را چاپ کند.
  - تمام اعداد سه‌رقمی که رقم وسط آن‌ها صفر باشد را چاپ کند.
  - تمام اعداد سه‌رقمی که مجموع رقم اول و دوم کوچک‌تر از رقم سوم باشد.
  - تمام اعداد چهاررقمی که قرینه یکدیگرند (مثلاً ۲۳۳۲). یعنی رقم اول برابر رقم چهارم و رقم دوم برابر رقم سوم باشد.
- برنامه‌ای که با استفاده از یک متد بازگشتی بزرگ‌ترین مقسوم‌علیه مشترک دو عدد را محاسبه و چاپ می‌کند.
- برنامه‌ای که اولین روز سال را می‌گیرد و تقویم سالانه را نمایش می‌دهد.
- برنامه‌ای که یک عدد زوج را گرفته، تمام زوج‌های اولی (دو عدد فرد اولی که مجموع آن‌ها برابر با آن عدد زوج باشد) را نمایش می‌دهد. به عنوان مثال، اگر کاربر عدد ۲۲ را وارد کند، خروجی به صورت زیر نمایش داده شود.
$$۲۲ = ۱۱ + ۱۱ \quad ۲۲ = ۱۷ + ۵ \quad ۲۲ = ۱۹ + ۳$$
- یونانیان باستان اعداد را به صورت هندسی دسته‌بندی می‌کردند. به عنوان مثال، آن‌ها یک عدد را مثلثی می‌نامیدند که می‌توانستند با آن تعدادی ریگ (به اندازه عدد ریگ)، در یک تقارن مثلثی بچینند. متدی به نام isTrain که یک عدد را گرفته تشخیص

می‌دهد، عدد مثلثی است یا خیر. اگر عدد مثلثی باشد، مقدار ۱، وگرنه مقدار صفر را برمی‌گرداند. دوازده عدد مثلثی در زیر آمده‌اند:

۰ ۱ ۳ ۶ ۱۰ ۱۵ ۲۱ ۲۸ ۳۶ ۴۵ ۵۵ ۶۶

➤ متدی به نام `isSquare` که تشخیص می‌دهد، یک عدد مربعی است یا خیر؟ چند عدد مربعی عبارت‌اند از:

۰ ۱ ۴ ۹ ۱۶ ۲۵ ۳۶ ۴۹ ۶۴ ۸۱ ۱۰۰ ۱۲۱ ...

➤ برنامه‌ای که `x` و `y` را خوانده، حاصل عبارت  $\sqrt{|x-y|y}$  را محاسبه کرده، نمایش می‌دهد.

➤ برنامه‌ای که نمره‌ای را به عدد بین صفر تا ۱۰۰ خوانده، توسط متدی مقدار حرفی معادل آن را برمی‌گرداند و نمایش می‌دهد (اگر نمره بین ۸۰ تا ۱۰۰ باشد، حرف `A`، اگر نمره بین ۶۰ تا ۸۰ باشد، حرف `B`، چنانچه نمره بین ۵۰ تا ۶۰ باشد، حروف `C`، وگرنه حروف `F` را نمایش می‌دهد).

➤ متد `main`: متغیر با نوع `int` به نام `grade` تعریف کرده، سپس با یک پیام مناسب نمره را از ورودی خوانده، در متغیر `grade` قرار می‌دهد، در ادامه با پیام مناسب و فراخوانی متد `selectGrade` نمره حرفی را نمایش می‌دهد.

➤ متد `selectGrade`: نمره عددی را از طریق پارامتر `grade` دریافت می‌کند و نمره حرفی را برمی‌گرداند. چون نوع متد `char` انتخاب گردید، با استفاده از `if`‌های متداخل یکی از حروف `A`، `B`، `C` یا `F` را برمی‌گرداند (با توجه به مقدار پارامتر `grade`).

➤ برنامه‌ای که سن فردی را به سال، ماه و روز دریافت کرده، توسط متدهای به‌روز، ساعت، دقیقه و ثانیه تبدیل کرده، برمی‌گرداند و سپس نمایش می‌دهد (هرسال ۳۶۵٫۲۵ روز و هرماه ۳۰ روز است).

➤ برنامه‌ای که حقوق کارمند را خوانده، توسط متدی مالیات بر حقوق را محاسبه کرده، نمایش می‌دهد (مالیات از طریق جدول زیر محاسبه می‌گردد):

## ۶۰۰ برنامه C++ با حل آن‌ها ۴۹

| از      | تا       | درصد مالیات |
|---------|----------|-------------|
| 0       | 483000   | 0           |
| 483001  | 600000   | 10%         |
| 600001  | 100000   | 15%         |
| 1000001 | 2000000  | 20%         |
| 2000001 | 99999999 | 30%         |

- برنامه‌ای که عددی را خوانده، با استفاده از متدی حاصل ضرب ارقام بالای ۵ آن را محاسبه کرده، نمایش می‌دهد.
- برنامه‌ای که عددی را خوانده، با استفاده از یک متد حاصل ضرب ارقام زوج آن‌ها را برمی‌گرداند و نمایش می‌دهد.
- برنامه‌ای که  $n$  را خوانده، با استفاده از یک متد  $n$  بار ۱۰ تا  $x$  را نمایش می‌دهد.
- برنامه‌ای که نمره پایان ترم و میان ترم دانشجویی را گرفته، با استفاده از یک متد نمره نهایی دانشجو را محاسبه و برمی‌گرداند (نمره نهایی دانشجو برابر با نمره میان ترم \* ۰,۴ + نمره پایان ترم \* ۰,۶ است).
- برنامه‌ای که استفاده از متغیرهای محلی و سراسری را نمایش می‌دهد.
- برنامه‌ای که اطلاعات جعبه‌ای از قبیل طول، عرض و ارتفاع را به‌عنوان پارامتر دریافت کرده، با استفاده از متدی حجم جعبه را محاسبه کرده، نمایش می‌دهد (هدف این برنامه، آشنایی بیش‌تر با متدها با پارامتری با مقادیر پیش‌فرض است). حجم جعبه برابر با ارتفاع  $\times$  عرض  $\times$  طول است.
- برنامه‌ای که قیمت کالا و درصد تخفیف را دریافت کرده، به متدی ارسال می‌کند و این متد میزان تخفیف کالا را برگردانده و در برنامه میزان تخفیف را نمایش می‌دهد.
- برنامه‌ای که میزان موجودی حساب بانکی را به همراه درصد بهره سالانه دریافت کرده، تعیین می‌کند سپس از چند سال موجودی حساب (بدون برداشت هیچ مبلغی از حساب) به مبلغ خاصی می‌رسد (موجودی انتهای هر سال توسط متدی محاسبه می‌گردد).
- برنامه‌ای که کاراکتری را از ورودی خوانده، توسط متدهای تعیین می‌کند که کاراکتر خوانده‌شده، رقم، حرف بزرگ و حرف کوچک است یا خیر؟
- برنامه‌ای که عدد  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد:

$$1! + 2! + \dots + n!$$

➤ این برنامه متدی برای محاسبه فاکتوریل و متد دیگری برای محاسبه مجموع دارد.

➤ برنامه‌ای که  $n$  را خوانده، حاصل عبارت مقابل را نمایش می‌دهد:

$$\frac{1}{1!} + \frac{2}{2!} + \frac{3}{3!} + \dots + \frac{n}{n!}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل عبارت مقابل را نمایش می‌دهد:

$$\frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

➤ این برنامه متدهای برای محاسبه فاکتوریل، توان ( $x^n$ ) و مجموع سری دارد.

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل جمع  $n$  جمله سری زیر را نمایش می‌دهد:

$$\frac{x}{1} + \frac{x^2}{2} + \frac{x^3}{1*3} + \dots + \frac{x^5}{1*3*5}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل  $n$  جمله سری زیر را نمایش می‌دهد:

$$\frac{x}{1} - \frac{x^2}{2} + \frac{x^3}{1*3} - \frac{x^4}{2*4} + \frac{x^5}{1*3*5} + \dots$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل  $n/2$  جمله عبارت زیر را نمایش می‌دهد:

$$-\frac{x^2}{2} + \frac{x^4}{2*4} - \frac{x^6}{2*4*6} + \dots$$

➤ برنامه‌ای که دو مقدار عددی را خوانده، سپس توسط متدی به عدد اول یک واحد اضافه نماید و از عدد دوم یک واحد کسر کرده، آن‌ها را در برنامه اصلی نمایش می‌دهد.

➤ برنامه‌ای که  $x$  و  $k$  را خوانده، لگاریتم  $x$  در مبنای  $k$  را با یک متد محاسبه کرده، نمایش می‌دهد.

➤ برنامه‌ای سه عدد را خوانده توسط متدی یکی از مقادیر زیر را برگرداند:

اگر سه عدد تشکیل مثلث را نمی‌دهند، صفر را برگرداند.

اگر سه عدد تشکیل مثلث قائم‌الزاویه را می‌دهند، عدد یک را برمی‌گرداند.

اگر سه عدد تشکیل مثلث متساوی‌الساقین را می‌دهند، عدد ۲ را برمی‌گرداند.

اگر سه عدد تشکیل مثلث متساوی‌الاضلاع را می‌دهند، عدد ۳ را برمی‌گرداند.

وگرنه عدد ۱- را برمی‌گرداند.

## ۶۰۰ برنامه C++ با حل آن‌ها ۵۱

➤ برنامه‌ای که اعداد  $n$  و  $b$  را خوانده، عدد  $n$  را به روش بازگشتی از مبنای ۱۰ به مبنای  $b$  تبدیل می‌کند (دقت داشته باشید که  $b$ ، مبنای کوچک‌تر از ۱۰ است).

➤ برنامه‌ای که با روش بازگشتی مقدار مشتق  $\frac{d(n)}{dn}(e^{ax})$   $n$ ام را محاسبه می‌کند:

توضیح: اگر از متد  $e^{ax}$  مشتق بگیریم، رابطه بازگشتی مقابل به دست می‌آید:

$$(e^{ax})^n = a(e^{ax})^{n-1}$$

➤ برنامه‌ای که  $\sin$ ،  $\cos$ ،  $\tan$  زوایای مضرب پنج از ۵ تا ۹۰ را در خروجی با فرمت جدول‌بندی مناسب تا سه رقم اعشار نمایش می‌دهد. برای محاسبه زوایا باید از درجه به رادیان تبدیل شوند. برای این منظور از فرمول زیر استفاده می‌شود:

$$\frac{R}{3.1415} = \frac{D}{180} \Rightarrow R = \frac{3.1415 \times D}{180}$$

➤ برنامه‌ای که ۱۵ جمله دنباله زیر را به کمک بازگشتی زیر نمایش می‌دهد:

$$F(1) = 2 \quad \text{اگر } n = 1$$

$$F(n) = 2 * F(n-1) + 1 \quad \text{وگرنه}$$

➤ برنامه‌ای که حاصل عبارت  $f(x) = 5x^2 - 3x + 4$  را برای مقادیر صحیح و اعشاری  $x$  با دو متد هم نام محاسبه کرده و نمایش می‌دهد.

➤ برنامه‌ای که قاعده و ارتفاع مثلث را خوانده، با استفاده دو متد هم نام مساحت مثلث را حساب می‌کند. مساحت مثلث برابر با قاعده ضرب در نصف ارتفاع است. در ضمن ارتفاع و قاعده می‌توانند دو عدد صحیح یا دو عدد اعشاری باشند.

➤ برنامه‌ای که کاراکتری را خوانده، بدون استفاده از عملگر  $+$  و  $if$  آن را به حروف کوچک تبدیل می‌کند و نمایش می‌دهد. این برنامه برای تبدیل حروف بزرگ و کوچک از یک متد استفاده می‌کند.

➤ برنامه‌ای که یک کاراکتر را خوانده، با استفاده از عملگر  $xor$  (^) اگر حرف کوچک بود، آن را به حروف بزرگ تبدیل کرده، یا اگر حرف بزرگ بود، آن را به حرف کوچک تبدیل نماید. برای این منظور، از یک متد استفاده نماید.

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل متد  $F(x, n) = x^{n!}$  را نمایش می‌دهد.

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل عبارت مقابل را نمایش می‌دهد

$$F(x, n) = \sum_{k=1}^n \frac{x^k}{k!} :$$

➤ برنامه‌ای که دو عدد صفر یا ۱ را خوانده، نتایج عملگرهای XOR، OR، AND، NOR، NAND را نمایش می‌دهد. در این برنامه متدهای برای برگشت نتایج عملگرهای XOR، OR، AND، XOR، NAND و XOR پیاده‌سازی شوند.

➤ برنامه‌ای که  $n$  را خوانده، تمام اعداد تام (کامل) کوچک‌تر یا مساوی  $n$  که مضرب ۳ هم باشند را نمایش می‌دهد. (عددی تام است که مجموع مقسوم‌های کوچک‌تر از خودش برابر خودش باشد).

➤ متد بازگشتی که دو عدد  $x, y$  را خوانده، حاصل عبارت زیر را برمی‌گرداند:

$$F(x, y) = \begin{cases} x - y & \text{اگر } x \text{ یا } y \text{ کوچک‌تر از صفر باشد} \\ F(x-1, y) + F(x, y-1) & \text{وگرنه} \end{cases}$$

برنامه‌ای که از این متد بازگشتی استفاده می‌کند.

➤ متد بازگشتی  $n$  را به‌عنوان پارامتر دریافت کرده، تمام اعداد مضرب ۳ کوچک‌تر از  $n$  را نمایش می‌دهد. برنامه که از این متد استفاده می‌کند.

➤ متد بازگشتی که عددی را به‌عنوان پارامتر دریافت می‌کند و تمام اعداد فرد بین این دو عدد را نمایش می‌دهد. برنامه‌ای که از این متد استفاده می‌کند.

➤ موسسه‌ای قصد دارد  $x$  درصد به حقوق کارمندان اضافه کند. متدی که ضریب افزایش حقوق و حقوق فعلی کارمند را دریافت کرده، حقوق جدید را برمی‌گرداند. برنامه‌ای که برای  $n$  کارمند از این متد استفاده می‌کند.

➤ متدی که دو عدد را به‌عنوان پارامتر دریافت کرده، تعیین می‌کند چند عدد اول بین این دو عدد وجود دارند. این برنامه، متدی برای تعیین عدد اول دارد. برنامه‌ای که از این متدها استفاده می‌کند. “عدد اول است که بر هیچ عدد کوچک‌تر از نصف خودش به جزء یک بخش‌پذیر نباشد.”

➤ متدی که تأثیر نیروی جاذبه بر اجسام در حال سقوط را محاسبه می‌کند. برنامه‌ای که زمان را دریافت کرده (برحسب ثانیه) و این تأثیر را محاسبه می‌نماید و چاپ می‌کند. این برنامه باید ارتفاع شیء سقوط‌کننده در هر ثانیه را نمایش می‌دهد. برای این منظور از فرمول‌های زیر استفاده می‌نماید:

$$\text{زمان } t = 9,80665, g = \text{فاصله و } \text{ارتفاع } t^2 * g * \frac{1}{2} = \text{فاصله}$$

### ۶۰۰ برنامه C++ با حل آن‌ها ۵۳

➤ برنامه‌ای که عدد  $n$  را خوانده، سپس یک عدد دیگر  $k$  را بخواند و حاصل ضرب اعداد کوچک‌تر یا مساوی عدد  $n$  که مضرب آن عدد  $k$  باشد را نمایش می‌دهد.

➤ اگر دو ضلع  $(B, C)$  از مثلث و زاویه بین این دو ضلع  $\text{Alpha}$  باشد، می‌توانیم طول ضلع سوم را با فرمول زیر محاسبه کنیم:

$$A^2 = B^2 + C^2 - 2 * B * C * \cos(\text{Alpha})$$

➤ برنامه‌ای که  $B, C$  و  $\text{Alpha}$  را خوانده، با استفاده از متدی مقدار  $A$  را محاسبه می‌کند و نمایش می‌دهد.

➤ برنامه‌ای که دو عدد، سپس یک کاراکتر را خوانده، اگر کاراکتر  $R$  یا  $r$  باشد، مساحت مستطیل را با یک متد محاسبه می‌کند. چنانچه کاربر کاراکتر  $T$  یا  $t$  را وارد نماید، مساحت مثلث را با فرمول ارتفاع \* نصف قاعده محاسبه می‌نماید. در نهایت مساحت مستطیل یا مثلث را نمایش می‌دهد.

➤ فرض کنید اساتید دارای حقوق کم‌تر از ۴۰۰۰۰۰۰ تومان ۵٪ درصد اضافه حقوق می‌گیرند، ولی اساتید بین ۴۰۰۰۰۰۰ تا ۷۰۰۰۰۰۰ تومان ۱۲ درصد و اساتید با حقوق بیش‌تر ۱۰ درصد اضافه حقوق می‌گیرند. متدی که اضافه حقوق استاد را محاسبه می‌کند. برنامه‌ای که حقوق استادی را خوانده، اضافه حقوق و حقوق جدید استاد را نمایش می‌دهد.

➤ برنامه‌ای که  $n$  و  $m$  را خوانده، توسط متد بازگشتی  $C_n^m$  را برگرداند، نمایش می‌دهد:

$$C_n^m = \begin{cases} 1 & \text{اگر } m = n \text{ یا } m = 0 \\ n & \text{اگر } m = 1 \\ \binom{n-1}{m} + \binom{n-1}{m-1} & \text{وگرنه} \end{cases}$$

➤ برنامه‌ای که عددی را خوانده با استفاده از متد بازگشتی مغلوب آن را نمایش می‌دهد. به‌عنوان مثال، اگر عدد ۳۲۵۷ باشد، مغلوب آن یعنی ۷۵۲۳ را نمایش می‌دهد.

➤ برنامه‌ای که  $n$  و  $m$  را خوانده، توسط یک متد بازگشتی  $A(m, n)$  را محاسبه می‌کند و نمایش می‌دهد  $A(m, n)$  به‌صورت زیر محاسبه می‌شود.

$$A(m, n) = \begin{cases} n + 1 & \text{اگر } m = 0 \\ A(m-1, 1) & \text{وگرنه } n = 0 \\ A(m-1, A(m, n-1)) & \text{وگرنه} \end{cases}$$

➤ برنامه‌ای که  $n$  و  $m$  را خوانده، توسط یک متد بازگشتی  $Q(m, n)$  را محاسبه می‌کند و

$$Q(m, n) = \begin{cases} 1 & \text{وگر نه } n == 1 \text{ یا } m == 1 \text{ باشد} \\ Q(m, n) & \text{اگر } m < n \text{ باشد} \\ 1 + Q(m, m-1) & \text{اگر } m == n \text{ باشد} \\ Q(m, n-1) + Q(m-n, n) & \text{اگر } m > n \text{ باشد} \end{cases}$$

نمایش می‌دهد:

➤ برنامه‌ای که مشتق متد  $1 + x^2$  را در نقطه  $x$  محاسبه می‌کند. در این برنامه به ازای  $h$  کوچک می‌توان نوشت:

$$f'(x_0) = \lim_{h \rightarrow 0} \frac{f(x_0 + h) - f(x_0)}{h}$$

➤ برنامه‌ای که مشتق دوم متد  $1 + x \sin x$  در نقطه  $x$  با فرمول زیر محاسبه می‌کند.

$$f''(x_0) = \lim_{h \rightarrow 0} \frac{f'(x_0 + h) - f'(x_0)}{h} = \frac{f'(x_0 + h) - f'(x_0)}{h}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\left(1 + \frac{x^n}{1}\right)^1 + \left(1 + \frac{x^{(n-1)}}{2}\right)^2 + \dots + \left(1 + \frac{x^1}{n}\right)^n$$

➤ برنامه‌ای که سه عدد  $x$ ،  $n$  و  $a$  را دریافت کرده، حاصل عبارت زیر را محاسبه می‌کند و نمایش می‌دهد:

$$(x+a)^n = \frac{n!}{0!(n-0)!} x^0 a^{n-0} + \frac{n!}{1!(n-1)!} x^1 a^{n-1} + \dots + \frac{n!}{n!(n-n)!} x^n a^{n-n}$$

➤ برنامه‌ای که اعداد صحیح  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\begin{aligned} & n! \frac{x^n}{1} + (n-1)! \frac{x^{(n-1)}}{1 - \frac{1}{x}} + (n+1)! \frac{x^{(n+1)}}{1 + \frac{1}{x}} \\ & + (n-2)! \frac{x^{(n-2)}}{2 - \frac{2}{x}} + (n+2)! \frac{x^{(n+2)}}{2 + \frac{2}{x}} + \dots + \\ & x^{(n-n)} \frac{(n-n)!}{n - \frac{n}{x}} + x^{(n+n)} \frac{(n+n)!}{n + \frac{n}{x}} \end{aligned}$$

➤ پروژه برنامه‌نویسی ۱. برنامه‌ای که تمامی جفت‌های اعداد متحابه از یک تا عدد خوانده‌شده را نمایش می‌دهد. دو عدد صحیح متحابه گویند، اگر مجموع مقسوم‌علیه‌های هر یک به جز خود عدد با عدد دوم برابر باشد. به‌عنوان مثال، مقسوم‌علیه‌های ۲۲۰ که از ۲۲۰ کوچک‌تر هستند، عبارت انداز: ۵۵، ۱۱۰، ۲۲، ۱۰، ۵، ۴، ۲، ۱ که مجموع

## ۶۰۰ برنامه C++ با حل آن‌ها ۵۵

آن‌ها عدد ۲۴۸ می‌شود و مقسوم‌علیه‌های ۲۴۸ که کوچک‌تر از ۲۴۸ هستند، برابر با ۱، ۲، ۴، ۷، ۱۶، ۳۱، ۶۲، ۹۶، ۱۲۴، ۱۵۲، ۱۹۶، ۲۴۸ هستند. پس این دو عدد متحابه می‌باشند.

➤ **پروژه برنامه‌نویسی ۲.** برنامه‌ای که مقداری را از یک مبنا دریافت کرده، به مبنای دیگر تبدیل می‌کند (مبناها می‌توانند بین ۲ تا ۱۶ باشند). این پروژه مبنای مبدأ، عددی در آن مبنا (چنانچه کاربر مقدار را خارج از مبنا وارد کند، جلوی او را می‌گیرد) و مبنای مقصد را از کاربر دریافت می‌کند.

➤ **پروژه برنامه‌نویسی ۳.** برنامه‌ای که ابتدا منوی زیر را نشان می‌دهد:

```
Calculate one
Calculate two
Calculate three
Exit
SELECT (۱ to ۴):
```

اگر کاربر عدد ۱ را وارد کند، حاصل عبارت زیر را محاسبه می‌کند و نمایش می‌دهد:

$$\frac{x^n (n)^5 (n)^{x+2}}{x! n! (x+n)!} \sum_{i=1}^n \frac{x^i i^x x^n}{x! n! (n+2)!}$$

اگر کاربر عدد ۲ را وارد کند، حاصل عبارت زیر را محاسبه می‌کند و نمایش می‌دهد:

$$x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots \pm \frac{x^n}{n!}$$

اگر کاربر عدد ۳ را وارد کند، حاصل عبارت زیر را محاسبه می‌کند و نمایش می‌دهد:

$$\frac{((x)^n)^3 (n)^{x+3}}{n! + (x+n+2)!} \sum_{i=2}^x \frac{x^i i^x n^x}{(x+i)! + (n+x)!} \quad i = 2, 4, \dots$$

اگر کاربر عدد ۴ را وارد کند، برنامه خاتمه یابد.

➤ **پروژه برنامه‌نویسی ۴.** فرض کنید که سیستم را برای یک بنگاه معاملات ملکی می‌نویسد. در این سیستم قرار است، منوی با ۵ گزینه نمایش داده شود تا کاربر با انتخاب یکی از گزینه‌ها شکل زمین فروخته‌شده را انتخاب کند. زمین‌ها به شکل‌های مربع، مستطیل، مثلث و دایره هستند. پس از انتخاب شکل زمین توسط کاربر، قیمت هر متر زمین و درصد حق السهم فروشنده دریافت می‌گردد و در ادامه با توجه به نوع انتخاب‌شده کاربر، اطلاعات زمین دریافت می‌گردد و مبلغی که خریدار باید پرداخت کند را نمایش می‌دهد. این برنامه تا زمانی ادامه می‌یابد که کاربر در منوی ظاهرشده گزینه ۵ را انتخاب نماید.

## فصل ۴

## آرایه‌ها و رشته‌ها

۱. برنامه‌ای که عددی صحیح را از ورودی خوانده، تمام اعداد اول قبل از آن را با استفاده از تعریف زیر تعیین کرده، به خروجی می‌برد. عددی اول است که بر هیچ عدد اول قبل از خودش قابل قسمت نباشد.

وظایف توابع

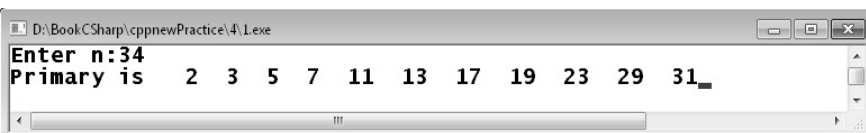
۱. تابع main: متغیر برنامه را تعریف کرده، تابع prim را فراخوانی می‌کند.

۲. تابع prim: تشخیص می‌دهد عددی اول است یا خیر؟ اگر عدد اول باشد، یک را برمی‌گرداند.

```
#include <iostream>
using namespace std;
int prim(int [],int,int);
int main()
{
    int n,p[300],i,tedad=0;
    cout << "Enter n:";
    cin >> n;
    for (i=2;i<=n;i++)
        if(prim(p,tedad,i)) tedad++;
    cout << "Primary is ";
    for (i=0; i<tedad;i++)
        cout << " " << p[i];
    return 0;
}

int prim( int a[],int tedad,int num)
{
    int i;
    for (i=0; i<tedad;i++)
        if (num % a[i] == 0)
            return (0);
    a[tedad]=num;
    return(1);
}
```

| تابع | متغیر | هدف                   |
|------|-------|-----------------------|
| main | N     | عدد خوانده شده        |
|      | p     | آرایه‌ای از اعداد اول |
|      | i     | شمارنده               |
|      | Tedad | تعداد اعداد اول       |
| prim | i     | شمارنده               |



۳. برنامه‌ای که شماره دانشجویی تعدادی از دانشجویان را از ورودی خوانده، در آرایه قرار می‌دهد. سپس عناصر آرایه را به روش انتخابی مرتب می‌کند. مرتب‌سازی آرایه به روش انتخابی به این صورت انجام می‌شود: کوچک‌ترین عنصر آرایه پیدا شده، جای آن با عنصر اول آرایه عوض می‌شود. در مرحله بعد بقیه عناصر آرایه برای یافتن کوچک‌ترین عنصر آرایه جستجو می‌شود و جای آن با عنصر دوم آرایه عوض می‌شود. این روند تا مرتب‌سازی کامل آرایه ادامه می‌یابد. پس از مرتب‌سازی نتیجه را در خروجی چاپ می‌کند. این برنامه سه تابع دارد که عبارت‌اند از: تابعی برای خواندن عناصر آرایه، تابعی برای مرتب‌سازی و تابعی برای چاپ عناصر آرایه.

## ۶۰۰ برنامه C++ با حل آن‌ها ۵۷

وظایف توابع:

۱. تابع `main`: متغیرهای برنامه را تعریف کرده، تعداد دانشجویان را می‌خواند. سپس توابع `readarray`، `select_sort` و `writearray` را فراخوانی می‌کند.
۲. تابع `readarray`: شماره‌های دانشجویی را می‌خواند و در آرایه قرار می‌دهد.
۳. تابع `select_sort`: شماره‌های دانشجویی موجود در آرایه را مرتب می‌کند.
۴. تابع `writearray`: شماره‌های دانشجویی موجود در آرایه را چاپ می‌کند.

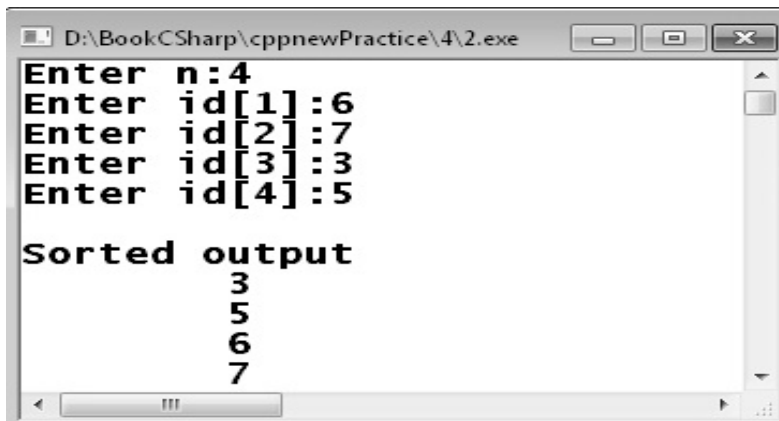
| تابع        | متغیر | هدف                                   |
|-------------|-------|---------------------------------------|
| main        | id    | آرایه‌ای برای نگهداری شماره دانشجویان |
|             | n     | تعداد دانشجویان                       |
|             | i     | شمارنده                               |
| readarray   | i     | شمارنده                               |
| select_sort | i, j  | شمارنده‌ها                            |
|             | temp  | متغیر کمکی                            |
|             | max   | بزرگ‌ترین عدد                         |
|             | i     | شمارنده                               |
| writearray  | i     | شمارنده                               |

```
#include <iostream>
using namespace std;
void readarray(int [],int);
void select_sort(int [],int);
void writearray(int[],int) ;
int main()
{
    int id[100],n;
    cout <<"Enter n:";
    cin >> n;
    readarray(id,n);
    select_sort(id,n);
    writearray(id,n);
    return 0;
}
void readarray(int id[],int n)
{
    int i;
    for(i=0;i<n;i++)
    {
        cout << "Enter id[" << i+1 << "]:";
        cin >> id[i];
    }
}
void select_sort(int id[],int n)
{
    int i,j,max,temp;
    for (i=0;i<n;i++)
    {
        max=i;
        for (j=i+1;j<n;j++)
            if(id[max]>id[j] ) max=j;
    }
}
```

```

    temp=id[i];
    id[i]=id[max];
    id[max]=temp;
}
}
void writearray(int id[],int n)
{
    int i;
    cout << endl << "Sorted output ";
    for(i=0;i<n;i++)
        cout << endl << "\t" << id[i];
}

```



۵. برنامه‌ای که تعدادی عدد را از ورودی خوانده، آن‌ها را به‌طور مرتب در آرایه‌ای قرار می‌دهد (دقت داشته باشید که اعداد در موقع گرفتن در آرایه، به‌طور صعودی مرتب شوند). سپس، آرایه مرتب‌شده، را به خروجی می‌برد.

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main()
{
    int a[100],i,j,l,n,num;
    cout << "Enter n:";
    cin >> n;
    for(i=0;i<n;i++)
    {
        cout << "Enter a[" << i+1 << "]:";
        cin >> num;
        for(j=0;j<i;j++)
            if (a[j]>num )
                break;
        for (l=i;l>j;l--) a[l]=a[l-1];
        a[j]=num;
    }
    cout << endl << "Result is : ";
    for(i=0;i<n;i++)
        cout << " " << a[i];
    return 0;
}

```

| متغیر   | هدف               |
|---------|-------------------|
| a       | آرایه‌ای از اعداد |
| n       | تعداد اعداد       |
| i, j, l | شمارنده‌ها        |
| num     | عدد خوانده‌شده    |

```

D:\BookCSharp\cppnewPractice\4\3.exe
Enter n:4
Enter a[1]:12
Enter a[2]:10
Enter a[3]:7
Enter a[4]:15

Result is :    7    10    12    15

```

۶. برنامه‌ای که دو رشته S<sub>۱</sub> و S<sub>۲</sub> را از ورودی خوانده، رشته S<sub>۱</sub> را در رشته S<sub>۲</sub> جست‌وجو می‌کند. در این برنامه، خواندن رشته‌ها توسط تابع اصلی و جست‌وجوی رشته توسط تابعی انجام می‌شود.

وظایف توابع:

۱. تابع main: دو رشته را خوانده، با فراخوانی تابع search\_string، رشته اول را در رشته دوم جست‌وجو می‌کند. سپس، پیغام مناسب چاپ می‌نماید.
۲. تابع search\_string: رشته اول را در رشته دوم جست‌وجو می‌کند. اگر رشته اول در رشته دوم باشد، مکان شروع رشته اول در رشته دوم، وگرنه صفر را برمی‌گرداند.

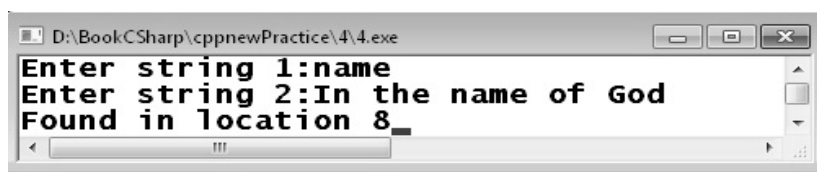
| تابع          | متغیر                           | هدف                                                 |
|---------------|---------------------------------|-----------------------------------------------------|
| main          | s <sub>۱</sub> , s <sub>۲</sub> | رشته اول                                            |
|               | location                        | S <sub>۱</sub> در رشته S <sub>۲</sub> محل وقوع رشته |
| search_string | i, j                            | شمارنده‌ها                                          |
|               | k                               | شمارنده از مکان اولین تساوی                         |

```

#include <iostream>
#include <stdio.h>
using namespace std;
int search_string(char s1[],char s2[]);
int main()
{
    char s1[255],s2[255];
    int location;
    cout << "Enter string 1:" ;
    cin.get(s1,255);
    cin.get();
    cout << "Enter string 2:" ;
    cin.get(s2,255);
    cin.get();
    location=search_string(s1,s2);
    if (location!=0)
        cout << "Found in location " << location;
    else
        cout << "Not found ";
    cin.get();
    return 0;
}
int search_string(char s1[],char s2[])
{
    int i,j,k;
    for (i=0;s2[i];i++)
    {
        k=i;
        j=0;
        while (s1[j]==s2[k] && s1[j] && s2[k])

```

```
{
    j++;
    k++;
}
if (!s1[j] )
    return (i+1);
if (!s1[j] && !s2[k] )
    return (i+1);
}
return 0;
}
```



۳. برنامه‌ای که با خواندن تعدادی عدد از ورودی، آن‌ها را در آرایه‌ای قرار می‌دهد. سپس، کلیه عناصر آرایه را بر عنصر وسط تقسیم می‌کند. اگر عنصر وسط صفر باشد، بر عنصر بعد از عنصر وسط تقسیم می‌نماید. اگر این عنصر نیز صفر باشد، بر عنصر قبل از عنصر وسط تقسیم می‌کند. اگر این عنصر صفر باشد، برای پیدا کردن عنصری غیر از صفر و انجام تقسیم، به روند قبلی ادامه می‌دهد. اگر همه عناصر آرایه صفر باشند، پیام مناسبی صادر می‌کند.

توضیح: پس از قرار دادن عناصر در آرایه، اندیس وسط را محاسبه کرده، عنصر وسط را پیدا می‌نماید. اگر صفر نبود، عمل تقسیم را انجام می‌دهد. این شمارنده به اندیس وسط اضافه می‌شود تا اگر مثبت بود، عنصر بعد از عنصر وسط و اگر منفی بود، عنصر قبل از عنصر وسط مورد بررسی قرار گیرد تا عمل تقسیم انجام شود. اگر عنصر غیر صفری در آرایه فراخوانی می‌شود تا تمام عناصر آرایه را بر آن عنصر تقسیم کند. اگر همه عناصر صفر باشند، برنامه action پیدا شود، پیام مناسبی صادر می‌کند.

| هدف                                             | متغیر | تابع   |
|-------------------------------------------------|-------|--------|
| تعداد عناصر آرایه                               | n     | main   |
| آرایه‌ای با ۱۰۰ عنصر                            | a     |        |
| عنصر وسط                                        | mid   |        |
| شمارنده                                         | i     |        |
| علامت (برای تعیین عنصر قبل و بعد به کار می‌رود) | sign  |        |
| شمارنده                                         | i     | action |

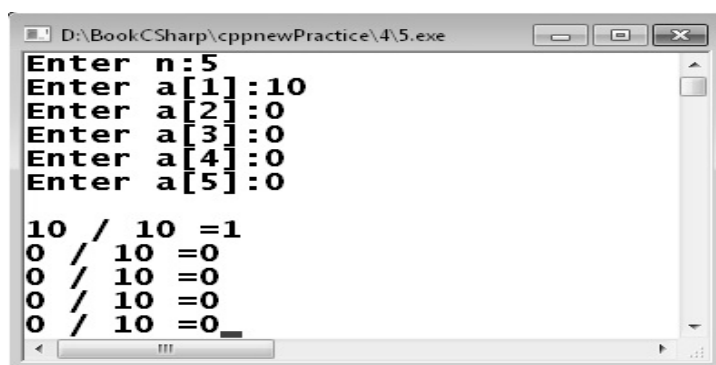
```
#include <iostream>
using namespace std;
void action(int a[],int loc,int n);
int main()
{
    int i,mid,n,a[100],sign=-1;
    cout << "Enter n:" ;
    cin >> n;
    mid=n/2;
    for(i=0;i<n;i++)
    {
        cout<<"Enter a["<<i+1<< "]:";
```

## ۶۰۰ برنامه C++ با حل آن‌ها ۶۱

```

    cin >> a[i];
}
for(i=0;i<=mid;i++)
{
    if ( a[mid+(sign*i)]!=0)
    {
        action(a,mid+(sign*i),n);
        break;
    }
    if (sign==1) i--;
    sign=-sign;
}
if(i>mid)
    cout << endl << "All element are zero";
return 0;
}
void action(int a[],int loc,int n)
{
    int i;
    for (i=0;i<n;i++)
        cout<<"\n"<< a[i]<< " / "<<a[loc]<<" =" <<(float)a[i]/a[loc];
}

```



۴. کارخانه‌ای دارای ۵ ردیف شغلی است که به کارمندان در مقابل ۴۰ ساعت کار در هفته، ماهانه مطابق جدول زیر حقوق ثابتی پرداخت می‌شود. در صورتی که کارمندی بیش از ۱۶۰ ساعت در ماه کار کند اضافه کاری به ازای هر ساعت مطابق به آنچه که در جدول مشخص شد، پرداخت می‌گردد. اگر کارمندی کمتر از ۱۶۰ ساعت در ماه کار کند، به ازای هر ساعت مبلغی مطابق جدول از حقوق وی کسر می‌شود.

برنامه‌ای که شماره کارمندی، ردیف شغلی و ساعت کارکرد در ماه برای هر کارمند را خوانده، دریافتی آن‌ها را چاپ می‌کند. برای خاتمه برنامه به جای شماره کارمندی ۹۹۹- وارد می‌شود.

| ردیف شغلی | نوع تخصص   | حقوق ثابت   | هر ساعت اضافه کاری | هر ساعت کم کاری |
|-----------|------------|-------------|--------------------|-----------------|
| ۰         | مهندس ارشد | ۱۰۰۰ و ۱۰۰۰ | ۵۰۰۰               | ۶۰۰۰            |
| ۱         | مهندس ساده | ۸۰۰ و ۱۰۰۰  | ۴۰۰۰               | ۵۰۰۰            |
| ۲         | تکنسین     | ۶۰۰ و ۱۰۰۰  | ۳۰۰۰               | ۴۰۰۰            |
| ۳         | کارگر ماهر | ۵۰۰ و ۱۰۰۰  | ۲۵۰۰               | ۳۰۰۰            |
| ۴         | کارگر ساده | ۴۶۰ و ۱۰۰۰  | ۲۰۰۰               | ۲۵۰۰            |

| متغیر | هدف                     |
|-------|-------------------------|
| table | جدولی با ۵ سطر و ۳ ستون |

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main()
{
    static long int table[5][3]
    = {{1000000,5000,6000},
      {800000,4000,5000},
      {600000,3000,4000},
      {500000,2500,3000},
      {460000,2000,2500}};
    int code,radif,hh,t,e;
    long int t_mablag;
    long int e_mablag,pay,base;
    for(;;)
    {
        t=0;
        e=0;
        cout << "Enter code,radif,hour:";
        cin >> code >> radif >> hh;
        if (code==-999) break;
        if (radif>4 || radif<0)
        {
            cout << "Enter radif between 0 to 4" << endl;
            continue;
        }
        if (hh>160)
            e=hh-160;
        else
            t=160-hh;
        base=table[radif][0];
        e_mablag=table[radif][1]*e;
        t_mablag=table[radif][2]*t;
        pay=base+e_mablag-t_mablag;
        cout << "Code=" << code << "\t hour=" << hh;
        cout << "\nbase=" << base << "\te_mablag=" << e_mablag <<
        "\tt_mablag=" << t_mablag << "\tpay=" << pay;
        cout << endl;
    }
    cin.get();
    return 0;
}
```

|                 |          |
|-----------------|----------|
| شماره کارمندی   | code     |
| ردیف شغلی       | radif    |
| ساعات کارکرد    | hh       |
| ساعات تأخیر     | t        |
| ساعات اضافه کار | e        |
| مبلغ تأخیر      | t_mablag |
| پایه            | base     |
| مبلغ اضافه کار  | e_mablag |
| قابل پرداخت     | pay      |

```
D:\BookCSharp\cppnewPractice\4\6.exe
Enter code,radif,hour:1 2 160
Code=1 hour=160
base=600000 e_mablag=0 t_mablag=0 pay=600000
Enter code,radif,hour:2 0 170
Code=2 hour=170
base=1000000 e_mablag=50000 t_mablag=0 pay=1050000
Enter code,radif,hour:3 4 150
Code=3 hour=150
base=460000 e_mablag=0 t_mablag=25000 pay=435000
Enter code,radif,hour:1 6 190
Enter radif between 0 to 4
Enter code,radif,hour:-999 0 170
```

## ۶۰۰ برنامه C++ با حل آن‌ها ۶۳

۱. برنامه‌ای که  $n$  عدد را خوانده، در آرایه‌ای قرار می‌دهد و توسط تابعی عناصر آن را معکوس (از آخرین عدد به اولین عدد) کرده، به برنامه برگرداند و برنامه آن را نمایش می‌دهد (برنامه، برای دریافت داده، معکوس کردن و چاپ عناصر آرایه از توابع جداگانه استفاده می‌کند).

وظایف توابع:

۱. تابع `main`: ابتدا تعداد عدد را می‌خواند، با فراخوانی تابع `input` تعدادی عدد را از ورودی خوانده، تابع `convert` آن‌ها را معکوس می‌کند و با فراخوانی `print` اعداد معکوس شده را چاپ می‌کند.
۲. تابع `input`: تعدادی عدد را خوانده و در آرایه قرار می‌دهد.
۳. تابع `convert`: اعداد موجود در آرایه را معکوس می‌کند.
۴. تابع `print`: عناصر موجود در آرایه را چاپ می‌کند.

```
#include <iostream>
#include <stdio.h>
using namespace std;
void input(int [],int);
void convert(int [],int);
void print(int [],int);
int main()
{
    static int num[100]={0},n;
    cout << "Enter n:";
    cin >> n;
    input(num,n);
    cout << "Input is:";
    print(num,n);
    convert(num,n);
    cout << "\nConvert is:";
    print(num,n);
    return 0;
}
void input(int a[],int n)
{
    int i;
    for (i=0;i<n;i++)
    {
        cout << "Enter number [" << i+1 << "]:";
        cin >> a[i];
    }
}
void convert(int a[],int n)
{
    int i,temp;
    for (i=0;i<n/2;i++)
    {
        temp=a[i];
        a[i]=a[n-i-1];
        a[n-i-1]=temp;
    }
}
void print(int a[],int n)
{
    int i;
    for (i=0;i<n;i++)
        cout << "    " << a[i];
```

| تابع    | نام  | هدف               |
|---------|------|-------------------|
| main    | n    | تعداد اعداد       |
|         | num  | آرایه‌ای از اعداد |
| input   | i    | شمارنده           |
| convert | temp | متغیر کمکی        |
|         | i    | شمارنده           |
| print   | i    | شمارنده           |

}

```
D:\BookCSharp\cppnewPractice\4\7.exe
Enter n:5
Enter number [1]:6
Enter number [2]:1
Enter number [3]:3
Enter number [4]:4
Enter number [5]:5
Input is: 6 1 3 4 5
Convert is: 5 4 3 1 6
```

۵. فرض کنید که در قسمت تخلفات ماشین‌ها در اداره راهنمایی رانندگی، ۱۰ نوع تخلف منظور گردید و کد تخلف از ۰ تا ۹ در نظر گرفته شد و هر تخلف جریمه خاصی دارد. برای هر ماشین اطلاعاتی مثل شماره ماشین، تعداد تخلفات و کد هر تخلف موجود است. برنامه‌ای که مبلغ جریمه را برای هر ماشین محاسبه می‌کند. برای خروج از برنامه کاربر به جای شماره ماشین ۹۹۹- را وارد می‌کند.

قراردادیم که ۱۰ سطر code توضیح: برای حل این برنامه کد تخلف به همراه مبلغ جریمه را در آرایه‌ای به نام دارد. در هر سطر مبلغ جریمه نگه‌داری می‌شود که اندیس آن کد تخلف است.

| متغیر  | هدف                   |
|--------|-----------------------|
| table  | جدول مبلغ جریمه       |
| number | شماره ماشین           |
| code   | کد تخلف               |
| sum    | مجموع تخلفات هر ماشین |
| i      | شمارنده               |
| n      | تعداد تخلف هر ماشین   |

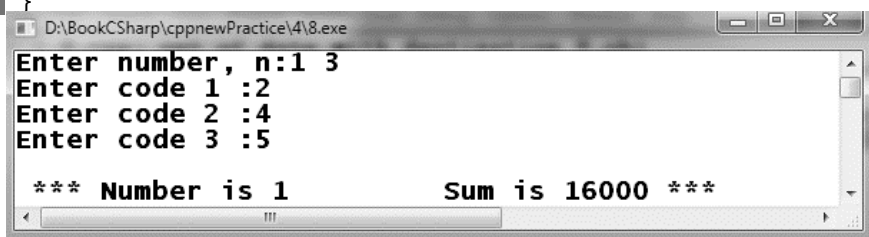
```
#include <iostream>
#include <stdio.h>
using namespace std;
int main()
{
    static long int
    table[]={1000,2000,3000,4000,5000,8000,10000,20000,30000,50000};
    int number,code,n,i;
    double sum=0.0;
    for(;;)
    {
        clrscr();
        cout << "Enter number, n:";
        cin >> number >> n;
        if (number== -999) break;
        sum=0.0;
        for (i=1;i<=n;i++)
        {
            cout << "Enter code " << i << " :";
            cin >> code;
```

## ۶۰۰ برنامه C++ با حل آن‌ها ۶۵

```

        if ( code>=0 && code<10)
            sum+=(float) table[code];
        else
        {
            i--;
            cout <<endl<< "Enter code between 0 to 9" << endl;
            continue;
        }
    }
    cout<< "\n *** Number is "<<number<<"\t Sum is "<<sum << " ***";
}
return 0;
}

```



۶. برنامه‌ای که دو عدد مبنای ۲ را خوانده، جمع کرده، به خروجی می‌برد (طول عدد مبنای دو ۳۲ رقم است).

وظایف توابع:

۱. تابع **right**: عدد دودویی دریافتی را به سمت راست انتقال می‌دهد.
۲. تابع **check**: تست می‌نماید که عدد دودویی دریافتی، درست وارد شده است یا خیر؟ چنانچه عدد دودویی درست وارد شده باشد، ۱ وگرنه ۰ را برمی‌گرداند.
۳. تابع **val**: معادل دودویی یک رشته را برمی‌گرداند.
۴. تابع **string**: معادل رشته‌ای عدد دودویی را برمی‌گرداند.
۵. تابع **addbin**: دو عدد دودویی را جمع کرده، در عدد دودویی دیگر قرار می‌دهد.

```

#include <iostream>
#include <stdio.h>
using namespace std;
#include <string.h>
int check(char []);
void right(char s[],int);
void val(char s[],int len);
void string(char s[],int len);
inline void addint(char s1[],char
s2[],char s3[],int len);
int main()
{

```

| تابع   | متغیر            | هدف                 |
|--------|------------------|---------------------|
| main   | bin <sup>۱</sup> | اولین عدد باینری    |
|        | bin <sup>۲</sup> | دومین عدد باینری    |
|        | result           | مجموع دو عدد باینری |
| check  | i                | شمارنده             |
| right  | i, j             | شمارنده‌ها          |
|        | len              | طول عدد باینری      |
| addbin | carry            | رقم نقلی            |
|        | i                | شمارنده             |
| val    | i                | شمارنده             |
| string | i                | شمارنده             |

```

char a1[33]='\0'},a2[33]='\0'},result[34]={0};
while(1)
{
    cout << "Enter a1:";
    cin.get(a1,32);
    cin.get();
    if (check(a1) ==0)
    {
        cout <<"Enter only 0 & 1";
    }
    else break;
}
while(1)
{
    cout << endl << "Enter a2:";
    cin.get(a2,32);
    cin.get();
    if (check(a2)==0)
    {
        cout << "Enter only 0 & 1";
    }
    else break;
}
right(a1,32);
right(a2,32);
val(a1,32);
val(a2,32);
addint(a1,a2,result,32);
string(a1,32);
string(a2,32);
string(result,33);
cout << endl << " " << a1;
cout << endl << " " << a2;
cout << endl;
for (int i=0;i<33 ;i++ ) cout << "=";
    cout << endl << result;
return 0;
}
check(char s[])
{
    int i;
    for (i=0; i<strlen(s);i++)
        if (!(s[i] >='0' && s[i] <='1') ) return 0;
    return 1;
}
void right(char s[],int k)
{
    int i,len,j;
    len=strlen(s);
    j=0;
    if (len-1<k)
    {
        for (i=len-1; i>=0;i--)
        {
            s[k-1-j]=s[i];
            j++;
        }
        s[k]='\0';
        for(i=0;i<k-len;i++)

```

## ۶۰۰ برنامه C++ با حل آن‌ها ۶۷

```

        s[i]='0';
    }
}
void addint(char s1[],char s2[],char s3[],int len)
{
    int i,carry=0;
    for (i=0;i<len;i++) s3[i]='0';
    s3[len+1]='\0';
    for(i=len-1;i>=0;i--)
    {
        s3[i+1]=(s2[i]+s1[i]+carry) % 2;
        carry=(s2[i]+s1[i]+carry) / 2;
    }
    s3[0]=carry;
}
void val(char s[],int len)
{
    int i;
    for (i=0;i<len;i++) s[i]-=48;
}
void string(char s[],int len)
{
    int i;
    for (i=0;i<len;i++) s[i]+=48;
}

```

۶. برنامه‌ای که عناصر آرایه‌ای ۴×۴ را خوانده، بزرگ‌ترین عنصر هر سطر را پیدا کرده، به همراه شماره سطر در خروجی چاپ می‌کند.

داری بزرگ‌ترین عنصر هر سطر در نظر گرفته شده است. توضیح: ستون پنجم آرایه برای نگه

```

#include <iostream>
using namespace std;
#include <string.h>
void readarray(int a[4][5]);
void writearray(int a[4][5]);
void findmaxrow(int a[4][5]);
int main()
{
    int a[4][5]={0};
    readarray(a);
    findmaxrow(a);
    cout << "\t*****result*****";
    writearray(a);
    return 0;
}
void readarray(int a[4][5])
{

```

| تابع       | متغیر | هدف                        |
|------------|-------|----------------------------|
| main       | a     | آرایه‌ای با ۴ سطر و ۵ ستون |
| readarray  | i, j  | شمارنده‌ها                 |
| findmaxrow | i, j  | شمارنده‌ها                 |
| writearray | i, j  | شمارنده‌ها                 |

```
int i,j;
for (i=0; i<4;i++)
{
    cout << "Enter row(" << i+1 << "):";
    for (j=0; j<4;j++)
        cin >> a[i][j];
}
}
void findmaxrow(int a[4][5])
{
    int i,j;
    for (i=0; i<4;i++)
    {
        a[i][4]=a[i][0];
        for (j=1; j<4;j++)
            if( a[i][4]<a[i][j]) a[i][4]=a[i][j];
    }
}
void writearray(int a[4][5])
{
    int i,j;
    for (i=0; i<4;i++)
    {
        cout << endl;
        for (j=0; j<4;j++)
            cout << " " << a[i][j];
        cout << "\t" << a[i][4];
    }
}
```

The screenshot shows a console window titled "D:\BookCSharp\cppnewPractice\4\10.exe". It displays the following input and output:

```
Enter row(1):1 3 4 5
Enter row(2):4 7 6 5
Enter row(3):3 2 1 4
Enter row(4):3 5 6 8
*****result*****
1 3 4 5 5
4 7 6 5 7
3 2 1 4 4
3 5 6 8 8
```

۷. برنامه‌ای که رشته‌ای را در رشته دیگر کپی می‌کند (بدون استفاده از تابع strcpy).

#### وظایف توابع:

۱. تابع main: دو رشته را خوانده، با فراخوانی strcpy رشته دوم را در رشته اول کپی می‌کند. سپس رشته اول و دوم را با پیغام مناسب چاپ می‌کند.
۲. تابع strcpy: یک رشته را در رشته دیگر کپی می‌کند.

```
#include <iostream>
#include <stdio.h>
using namespace std;
void strcpy(char [],char []);
int main()
{
    char str1[255]={'\0'},str2[255]= {'\0'};
    cout << "Enter a string:";
    cin.get(str1, 255);
```

| تابع   | متغیر            | هدف       |
|--------|------------------|-----------|
| main   | str <sup>۱</sup> | رشته مبدأ |
|        | str <sup>۲</sup> | رشته مقصد |
| strcpy | i                | شمارنده   |

## ۶۰۰ برنامه C++ با حل آن‌ها ۶۹

```
cin.get();
strcpy(str2, str1);
cout << "Result is " << str2;
return 0;
}
void strcpy(char des[], char sou[])
{
    int i;
    for (i=0; sou[i]; i++)
        des[i]=sou[i];
    des[i]='\0';
}
```

➤ برنامه‌ای که  $n$  عدد را خوانده، در آرایه‌ای قرار می‌دهد و توسط متدی عناصر آن را معکوس (از آخرین عدد به اولین عدد) کرده، به برنامه برگرداند و برنامه آن را نمایش می‌دهد (برنامه، برای دریافت داده، معکوس کردن و چاپ عناصر آرایه از متدها جداگانه استفاده می‌کند).

➤ فرض کنید که در قسمت تخلفات ماشین‌ها در اداره راهنمایی رانندگی، ۱۰ نوع تخلف منظور گردید و کد تخلف از ۰ تا ۹ در نظر گرفته شد و هر تخلف جریمه خاصی دارد. برای هر ماشین اطلاعاتی مثل شماره ماشین، تعداد تخلفات و کد هر تخلف موجود است. برنامه‌ای که مبلغ جریمه را برای هر ماشین محاسبه می‌کند. برای خروج از برنامه کاربر به جای شماره ماشین ۹۹۹- را وارد می‌کند.

توضیح: برای حل این برنامه کد تخلف به همراه مبلغ جریمه را در آرایه‌ای به نام `code` قراردادیم که ۱۰ سطر دارد. در هر سطر مبلغ جریمه و کد تخلف نگه‌داری می‌شود.

➤ برنامه‌ای که عناصر آرایه‌ای  $4 \times 4$  را خوانده، بزرگ‌ترین عنصر هر سطر را پیدا کرده، به همراه شماره سطر در خروجی چاپ می‌کند.

توضیح: ستون پنجم آرایه برای نگه‌داری بزرگ‌ترین عنصر هر سطر در نظر گرفته شده است.

➤ برنامه‌ای که رشته‌ای را در رشته دیگری می‌کند (بدون استفاده از متد `strcpy`).

➤ ماتریس جادویی یک ماتریس  $N \times N$  است که هر عنصر آن یک عدد صحیح ۱ تا  $N^2$  است. حاصل جمع هر سطر، هر ستون و همچنین عناصر قطر همگی با یک دیگر مساوی‌اند. برای تولید این ماتریس به صورت زیر عمل می‌شود:

الف. عدد یک در ستون میانی اولین سطر قرار می‌گیرد.

ب. به طور مورب به سمت چپ و بالا حرکت می‌کنیم.

ج: اگر این خانه پر باشد، یک خانه به طرف پائین حرکت می کنیم و عدد بعدی را در آن مکان می نویسیم.  
د. اگر در حرکت به صورت مورب، از محدوده ماتریس خارج شویم، باید به آخرین عنصر در آن سطر یا ستون برگردیم.  
هـ. اگر در محدوده سطر و ستون از هر دو خارج شویم، یک خانه به سمت پائین حرکت می کنیم.  
برنامه ای که ماتریس جادویی را تولید کرده و چاپ می نماید (ماتریس  $5 \times 5$  زیر را در نظر بگیرید):

|  |  |  |  |  |
|--|--|--|--|--|
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

- برنامه ای که رشته ای را به انتهای رشته دیگر الحاق می کند (بدون استفاده از متد Strcat).
- برنامه ای که رشته ای را از ورودی خوانده، به متدی ارسال می کند و متد آن را به طور معکوس به خروجی می برد.
- برنامه ای که رشته عددی را که حاوی نقطه اعشار است از ورودی خوانده، آن را به عدد اعشاری تبدیل کند. به عنوان مثال، رشته "۱۲,۳۴۲" را به ۱۲,۳۴۲ تبدیل نماید. این برنامه، متدی برای خواندن رشته، متدی برای تبدیل و متدی برای نوشتن عدد در خروجی دارد. پارامترها از طریق فراخوانی با ارجاع به متدها ارسال می شوند.
- برنامه ای که بازی دوز را شبیه سازی می کند. در این بازی یک ماتریس  $3 \times 3$  داریم که دو بازیکن ۱ و ۲ باهم بازی می کنند که هر یک از این بازیکنان سه مهره دارند (هر سه مهره شماره همان بازیکن را دارد). بازیکنی برنده است که بتواند سریع تر مهره های خود را به صورت سطری یا ستونی و یا قطری ردیف کند. به عنوان مثال، در ماتریس زیر بازیکن شماره ۲ برنده است:

## ۶۰۰ برنامه C++ با حل آن‌ها ۷۱

➤ هر عدد زوج بزرگ‌تر از ۶ برابر با مجموع دو عدد اول کوچک‌تر از خودش است. برنامه‌ای که چند عدد زوج بزرگ‌تر از ۶ را می‌خواند. مجموع دو عدد اولی که برابر با اعداد خوانده شده باشد را پیدا کرده چاپ می‌نماید (برنامه تا زمانی که کاربر بخواهد ادامه می‌یابد).

➤ برنامه‌ای که رشته عددی را که حاوی نقطه اعشار است از ورودی خوانده، آن را به صورتی که گفته شد به عدد اعشاری تبدیل می‌کند. به عنوان مثال، رشته '۱۲۴۵,۴۵۲' را به عدد ۱۲۴۵,۴۵۲ تبدیل می‌کند. برنامه، متدی برای خواندن رشته، متدی برای تبدیل و متدی برای نوشتن عدد در خروجی دارد. پارامترها از طریق فراخوانی با ارجاع به متدها ارسال می‌شوند.

➤ متدی که طول رشته را برمی‌گرداند (با استفاده از اشاره گر). سپس، برنامه‌ای که از این متد استفاده می‌کند. برنامه، متدی برای خواندن رشته، متدی برای محاسبه طول رشته و متدی برای چاپ طول رشته دارد.

➤ متدی که توان  $n$  یک عدد را برمی‌گرداند. سپس، برنامه‌ای که از این متد استفاده می‌کند.

➤ برنامه‌ای که سه رشته را خوانده، رشته دوم و سوم را به انتهای رشته اول متصل می‌کند.

➤ برنامه‌ای که رشته‌ای را خوانده، اعمال زیر را انجام می‌دهد:  
الف. کلیه حروف بزرگ رشته را به حروف کوچک تبدیل می‌کند.  
ب. کلیه ارقام رشته را شمارش می‌کند و مجموع ارقام موجود در رشته را چاپ می‌کند.  
ج: تعداد حروف رشته را شمارش می‌کند.  
برنامه، برای هر یک از این درخواست‌ها متدی دارد و کلیه پارامترهای متد را با ارجاع ارسال می‌نماید.

➤ برنامه‌ای که میزان میانگین بارش باران را در سال قبل می‌گیرد و سپس میزان واقعی بارش باران را در ۱۲ ماه گرفته، در آرایه‌ای قرار می‌دهد (برای خواندن میزان واقعی بارش باران در ۱۲ ماه از یک متد استفاده می‌کند). سپس، با یک متد دیگر اختلاف بارش باران در هر ماه را با میانگین بارش باران در سال قبل محاسبه کرده، نمایش می‌دهد.

➤ برنامه‌ای که ۵ عدد را از ورودی خوانده، در آرایه‌ای به نام S قرار می‌دهد. ابتدا میانگین مقادیر را محاسبه می‌کند (با یک متد). سپس، با استفاده از فرمول زیر انحراف

معیار را محاسبه کرده، چاپ می نماید (برای محاسبه انحراف معیار نیز از یک متد استفاده

$$\text{انحراف معیار} = \frac{\sqrt{\sum_{i=1}^N (s_i - a)^2}}{N}$$

می کند):

a میانگین اعداد و N تعداد اعداد است.

➤ برنامه ای که رشته ای را از ورودی می خواند. تمام کلمات چهار حرفی آن را با کلمه "Love" جایگزین می کند. مثلاً رشته "hate you, you doer" به رشته "love you, you love" تبدیل می گردد.

➤ در یک دوره مسابقات ورزشی تعدادی تیم (حداکثر ۱۰۰ تیم) شرکت کرده اند، می خواهیم در انتهای مسابقه گزارشی از مجموع امتیازات هر تیم را چاپ کنیم. برنامه ای که برای هر مسابقه در یک خط ورودی شامل آیتیم های برنده و امتیاز کسب شده را می خواند و نتیجه را نمایش می دهد. اگر به جای تیم برنده ۱- وارد شد، برنامه خاتمه می یابد.

➤ برنامه ای که یک رشته را خوانده، تعداد تکرار هر حرف را نمایش می دهد. به عنوان مثال، ACCURENCE را در نظر بگیرید که حرف C سه بار تکرار شده است.

➤ برنامه ای که یک رشته را خوانده تشخیص می دهد آیا متجانس است یا خیر؟ رشته ای متجانس است که از دو طرف یکی خوانده شود.

➤ برنامه ای که حداکثر ۱۰ عدد را از ورودی خوانده، می گوید هر عدد چند بار تکرار شده است. به عنوان مثال، اگر ورودی به صورت زیر باشد:

۷۰ ۱۴۳ ۱۰۰ ۷۲ ۱۴۳ ۵۲ ۱۰۰ ۱۴۳ ۱۰۰ ۷۰

خروجی زیر را چاپ می کند:

۱۴۳ OCCURS ۳ TIMES, AT POSITIONS ۳ ۶ ۹

۷۰ OCCURS ۲ TIMES, AT POSITIONS ۱ ۱۰

➤ برنامه ای ماتریس  $A[3 \times 4]$  را از ورودی می خواند و ترانواده آن را در B ذخیره می کند. در ترانواده باید برای هر  $i, j$  رابطه مقابل برقرار باشد:

$$B_{i,j} = A_{j,i}$$

➤ برنامه ای که انداختن دو تاس را شبیه سازی می نماید. این برنامه برای انداختن تاس ها از متد rand استفاده می کند که عدد بین ۱ تا ۶ را تولید می نماید. سپس مجموع این دو مقدار محاسبه می گردد. از آنجایی که تاس ها مقادیر بین ۱ تا ۶ را نشان می دهند، پس مجموع این دو مقدار بین ۲ تا ۱۲ است که مجموع ۷، بیش ترین دفعات و مجموع ۱۲ کم ترین دفعات پیش می آیند. جدول زیر ترکیبات ممکن برای این دو تاس را نشان می دهد. این برنامه، ۳۶۰۰۰ بار تاس می ریزد و با استفاده از یک آرایه، تک بعدی تعداد

## ۶۰۰ برنامه C++ با حل آن‌ها ۷۳

دفعاتی را که هر یک از این مجموعه‌های ممکن پیش می‌آیند، را ثبت می‌کند و سپس تحقیق می‌کند که آیا تعداد مجموعه‌های حاصل معقول است (مثلاً به شش طریق، مجموع ۷ حاصل می‌شود. بنابراین تقریباً  $1/6$  همه پیشامدها باید ۷ باشد).

|   | ۱ | ۲ | ۳ | ۴  | ۵  | ۶  |
|---|---|---|---|----|----|----|
| ۱ | ۲ | ۳ | ۴ | ۵  | ۶  | ۷  |
| ۲ | ۳ | ۴ | ۵ | ۶  | ۷  | ۸  |
| ۳ | ۴ | ۵ | ۶ | ۷  | ۸  | ۹  |
| ۴ | ۵ | ۶ | ۷ | ۸  | ۹  | ۱۰ |
| ۵ | ۶ | ۷ | ۸ | ۹  | ۱۰ | ۱۱ |
| ۶ | ۷ | ۸ | ۹ | ۱۰ | ۱۱ | ۱۲ |

➤ در هر سطر ورودی برای هر کالا دو عدد وجود دارد. عدد اول شماره کالا و عدد دوم موجودی آن در انبار است.

الف. برنامه‌ای که این اطلاعات را می‌خواند و در یک آرایه ذخیره می‌کند. انتهای اطلاعات با دو عدد ۱- مشخص شده است.

ب: پس از ورود اطلاعات اولیه تعدادی ورودی دیگر وجود دارد. این ورودی‌ها عبارت‌اند از: شماره کالا، تعداد و کاراکتری که مشخص‌کننده این است که تعداد ورودی سفارش (کاراکتر s یا S) یا خرید (کاراکتر p یا P) است. برنامه‌ای که اطلاعات را خوانده، مقدار خروجی انبار هر کالا را نمایش می‌دهد. اگر سفارش بیش از موجودی انبار باشد، آن را انجام نمی‌دهد و پیام مناسبی چاپ می‌کند.

➤ برنامه‌ای که تعدادی کلمه را از ورودی خوانده، تعداد کلمات ۱ تا ۵ کاراکتری، ۶ تا ۱۲ کاراکتری و بیش‌تر از ۱۲ کاراکتر را نمایش می‌دهد (تعداد کلمات را از ورودی می‌خواند).

➤ برنامه‌ای که چندجمله‌ای زیر را در یک آرایه نمایش می‌دهد (سپس، x را می‌خواند) و عبارت p را محاسبه کرده، چاپ می‌نماید (n حداکثر ۱۰۰ باشد).

$$p = anx^n + an - 1x^{n-1} + \dots + a_1x^1 + a.$$

➤ برنامه‌ای که یک رشته حداکثر ۱۰۰ کلمه‌ای را خوانده، تمام کلمات آن را وارونه می‌کند و چاپ می‌نماید (بین کلمات کاراکتر blank (فاصله) قرار می‌گیرد). به عنوان مثال، Ali and Reza به صورت azeR dna ila تبدیل می‌شود.

➤ برنامه‌ای که یک متن را از ورودی خوانده، اعمال زیر را انجام می‌دهد:

الف: درصد تکرار هر حرف را در متن پیدا می کند (حروف بین A تا Z و حروف کوچک را بزرگ در نظر بگیرد).  
 ب: درصد تکرار هر کلمه را در متن پیدا می کند (کلمات بافاصله از یکدیگر جدا می شوند (حداکثر ۱۰۰ کلمه داریم).

➤ برنامه ای که تعدادی عدد را خوانده، تشخیص می دهد مرتب شده هستند یا خیر؟

➤ برنامه ای که n عدد را خوانده، در آرایه قرار می دهد. سپس، توسط متدی عنصر n را حذف می کند.

➤ برنامه ای که دو آرایه m عنصری و n عنصری را خوانده، n عنصر اول آرایه دوم را به انتهای آرایه اول اضافه می کند.

➤ برنامه ای که عناصر یک آرایه و مقدار x را خوانده، تعداد تکرار عنصر x در آرایه a را می شمارد. در این برنامه متدها زیر نوشته شده است:  
 متدی برای خواندن عناصر آرایه a  
 متدی برای شمارش تعداد تکرار x در آرایه a

➤ برنامه ای که دو آرایه n عنصری یک بعدی را خوانده، حاصل ضرب بیرونی n عنصر اول آرایه a با n عنصر اول آرایه b را برمی گرداند (حاصل ضرب این دو آرایه، در آرایه دیگری به نام c قرار می گیرد). به عنوان مثال، فرض کنید آرایه های زیر را داشته باشیم.

| a   | b    |     |      |     |
|-----|------|-----|------|-----|
| 2.2 | 2.0  | 4.4 | -2.2 | 0.0 |
| 3.3 | -1.0 | 6.6 | -3.3 | 0.0 |
| 4.4 | 0.0  | 8.8 | -4.4 | 0.0 |

متدی برای ضرب بیرونی (outer product) داشته باشید.

$$C[i][j] = a[i] \times b[j]$$

متدی برای چاپ حاصل ضرب داشته باشید.

➤ برنامه ای که عناصر یک آرایه دوبعدی را ۹۰ درجه چرخش می دهد. به عنوان مثال، اگر آرایه زیر را داشته باشیم:

|    |    |    |
|----|----|----|
| ۱۱ | ۲۲ | ۳۳ |
| ۴۴ | ۵۵ | ۶۶ |
| ۷۷ | ۸۸ | ۹۹ |

## ۶۰۰ برنامه C++ با حل آن‌ها ۷۵

با چرخش ۹۰ درجه آرایه به صورت زیر تبدیل می‌شود:

|    |    |    |
|----|----|----|
| ۷۷ | ۴۴ | ۱۱ |
| ۸۸ | ۵۵ | ۲۲ |
| ۹۹ | ۶۶ | ۳۳ |

➤ برنامه‌ای که آرایه‌ای ۱۰ عنصری تعریف کرده، سپس ۱۰ عدد را خوانده و تعداد تکرار هر عدد را می‌شمارد.

➤ برنامه‌ای که ۵ عدد را خوانده، در آرایه‌ای قرار می‌دهد. سپس یک عدد دیگر خوانده، تعداد تکرار این عدد و مکان‌های وقوع آن را نمایش می‌دهد.

➤ برنامه‌ای که ۱۰ عدد را خوانده، در آرایه‌ای قرار می‌دهد. سپس یک عدد دیگر را می‌خواند و چنانچه آن عدد در آرایه وجود داشته باشد، آن عدد را از آرایه حذف کرده، آرایه را نمایش می‌دهد (دقت کنید که اگر عدد چند بار در آرایه تکرار شده باشد، همه تکرارهای آن عدد را از آرایه حذف می‌کند).

➤ برنامه‌ای که یک آرایه  $4 \times 4$  تعریف کرده، سپس عناصر آرایه را خوانده و حاصل ضرب عناصر غیر صفر محیطی آرایه را محاسبه می‌کند و نمایش می‌دهد. عناصر محیطی آرایه، عناصری هستند که در سطر اول، سطر آخر، ستون اول و ستون آخر آرایه می‌باشند.

➤ برنامه‌ای که یک آرایه ۵ در ۶ تعریف کرده، عناصر آن را از ورودی می‌خواند و سپس حاصل ضرب عناصر غیر محیطی که صفر نیستند را محاسبه می‌کند و نمایش می‌دهد. عناصر غیر محیطی عناصری هستند که در سطر اول، سطر آخر، ستون اول و ستون آخر نباشند.

➤ برنامه‌ای که یک ماتریس  $4 \times 4$  را می‌خواند. اگر مجموع عناصر روی قطر اصلی و فرعی برابر مجموع عناصر محیطی ماتریس باشد، برنامه "Yes"، وگرنه "No" را نمایش می‌دهد. عناصری روی قطر اصلی قرار دارند، که  $i == j$  باشد (یعنی، شماره سطر برابر شماره ستون باشد) و عناصری روی قطر فرعی قرار دارند که  $i + j == n - 1$  باشد (یعنی، مجموع شماره سطر و شماره ستون برابر با تعداد سطرها منتهی یک باشد). اما، عناصر سطر اول، سطر آخر، ستون اول و ستون آخر عناصر محیطی نام دارند.

➤ برنامه‌ای که یک ماتریس  $4 \times 4$  را خوانده، اگر حداقل یک عنصر ماتریس تکراری باشد (بیش از یک بار تکرار شده باشد)، عبارت "Yes" وگرنه، عبارت "No" را نمایش می‌دهد.

➤ برنامه‌ای که یک ماتریس  $4 \times 4$  را خوانده، اگر همه عناصر آرایه برابر باشند، "Yes"، وگرنه "No" را نمایش می‌دهد.

- برنامه‌ای که یک ماتریس  $4 \times 4$  را خوانده، اگر تمام عناصر غیر از عناصر محیطی آن باهم برابر باشند، "Yes"، وگرنه "No" را نمایش می‌دهد.
- برنامه‌ای که عناصر آرایه  $4 \times 4$  را خوانده، اگر عناصر هر سطر به صورت صعودی مرتب باشند، "Yes"، وگرنه "No" را برمی‌گرداند.
- برنامه‌ای که یک آرایه  $4 \times 4$  را می‌خواند. اگر تمام عناصر آن به صورت صعودی مرتب باشند، عبارت "Yes"، وگرنه "No" را نمایش می‌دهد.
- برنامه‌ای که یک ماتریس  $4 \times 4$  را خوانده، مجموع عناصر قطر اصلی و فرعی را نمایش می‌دهد. (در عناصر قطر اصلی آرایه  $z == i$  است و اگر عنصری روی قطر فرعی باشد،  $i = n - z + 1$  است.  $i$  شماره سطر و  $z$  شماره ستون آرایه دوبعدی است).
- برنامه‌ای که عناصر یک ماتریس  $5 \times 5$  را خوانده، بزرگ‌ترین عنصر محیطی آن را برمی‌گرداند.
- برنامه‌ای که یک رشته و عدد  $k$  را خوانده، تمام حروف الفبای رشته را به صورت زیر رمز کرده، نمایش می‌دهد.  

$$\%26 : (k + \text{حرف اصلی}) = \text{حرف رمز شده}$$

$$\%, \text{باقی مانده تقسیم صحیح به } 26 \text{ است.}$$
- برنامه‌ای که متن رمز شده و عدد  $k$  را خوانده، متن ساده را با روش زیر به دست می‌آورد و نمایش می‌دهد (یعنی، تمام حروف الفبایی را به صورت زیر رمزگشایی می‌کند):  

$$\%26 : (k - \text{حرف رمز شده}) = \text{حرف ساده}$$
- برنامه‌ای که تعدادی عدد را خوانده، در آرایه‌ای قرار می‌دهد. سپس مجموع اعداد اول در این اعداد را محاسبه می‌کند و در پایان، اعدادی که بزرگ‌تر از مجموع اعداد اول هستند را چاپ می‌کند.
- برنامه‌ای که ۱۰ عدد تصادفی بین ۱ تا ۴۰ تولید کرده، در آرایه اول قرار می‌دهد. سپس، ۱۰ عدد تصادفی دیگر بین ۱ تا ۵۰ تولید نموده، در آرایه دوم قرار می‌دهد. در پایان، عناصری که در آرایه اول وجود دارند، ولی در آرایه دوم نیستند را نمایش می‌دهد. یعنی تفاضل دو آرایه را نمایش می‌دهد.
- برنامه‌ای که ۱۰ عدد را خوانده، در آرایه‌ای قرار می‌دهد. سپس یک عدد بین ۰ تا ۸ گرفته، عناصر آرایه را به تعداد عدد خوانده شده به سمت چپ چرخش می‌دهد و چاپ می‌کند.

➤ برنامه‌ای که یک رشته را خوانده، تمام کلمات دوحرفی رشته را حذف کرده، آن را نمایش می‌دهد.

➤ برنامه‌ای که یک رشته را خوانده، تعداد کلمات یک حرفی، دوحرفی، سه حرفی، چهار حرفی و غیره را می‌شمارد.

➤ برنامه‌ای که ۱۰ عدد را خوانده، در آرایه‌ای قرار دهد. سپس تعیین می‌کند چه عددی بیش از همه تکرار شده است.

➤ برنامه‌ای که یک رشته را خوانده، کلیه جاهای خالی از قبیل کاراکتر blank، '\n' و '\t' رشته را حذف می‌کند و نمایش می‌دهد.

➤ یک عدد صحیح اول است، وقتی که بر خودش و یک بخش پذیر باشد. غربال اراتوتسن روشی را برای تعیین عدد اول دارد. در این روش آرایه‌ای در نظر می‌گیرد و اعمال زیر را بر روی آن انجام می‌دهد تا اعداد اول را تعیین کند:

۱. مقدار اولیه همه عناصر آرایه را برابر یک قرار می‌دهد.

۲. با شروع از اندیس ۲ (اندیس یک حتماً اول است). هر بار که عنصری با مقدار یک یافت می‌شود، بقیه عناصری که اندیششان مضربی از ۲ باشد (اندیس‌های ۴، ۶، ۸، ۱۰ و ...) را برابر صفر قرار می‌دهد. برای اندیس ۳، همه عناصر بعد از ۳ که اندیششان مضربی از ۳ است (اندیس‌های ۶، ۹، ۱۲، ۱۵ و ...) صفر قرار می‌گیرد و به همین ترتیب این روند ادامه می‌یابد.

۳. در پایان عناصری که هنوز یک هستند، شماره اندیس‌هایشان اول‌اند. برنامه‌ای که با استفاده از یک آرایه ۱۰۰۰ تایی، اعداد اول ۱ تا ۹۹۹ را به دست آورده، نمایش می‌دهد.

➤ برنامه‌ای که آرایه‌ای ۱۰ عنصری با مقادیر اولیه را تعریف کرده و به روش بازگشتی مجموع عناصر آرایه را محاسبه و نمایش می‌دهد.

➤ برنامه‌ای که مقادیر آرایه ۱۰ عنصری که مقدار اولیه دارد، m واحد به سمت راست شیفت می‌دهد.

➤ متد بازگشتی که ترکیبات یک مجموعه را برمی‌گرداند و نمایش می‌دهد. این متد دارای سه پارامتر است که عبارت‌اند از: شروع آرایه (مجموعه)، تعداد عناصر مجموعه و ترکیبات مجموعه. سپس، برنامه‌ای که تعداد ترکیبات مجموعه را نمایش می‌دهد. به عنوان مثال، مجموعه {A, B, C} را در نظر بگیرید. برای این مجموعه حالت‌های زیر را داریم:

{AAA, AAB, AAC, ABA, ABB, ABC, ACB, ACC, ...}

➤ همان طور که می دانید فاکتوریل اعداد بزرگ در یک double جا نمی شود. برنامه ای که فاکتوریل اعداد بزرگ را به کمک آرایه محاسبه می کند و نمایش می دهد.

➤ برنامه ای که دو آرایه  $3 \times 4$  به نام های x و y را خوانده، حاصل عبارت زیر را در آرایه دیگر به نام z قرار می دهد:

$$Z = 2x - 3y$$

➤ برنامه ای که بین تعدادی عدد که در آرایه قرار دارند، بزرگ ترین، کوچک ترین مقدار و مکان های آن ها را پیدا کرده، نمایش می دهد.

➤ برنامه ای که ۲۰ عدد تصادفی (دو سری اعداد تصادفی ۱۰ تایی) ایجاد کرده، در دو آرایه قرار می دهد. سپس یک عدد را خوانده، تمام عناصر آرایه دوم را در آرایه اول از مکان وارد شده درج می نماید.

➤ برنامه ای که یک رشته را خوانده، سپس دو عدد را می خواند و از کاراکترهای عدد اول وارد شده تا عدد دوم وارد شده رشته ورودی را حذف می کند (یعنی، بخشی از رشته را حذف می نماید).

➤ برنامه ای که تعداد اعداد تصادفی تولید کرده، در آرایه  $4 \times 4$  قرار می دهد. سپس هر سطر آرایه را جداگانه مرتب می نماید. در پایان، آرایه ای که عناصر سطرهای آن مرتب شده اند را نمایش می دهد.

➤ برنامه ای که آرایه ۵ عنصری را خوانده، به مقدار هر یک از عناصر آرایه توسط علامت ستار (\*) میله افقی رسم می کند.

➤ برنامه ای که ۷ عدد را خوانده، سپس دو عدد دیگر را می خواند و از این دو عدد خوانده شده، عدد دوم را جایگزین عدد اول در آن هفت عدد می نماید.

➤ برنامه ای که عدد صحیح n (کوچک تر از ۹) را از ورودی خوانده، خروجی زیر را با استفاده از آرایه دوبعدی (ماتریس) تولید و چاپ می کند:

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 1   | 2   | 3   | ... | n   |
| 2   | 3   | 4   | ... | n+1 |
| ... | ... | ... | ... | ... |
| n+1 | n+2 | n+3 | ... | 2n  |

➤ برنامه ای که رشته ای را خوانده، با استفاده از یک متد فضای خالی سمت چپ (ابتدا رشته) را حذف می کند و نمایش می دهد (کد اسکی کاراکترهای خالی ۳۲ است).

➤ برنامه‌ای که رشته‌ای را خوانده، با استفاده از اشاره گر و بازگشتی، طول رشته را حساب کرده، نمایش می‌دهد.

➤ **پروژه برنامه‌نویسی ۱:** یک پروژه هواپیمایی قصد دارد سیستم ذخیره و فروش بلیط را کامپیوتری کند. برنامه سیستم رزرو بلیط مراحل زیر را انجام می‌دهد (تعداد صندلی ۴۰ می‌باشد که شماره ۱ تا ۲۰ برای سیگاری‌ها و از ۲۱ تا ۴۰ برای غیر سیگاری‌ها در نظر گرفته می‌شود. در ضمن حداکثر ۱۰ شماره پرواز از ۱ تا ۱۰ داریم):

الف: آرایه دوبعدی با ۴۰ سطر و ۱۰ ستون در نظر گرفته شده که ستون‌های صفر تا ۹ شماره پروازهای ۱ تا ۱۰ را تعیین می‌کنند.

ب: شماره مسافر، نوع مسافر (سیگاری (۱)، غیر سیگاری (۰)) و شماره پرواز را دریافت می‌کند، سپس بلیط را رزرو می‌کند.

ج: اگر نوع مسافر سیگاری باشد، با توجه به شماره پرواز، به آن شمار پرواز در بخش سیگاری صندلی تخصیص می‌دهد. در صورتی که پر بودن بخش سیگاری‌ها از مشتری سؤال می‌کند آیا در بخش غیر سیگاری‌ها پذیرفته شود یا خیر و برعکس.

د: شماره مسافر به شماره صندلی در آرایه تخصیص می‌یابد.

سپس کلید n را فشار دهید تا وضعیت صندلی‌های پرواز را به شکل زیر ببینید:

➤ **پروژه برنامه‌نویسی ۲:** برنامه‌ای که منوی نمایش می‌دهد. از این منو کاربر می‌تواند یکی از گزینه‌های زیر را انتخاب کند:

۱ Read Array، اطلاعات آرایه اول را می‌خواند.

۲ Read Array، اطلاعات آرایه دوم را می‌خواند.

Find Share، اشتراک بین آرایه اول و دوم را پیدا می‌کند.

Find Union، اجتماع بین آرایه اول و دوم را پیدا می‌کند.

Find difference a Form b، تفاضل آرایه a را از b حساب می‌کند.

Find difference b Form a، تفاضل آرایه b را از a حساب می‌کند.

➤ **پروژه برنامه‌نویسی ۳:** برنامه‌ای که دو آرایه  $3 \times 3$  را خوانده، دو آرایه را با هم جمع، تفریق و ضرب می‌نماید. در این برنامه امکاناتی برای نمایش دو آرایه دارد. در ضمن این برنامه می‌تواند N، M و k را بخواند. سپس حاصل عبارت  $N * A - M * B + K$  را محاسبه کنند و نمایش دهد (A و B به ترتیب آرایه‌های اول و دوم هستند).

➤ **پروژه برنامه‌نویسی ۴:** برنامه‌ای که عناصر یک ماتریس سه در سه را خوانده، اعمال زیر را انجام می‌دهد:

تعیین می‌کند ماتریس بالا مثلثی است یا نه؟

تعیین می‌کند ماتریس پایین مثلثی است یا نه؟

تعیین می کند ماتریس بالا و پایین مثلثی است یا نه؟  
تعیین می کند که آیا تمام عناصر هر سطر ماتریس برابرند یا نه؟  
تعیین می کند که تمام عناصر هر ستون ماتریس برابرند یا نه؟

➤ پروژه برنامه نویسی ۵. فرض کنید بخواهیم وضعیت مدال های بین المللی یک کشور را در رشته ورزشی خاصی در مهر و موم های ۱۳۸۵ تا ۱۳۹۵ مطابق جدول زیر دریافت و پردازش کنیم:

| سال  | تعداد<br>مدال<br>طلا | تعداد<br>مدال<br>نقره | تعداد<br>مدال<br>برنز |
|------|----------------------|-----------------------|-----------------------|
| ۱۳۸۵ |                      |                       |                       |
| ۱۳۷۱ |                      |                       |                       |
| ۱۳۷۲ |                      |                       |                       |
|      |                      |                       |                       |
| ۱۳۹۴ |                      |                       |                       |
| ۱۳۹۵ |                      |                       |                       |

برنامه ای که ضمن دریافت داده های لازم (خانه های خالی طول) موارد زیر را نمایش می دهد.

تعداد کل مدال های کسب شده در همه ی سال ها  
تعداد کل مدال های طلای کسب شده در همه ی سال ها  
شماره سال هایی که هیچ مدالی کسب نشده  
سالی که بیش ترین مدال کسب شده  
سالی که بیش ترین مدال طلا کسب شده است.  
امتیاز تمام سال ها به تفکیک سال، با فرض این که هر مدال طلا ۳ امتیاز، هر مدال نقره ۲ امتیاز و هر مدال برنز ۱ امتیاز داشته باشد.  
تعداد سال هایی که امتیاز بیش از ۱۰ کسب شده است.

➤ پروژه برنامه نویسی ۶. در یک تورنمنت ژیمناستیک با شرکت ۱۰ ورزش کار و ۵ داور امتیاز منظر خود را در یک جدولی به شکل زیر یادداشت می کنند. ملاک انتخاب قهرمان امتیازات داده شده توسط داوران به هر ورزشکار است. گزارش گردید یکی از داوران از یک ورزشکار رشوه گرفته تا به قهرمان شدن او کمک کند. اما داور و ورزشکار متخلف شناسایی نشدند. با توجه به این که داور متخلف امتیازات پرت تری نسبت به همکارانش ثبت کرده است، کمیته برگزاری تصمیم گرفته با بررسی داده ها او را شناسایی کرده و امتیازات داده شده توسط وی را از جدول حذف نموده و در نهایت

## ۶۰۰ برنامه C++ با حل آن‌ها ۸۱

نام قهرمان واقعی و نام ورزشکار رشوه دهنده را مشخص کند. این برنامه‌ای کار را انجام می‌دهد.

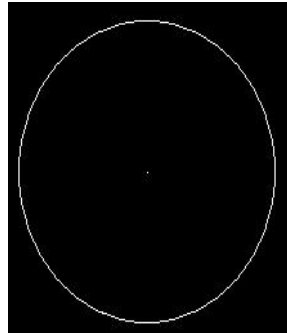
| امتیاز<br>داور<br>پنجم | ..... | امتیاز<br>داور<br>دوم | امتیاز<br>داور<br>اول | نام<br>ورزشکار | شماره<br>پیراهن |
|------------------------|-------|-----------------------|-----------------------|----------------|-----------------|
|                        |       |                       |                       |                | ۱               |
|                        |       |                       |                       |                | ۲               |
|                        |       |                       |                       |                | ۳               |
|                        |       |                       |                       |                | :               |
|                        |       |                       |                       |                | ۹               |
|                        |       |                       |                       |                | ۱۰              |

## فصل ۵

# گرافیک در C++

۱. برنامه‌ای که شکل زیر را رسم می‌کند:

```
#include <iostream>
#include <graphics.h>
using namespace std;
int main()
{
    initwindow(800,600);
    circle(300,300,100);
    putpixel(300,300,15);
    getch();
    closegraph();
    return 0;
}
```

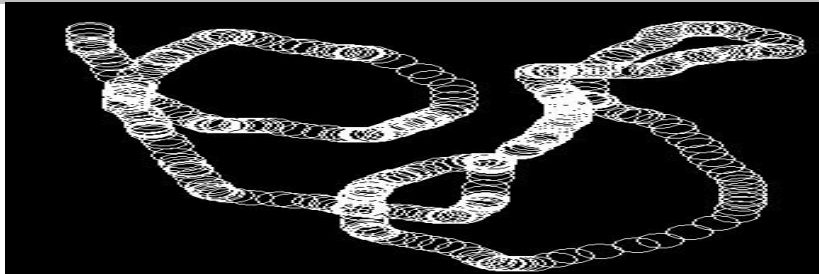


۲. برنامه‌ای که خروجی زیر را رسم می‌نماید:

```
#include <iostream>
#include <graphics.h>
using namespace std;
int main()
{
    int i=0;
    initwindow(1280,800,"",-3,-3);
    while(i<256)
    {
        setcolor(COLOR(i,0,0));
        line(i,0,i,800);
        i++;
    }
    getch();
    closegraph();
    return 0;
}
```



۳. برنامه‌ای که کاربر با حرکت ماوس می‌تواند تصویری مانند زیر را رسم کند:



```
#include <iostream>
#include <graphics.h>
using namespace std;
```

### ۶۰۰ برنامه C++ با حل آن‌ها ۸۳

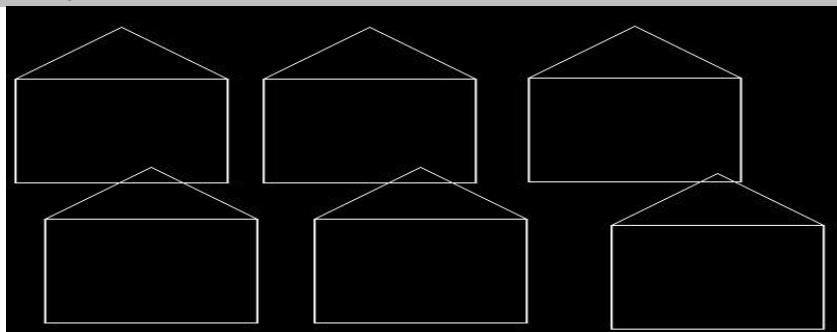
```
int main()
{
    initwindow(1280,800,"",-3,-3);
    while(!kbhit())
    {
        circle(mousex(),mousey(),10);
    }
    getch();
    closegraph();
    return 0;
}
```

۴. برنامه‌ای که مکان نمای ماوس را به شکل دایره و یک فلش تغییر می‌دهد (شکل زیر):

```
#include <iostream>
#include <graphics.h>
using namespace std;
int main()
{
    int x=-1,y=-1;
    initwindow(1280,800,"",-3,-3);
    while(!kbhit())
    {
        if((x!=mousex())||(y!=mousey()))
        {
            setcolor(0);
            circle(x,y,7);
            x=mousex();
            y=mousey();
            setcolor(15);
            circle(x,y,7);
        }
    }
    closegraph();
    return 0;
}
```



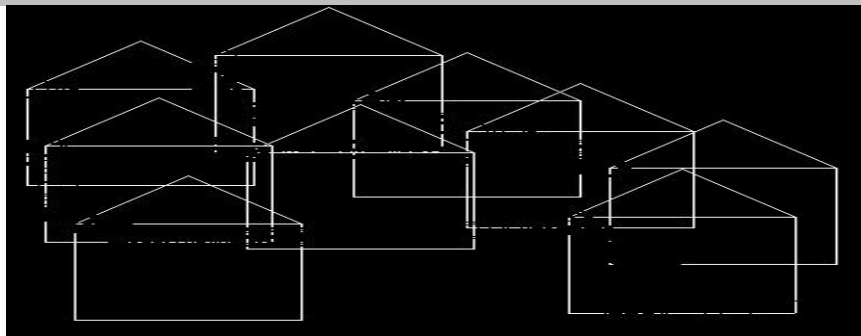
۵. برنامه‌ای که اشکال زیر را رسم می‌کند (با هر کلیک یک شکل از اشکال زیر رسم می‌شود):



```
#include <iostream>
#include <graphics.h>
using namespace std;
int main()
{
    int x=-1,y=-1;
```

```
initwindow(1280,800,"",-3,-3);
while(!kbhit())
{
    if((x!=mousex())||(y!=mousey()))
    {
        setcolor(0);
        circle(x,y,7);
        x=mousex();
        y=mousey();
        setcolor(15);
        circle(x,y,7);
    }
}
closegraph();
return 0;
}
```

۶. برنامه که با کلیک روی صفحه یکی از اشکال زیر رسم می‌شود و با کلیک دوباره آن شکل، بخشی از شکل با نقطه پاک می‌شود.

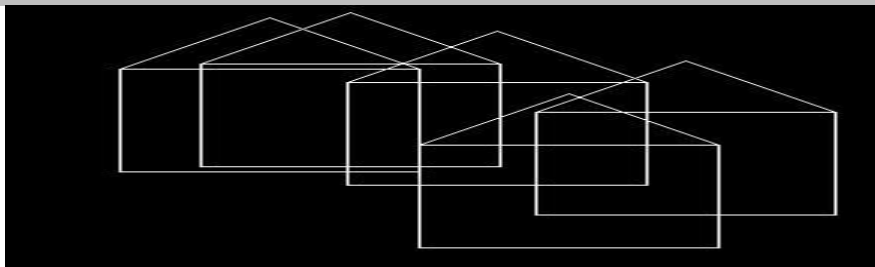


```
#include <iostream>
#include <graphics.h>
using namespace std;
int main()
{
    int x=-1,y=-1;
    initwindow(1280,800,"",-3,-3);
    while(!kbhit())
    {
        if((x!=mousex())||(y!=mousey()))
        {
            setcolor(0);
            circle(x,y,10);
            x=mousex();
            y=mousey();
            setcolor(15);
            circle(x,y,10);
        }
        if(ismouseclick(WM_LBUTTONDOWN))
        {
            rectangle(x,y,x+100,y+100);
            line(x,y,x+50,y-50);
        }
    }
}
```

## ۶۰۰ برنامه C++ با حل آن‌ها ۸۵

```
    line(x+50,y-50,x+100,y);
    clearmouseclick(WM_LBUTTONDOWN);
}
}
closegraph();
return 0;
}
```

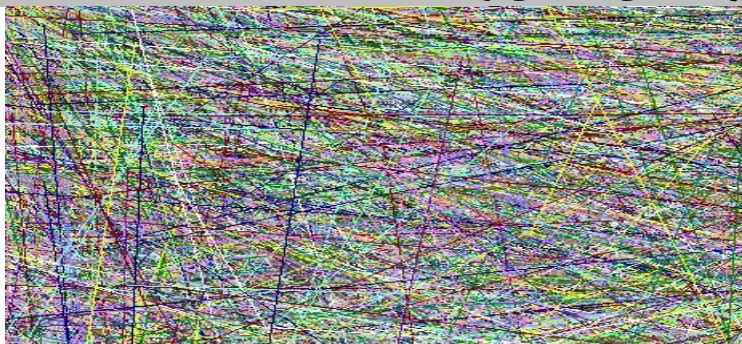
۷. برنامه‌ای که با کلیک ماوس اشکال زیر را رسم می‌کند و با حرکت دادن دکمه وسط ماوس اشکال رسم شده با شکل جدید XOR می‌گردد.



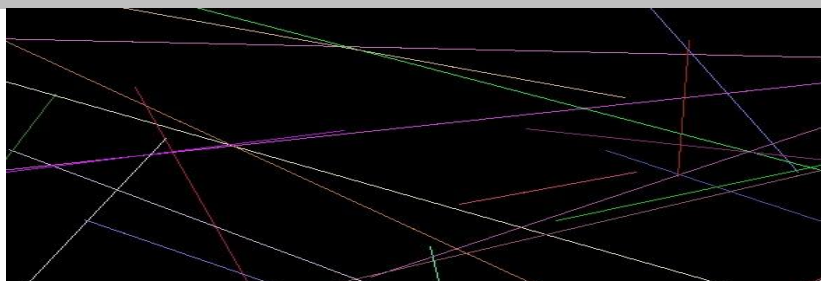
```
#include <iostream>
#include <graphics.h>
using namespace std;
int main()
{
    int x=-100,y=-100;
    initwindow(1280,800,"",-3,-3);
    while(!kbhit())
    {
        if((x!=mousex())||(y!=mousey()))
        {
            setwritemode(XOR_PUT);
            setcolor(15);
            circle(x,y,10);
            x=mousex();
            y=mousey();
            setcolor(15);
            circle(x,y,10);
            setwritemode(COPY_PUT);
        }
        if(ismouseclick(WM_LBUTTONDOWN))
        {
            setwritemode(XOR_PUT);
            setcolor(15);
            circle(x,y,10);
            setwritemode(COPY_PUT);
            rectangle(x,y,x+100,y+100);
            line(x,y,x+50,y-50);
            line(x+50,y-50,x+100,y);
            clearmouseclick(WM_LBUTTONDOWN);
        }
    }
}
```

```
setwritemode(XOR_PUT);
setcolor(15);
circle(x,y,10);
setwritemode(COPY_PUT);
}
}
closegraph();
return 0;
}
```

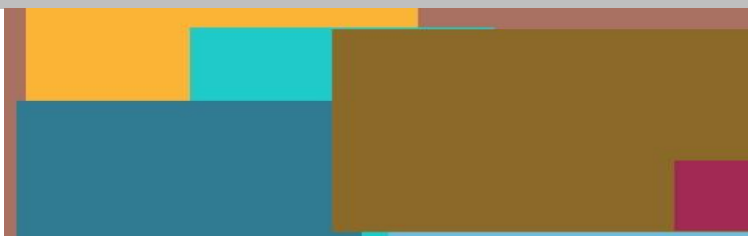
۸. برنامه‌ای که به صوت تصادفی صفحه مانیتور را با خطوط رنگی پر می‌کند (رنگ، شروع و پایان خطوط به صورت تصادفی انتخاب می‌گردد).



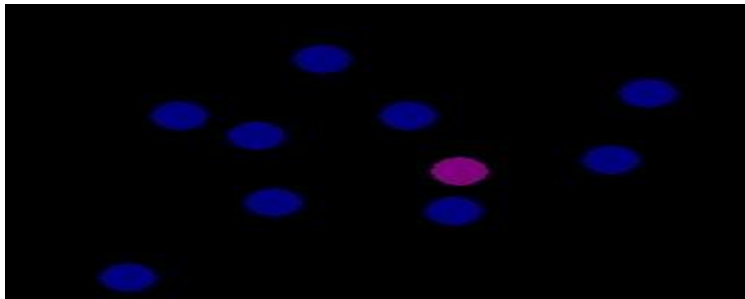
۹. برنامه که خطوط با رنگ‌های مختلف به صورت تصادفی با سرعت پایین بر روی صفحه‌نمایش رسم می‌کند.



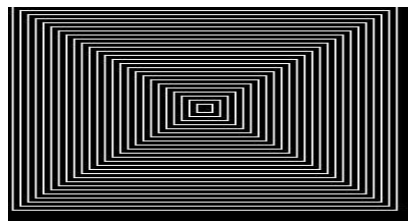
۱۰. برنامه‌ای که مستطیل‌های توپر به صورت تصادفی با رنگ‌های مختلف بر روی صفحه‌نمایش رسم می‌کند.



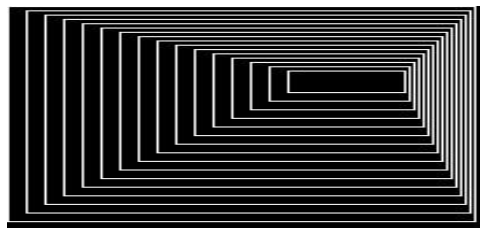
۱۱. برنامه‌ای که با کلیک ماوس دایره‌هایی به شکل‌های زیر رسم می‌کند:



۱۲. برنامه‌ای که شکل زیر را رسم می‌کند:

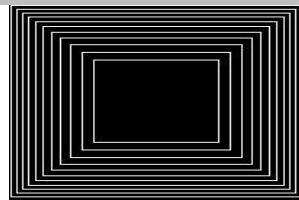


۱۳. برنامه‌ای که شکل زیر را رسم می‌کند:

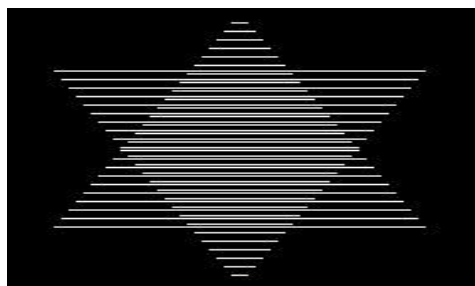


۱۴. برنامه‌ای که شکل زیر را رسم می‌کند:

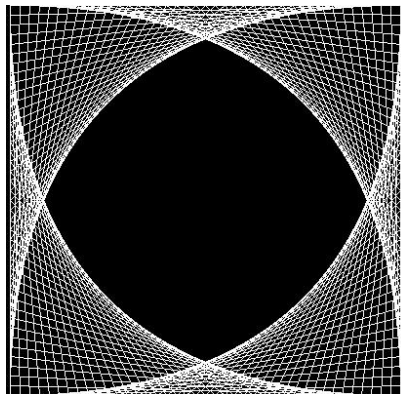
#



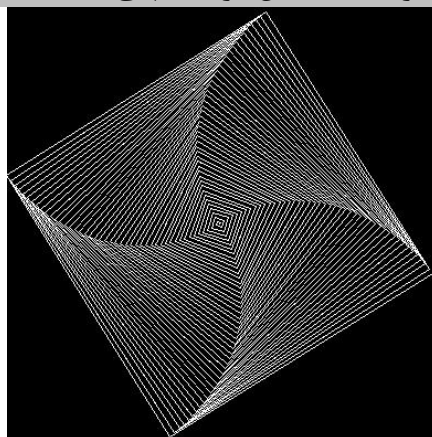
۱۵. برنامه‌ای که شکل زیر را رسم می‌کند:



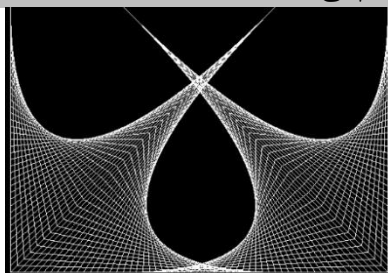
۱۶. برنامه‌ای که شکل زیر را رسم می‌کند:



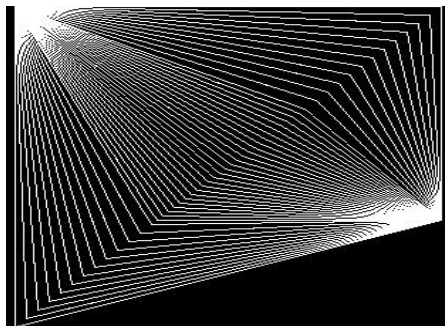
۱۷. برنامه‌ای که شکل زیر را رسم می‌کند:



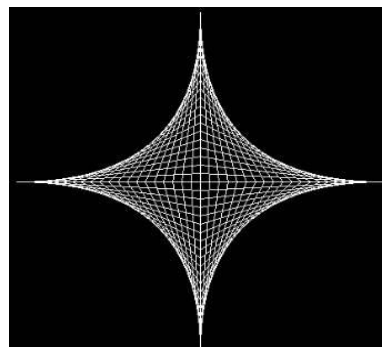
۱۸. برنامه‌ای که شکل زیر را رسم می‌کند:



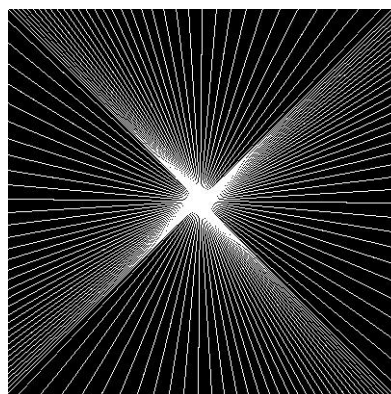
۱۹. برنامه‌ای که شکل زیر را رسم می‌کند:



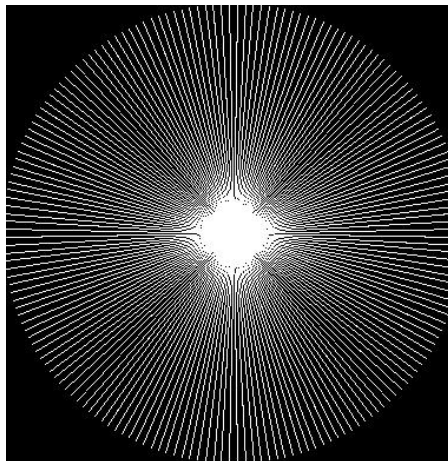
۲۰. برنامه‌ای که شکل زیر را رسم می‌کند:



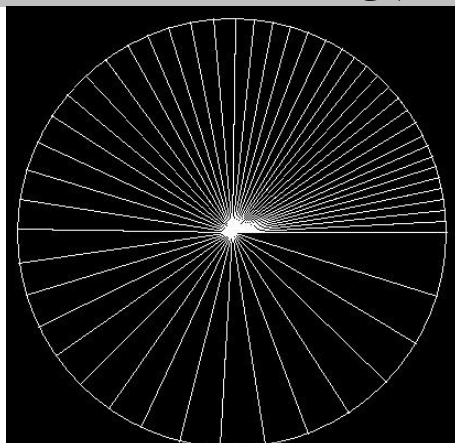
۲۱. برنامه‌ای که شکل زیر را رسم می‌کند:



۲۲. برنامه‌ای که شکل زیر را رسم می‌کند:



۲۳. برنامه‌ای که شکل زیر را رسم می‌کند:



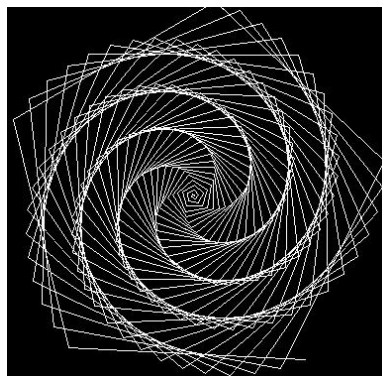
```
#include <iostream>
#include <graphics.h>
#include <math.h>
using namespace std;
int main()
{
    initwindow(۱۲۸۰,۸۰۰,"",-۳,-۳);
    int x, y;
    //ellipse(۱۲۵۰,۳۶۰, ۳۹۰,۳۹۰);

    for (int i = ۰; i <= ۳۴۰; i++)
    {
        i += i / ۳۰;
        x = ۳۲۰+۱۹۵* cos(۳,۱۴۱۵ * i / ۱۸۰);
```

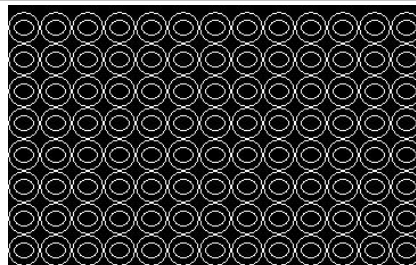
## ۶۰۰ برنامه C++ با حل آن‌ها ۹۱

```
y = ۲۰۰+۱۹۵*sin(۳,۱۴۱۵ * i / ۱۸۰);  
line( x, y, ۳۲۰, ۲۰۰);  
}  
getch();  
closegraph();  
return ۰;  
}
```

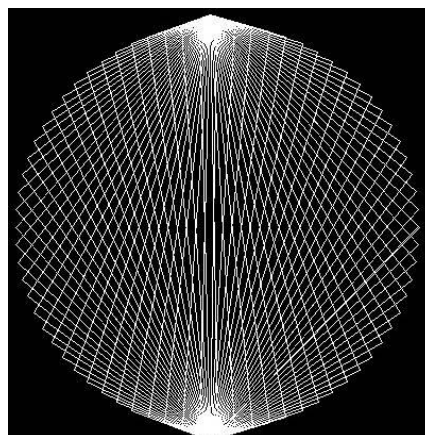
۲۴. برنامه‌ای که شکل زیر را رسم می‌کند:



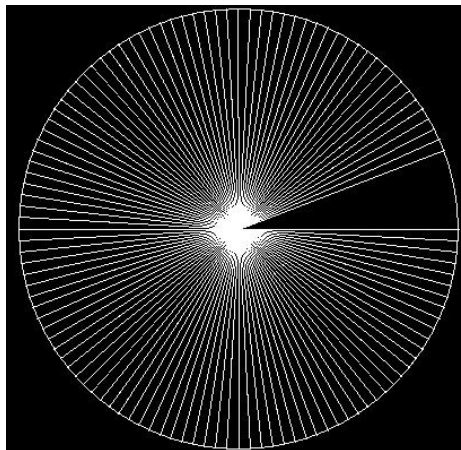
۲۵. برنامه‌ای که شکل زیر را رسم می‌نماید:



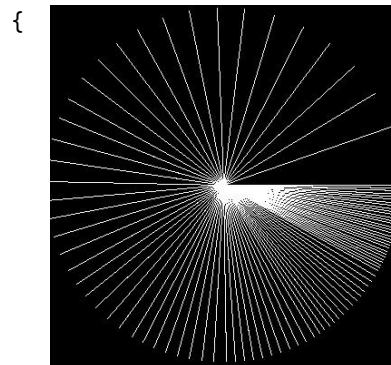
۲۶. برنامه‌ای که شکل زیر را رسم می‌کند:



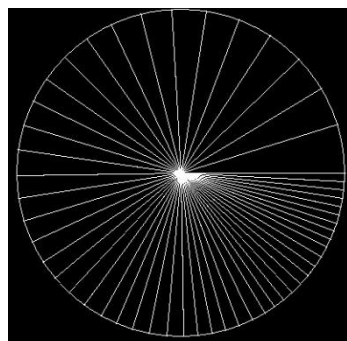
۲۷. برنامه‌ای که شکل زیر را رسم می‌نماید:



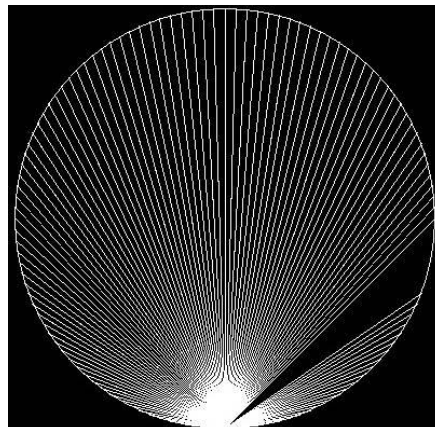
۲۸. برنامه‌ای که شکل زیر را رسم می‌کند:



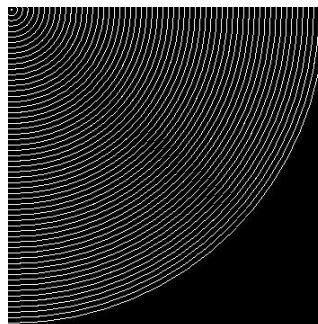
۲۹. برنامه‌ای که شکل زیر را رسم می‌کند:



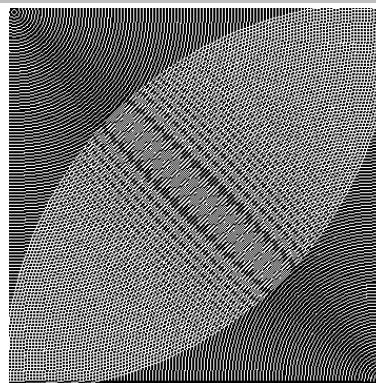
۳۰. برنامه‌ای که شکل زیر را رسم می‌کند:



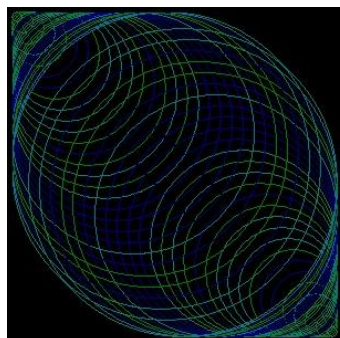
۳۱. برنامه‌ای که شکل زیر را رسم می‌کند:



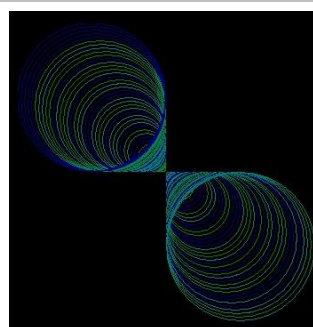
۳۲. برنامه‌ای که شکل زیر را رسم می‌کند:



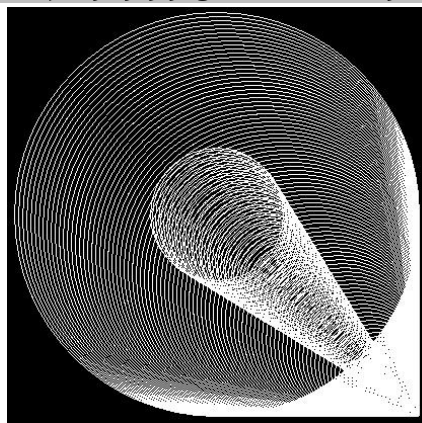
۳۳. برنامه‌ای که شکل زیر را رسم می‌کند:



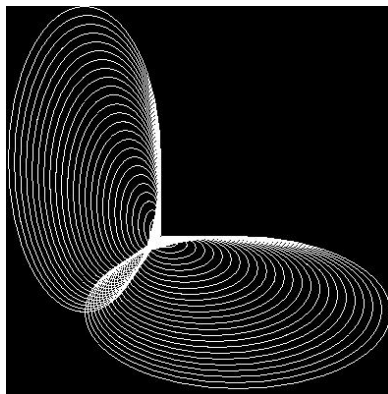
۳۴. برنامه‌ای که شکل زیر را رسم می‌کند:



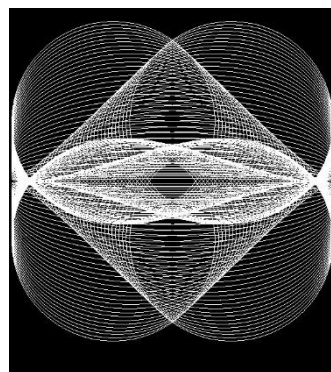
۳۵. برنامه‌ای که شکل زیر را رسم می‌کند:



۳۶. برنامه‌ای که شکل زیر را رسم می‌کند:



۳۷. برنامه‌ای که شکل زیر را رسم می‌کند:



۳۸. برنامه‌ای که شکل زیر را رسم می‌کند:



## ساختمان و کلاس‌ها

## فصل ۶

۱. برنامه‌ای که اطلاعات ۵ کتاب از قبیل ISBN، نام و قیمت را خوانده، در آرایه‌ای از ساختمان قرار می‌دهد. سپس، اعمال زیر را انجام می‌دهد:

اطلاعات کتاب‌ها را بر اساس قیمت مرتب کرده، نمایش می‌دهد.

اطلاعات کتاب‌ها را بر اساس نام کتاب مرتب کرده، نمایش می‌دهد.

شابلون کتابی را دریافت کرده، آن کتاب را حذف می کند و بقیه کتاب ها را نمایش می دهد.

| تابع                            | متغیر یا نوع |                                                                                        |
|---------------------------------|--------------|----------------------------------------------------------------------------------------|
| همه توابع                       | نوع Book     | ساختمانی است که شامل فیلدهای ISBN (شابلون)، bookName (نام کتاب)، price (قیمت) می باشد. |
|                                 | پارامتر book | آرایه ای از نوع ساختمان Book به عنوان پارامتر توابع است.                               |
|                                 | پارامتر n    | تعداد عناصر آرایه ساختمان Book به عنوان پارامتر                                        |
| sortWithPrice, sortWithBookName | i            | شمارنده ای برای پیمایش آرایه book                                                      |
|                                 | j            | شمارنده ای برای پیمایش عناصر آرایه جهت مرتب سازی                                       |
|                                 | t            | متغیر کمکی از نوع ساختمان Book برای تعویض دو عنصر آرایه با نوع ساختمان Book            |
| stringCompare                   | پارامتر *s1  | رشته اول با نوع اشاره گر که باید با رشته دوم مقایسه گردد.                              |
|                                 | پارامتر *s2  | رشته دوم که باید با رشته اول مقایسه گردد.                                              |
| deleteBook                      | flagDelete   | آیا شابلون کتاب برای حذف پیدا شده یا خیر.                                              |
| Main                            | isbn         | شماره شابلون که می خواهید حذف کنید.                                                    |

```
#include <iostream>
using namespace std;
struct Book
{
    int ISBN;
    char bookName[30];
    int price;
};
void readBook(Book book[], int n)
{
    for(int i = 0; i < n; i++)
    {
        cout << "please enter a ISBN , price && bookName:";
        cin >> book[i].ISBN >> book[i].price;
        cin.getline(book[i].bookName, 30);
    }
}
void sortWithPrice(Book book[], int n)
{
    for ( int i = 0; i < n-1; i++)
    {
        for(int j = i+ 1; j < n; j++)
            if (book[i].price > book[j].price )
            {
                Book t= book[i];
                book[i]= book[j];
                book[j] = t;
            }
    }
}
int stringCompare(char *s1, char *s2)
{
    while (*s1 == *s2 && *s1 && *s2)
    {
        s1++;
        s2++;
    }
}
```

```

    }
    if (*s1 > *s2 )
        return (1);
    else if(*s1 < *s2 )
        return -1;
    else return 0;
}
void sortWithbookName(Book book[], int n)
{
    for ( int i = 0; i < n-1; i++)
    {
        for(int j = i+ 1; j < n; j++)
            if(stringCompare(book[i].bookName, book[j].bookName) > 0 )
            {
                Book t= book[i];
                book[i]= book[j];
                book[j] = t;
            }
    }
}
void deleteBook(Book book[], int *n,int ISBN1)
{
    int i;
    for( i=0; i < *n; i++)
        if ( ISBN1 == book[i].ISBN) break;
    if (i == *n - 1 )
    {
        *n --;
        return;
    }
    int flagDelete =0;
    for ( int j = i; j < *n-1; j++ )
    {
        flagDelete=1;
        book[j] = book[j+1];
    }
    if (flagDelete !=0)
    {
        (*n) --;
    }
}
void printBook(Book book[], int n)
{
    cout << "ISBN" << "\t" << "price\tbookName\n";
    cout << "=====\t=====\t=====\n";
    for(int i=0; i < n; i++)
    {
        cout << book[i].ISBN << "\t" << book[i].price << "\t" <<
            book[i].bookName << "\n";
    }
}
int main()
{
    Book    book[5];
    int n = 5, isbn;
    readBook(book, n);
    sortWithPrice(book, n);
    cout << "\n    Sort By Price \n" ;
    printBook(book, n);
}

```

```
sortWithbookName(book, n);
cout << "\n Sort By Name \n" ;
printBook(book, n);
cout << "\nEnter ISBN for deleted:";
cin >> isbn;
deleteBook(book, &n, isbn);
printBook(book, n);
return 0;
}
```

The screenshot shows a Windows command prompt window titled "D:\BookCSharp\csharp\Practice\6\Main\Debug\1.exe". The program prompts the user to enter ISBN, price, and book name for five books. The input data is as follows:

| ISBN | price | bookName    |
|------|-------|-------------|
| 1    | 4500  | C           |
| 2    | 6100  | C#          |
| 3    | 6000  | LINQ        |
| 4    | 6500  | Database C# |
| 5    | 6000  | Database    |

The program then displays two sorted lists:

**Sort By Price**

| ISBN | price | bookName    |
|------|-------|-------------|
| 1    | 4500  | C           |
| 3    | 6000  | LINQ        |
| 5    | 6000  | Database    |
| 2    | 6100  | C#          |
| 4    | 6500  | Database C# |

**Sort By Name**

| ISBN | price | bookName    |
|------|-------|-------------|
| 1    | 4500  | C           |
| 2    | 6100  | C#          |
| 5    | 6000  | Database    |
| 4    | 6500  | Database C# |
| 3    | 6000  | LINQ        |

The program then prompts the user to enter an ISBN for deletion. The user enters 3. The program displays the remaining books:

| ISBN | price | bookName |
|------|-------|----------|
| 1    | 4500  | C        |
| 2    | 6100  | C#       |
| 5    | 6000  | Database |
| 3    | 6000  | LINQ     |

The program returns 0 (0x0) and execution time is 123.628 s. Press any key to continue.

عملکرد توابع:

- تابع readBook()** آرایه book با نوع ساختمان Book و تعداد عناصر آن n را به عنوان پارامتر دریافت کرده، اطلاعات n کتاب را می خواند و در آرایه قرار می دهد.
- تابع sortWithPrice()** آرایه book با نوع ساختمان و تعداد عناصر آن n را به عنوان پارامتر دریافت کرده، عناصر آرایه را بر اساس قیمت مرتب می نماید (مرتب سازی، حبابی است).
- تابع stringCompare()** دو رشته S1 و S2 با نوع اشاره گر را به عنوان پارامتر دریافت کرده، اگر رشته اول بزرگ تر از رشته دوم باشد، مقدار ۱ (یک)، و گرنه، اگر رشته دوم بزرگ تر از رشته اول باشد، مقدار -۱، و گرنه صفر را برمی گرداند.
- تابع sortWithBookName()** آرایه book با نوع ساختمان Book و تعداد عناصر آن (n) را به عنوان پارامتر دریافت می کند و عناصر آرایه را بر اساس نام کتاب مرتب می کند.
- تابع deleteBook()** آرایه book با نوع ساختمان Book، اشاره گر n (تعداد عناصر آرایه book) و ISBN (شماره شابکی که باید حذف شود) را به عنوان پارامتر دریافت می کند. اگر کتاب با شماره شابک ISBN در آرایه book وجود داشته باشد آن را حذف خواهد کرد.
- تابع printBook()** آرایه book با نوع ساختمان Book و تعداد عناصر آن (n) را به عنوان پارامتر دریافت می کند و اطلاعات کتاب ها را با یک پیغام مناسب نمایش می دهد.

🚩 تابع (main)، ابتدا، آرایه book را با ۵ عنصر با نوع ساختمان Book تعریف می‌کند. سپس، ISBN و n را تعریف کرده، n را برابر ۵ قرار می‌دهد. تابع readBook را فراخوانی می‌کند تا اطلاعات کتاب‌ها را بخواند. تابع sortWithPrice() را فراخوانی می‌کند تا اطلاعات را بر اساس قیمت مرتب نماید و با فراخوانی تابع printBook() اطلاعات مرتب‌شده را نمایش می‌دهد. در ادامه، تابع sortWithBookName را فراخوانی می‌کند تا اطلاعات را بر اساس نام کتاب مرتب کند و با فراخوانی تابع printBook، اطلاعات مرتب‌شده را نمایش می‌دهد. در پایان، با یک پیغام شبکی را خوانده با فراخوانی تابعي deteteBook چنانچه آن شبک در آرایه book وجود داشته باشد، آن را حذف می‌کند و اطلاعات کتاب‌ها را با فراخوانی تابع printBook() نمایش می‌دهد.

۲. برای مشتریان سازمان تأمین اجتماعی یک ساختمان که دارای اطلاعاتی از قبیل شماره بیمه‌شده، نام بیمه‌شده، ساختمان دیگری شامل تاریخ بیمه و با فیلدهای روز، ماه و سال باشد، داریم. برنامه‌ای که اطلاعات مشتریان این سازمان را خوانده در آرایه‌ای از ساختمان‌ها قرار می‌دهد. سپس، سال فعلی را از ورودی خوانده سابقه بیمه تمام مشتریان را به همراه شماره بیمه، تاریخ بیمه و نام آن‌ها را به خروجی می‌برد. تابعی برای خواندن اطلاعات و تابعی برای تهیه گزارش بنویسید.

```
#include
<iostream>
#include
<stdio.h>
#include
<stdlib.h>
#define MAX 200
using namespace
std;
struct date {
    int year,
    month, day;
};
struct
custom_insure {
    int code;
    char name[30];
    struct date date1;
};
void gets_info(struct custom_insure custom1[], int n)
{
    char s[20];
    int i;
    for (i=0; i< n; i++)
```

| هدف                                                                                                                         | نام                                                                         | تابع      |
|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|-----------|
| نوع ساختمان سابقه<br>حداکثر تعداد مشتریان<br>نوع ساختمان مشتریان بیمه                                                       | Date<br>MAX<br>custom_insure                                                | عمومی     |
| آرایه با نوع ساختمان custom_insure<br>تعداد مشتریان                                                                         | custom۱<br>n                                                                | main      |
| رشته کمکی<br>شمارنده                                                                                                        | S<br>i                                                                      | get_info  |
| سال جاری<br>ماه جاری<br>روز جاری<br>سابقه به سال<br>سابقه به ماه<br>سابقه به روز<br>شمارنده<br>رقم نقلی ماه<br>رقم نقلی سال | year۱<br>month۱<br>day۱<br>yy<br>mm<br>dd<br>i<br>carry_month<br>carry_year | puts_info |

```
{
    cout << "Enter code[" << i + 1 << "]:";
    gets(s);
    custom1[i].code = atoi(s);
    cout << "Enter name[" << i+1 << "]:" ;
    gets(custom1[i].name);
    cout << "Enter date(yy/mm/dd)[" <<i+1 << "]:";
    cin >> custom1[i].date1.year >> custom1[i].date1.month >>
        custom1[i].date1.day;
    getchar();
}
}
void puts_info(struct custom_insure custom1[], int n)
{
    int i, year1, month1, day1, yy, mm, dd, carry_month,
        carry_year;
    cout << "Enter date(yy/mm/dd):";
    cin >> year1 >> month1 >> day1;
    cout << "\nCode\t Name\t\tDate\t\tWork date";
    cout << "\n====\t\====\t\t\t=====\t\t\t=====";
    for (i=0; i< n; i++)
    {
        carry_month = 0;
        carry_year = 0;
        if (custom1[i].date1.day > day1)
        {
            dd = day1 + 30 - custom1[i].date1.day;
            carry_month = 1;
        }
        else
            dd = day1 - custom1[i].date1.day;
        if (custom1[i].date1.month > month1)
        {
            mm = month1 + 12 - custom1[i].date1.month - carry_month;
            carry_year = 1;
        }
        else
            mm = month1 - custom1[i].date1.month - carry_month;
        yy = year1 - custom1[i].date1.year - carry_year;
        cout << "\n" << custom1[i].code << "\t" << custom1[i].name
            << "\t\t" << custom1[i].date1.year << "/" <<
                custom1[i].date1.month << "/" << custom1[i].date1.day
            << "\t" << yy << "/" << mm << "/" << dd;
    }
}
int main()
{
    int n;
    struct custom_insure custom1[MAX];
    cout << "Enter n(custom number):";
    cin >> n;
    getchar();
    gets_info(custom1, n);
    puts_info(custom1, n);
    return 0;
}
```

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۰۱

```

D:\BookCSharp\cppnew\Practice\6\2\bin\Debug\2.exe
Enter n(custom number):2
Enter code[1]:1
Enter name[1]:Ali
Enter date(yy/mm/dd)[1]:12 10 20
Enter code[2]:2
Enter name[2]:Reza
Enter date(yy/mm/dd)[2]:14 10 20
Enter date(yy/mm/dd):96 9 4

Code      Name      Date      Work date
=====
1         Ali      12/10/20  83/10/14
2         Reza      14/10/20  81/10/14
Process returned 0 (0x0)   execution time : 35.873 s
Press any key to continue.

```

۳. فرودگاهی دارای ۲۰ باند پرواز است. برنامه‌ای برای تنظیم و فرود هواپیماها به شرح زیر بنویسید:  
 الف) آرایه‌ای ۲۰ عنصری از ساختمان تعریف کنید که فیلدهای ساختمان به این شرح باشند: شماره باند فرود، وضعیت باند (۱= پر یا رزرو شده، ۰= خالی) و نام کسی که باند را کنترل می‌کند.

ب) منویی با گزینه‌های زیر ظاهر کند:

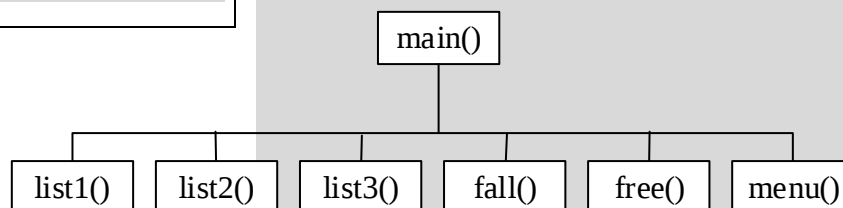
ج) تمام عملیات موجود در منو را انجام دهد.

وظایف توابع:

تابع main: شماره باندها را مقدار داده، سپس، هر یک از توابع را مطابق با سلسله‌مراتب زیر فراخوانی می‌کند.

تابع list1: باندهای خالی را نمایش می‌دهد.

۱. نمایش تعداد باندهای خالی
  ۲. نمایش شماره باندهای خالی
  ۳. نمایش باندهای پر یا رزرو شده
  ۴. تخصیص باند فرودگاه به هواپیما
  ۵. آزاد کردن باند فرودگاه
  ۶. خروج
- انتخاب خود را وارد کنید (۱-۶):



تابع list2: لیست باندهای خالی (شماره باندهای خالی) را نمایش می‌دهد.

تابع list3: لیست باندهای پر راه همراه با افرادی که آن‌ها را پر نموده‌اند، نمایش می‌دهد.

تابع full: شماره باندی را دریافت کرده، در صورتی که این باند پر نباشد، آن را پر می‌نماید.

تابع free: شماره باندی را دریافت کرده، در صورت پر بودن این باند آن را آزاد می‌کند.

تابع menu: منویی را نمایش می‌دهد تا کاربر بتواند گزینه‌ای را انتخاب کند.

| تابع  | نام         | هدف                                                                        |
|-------|-------------|----------------------------------------------------------------------------|
| عمومی | MAX<br>band | ماکروبی که تعداد باندها را مشخص می‌کند<br>آرایه‌ای که از نوع ساختمان bands |
| Main  | i           | شمارنده‌ای که برای مقداردهی به شماره باندها به کار می‌رود                  |
| menu  | choose      | شماره انتخابی را که توسط کاربر وارد شده است                                |

|                                                   |       |       |
|---------------------------------------------------|-------|-------|
| مشخص می کند                                       |       |       |
| شمارنده                                           | i     | list۱ |
| تعداد باندهای خالی را مشخص می کند                 | count |       |
| شمارنده                                           | i     | list۲ |
| شمارنده                                           | i     | list۳ |
| شماره باندهایی را تعیین می کند که باید پر شوند.   | num   | Full  |
| شماره باندهایی را تعیین می کند که باید آزاد شوند. | num   | Free  |

```
#include <iostream>
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#define MAX 20
using namespace std;
struct bands {
    int no;
    char name[30];
    int full;
};
struct bands    band[MAX];
int menu()
{
    int choose = 0;
    cout << "\n1. List number of free band";
    cout << "\n2. List free band numbers";
    cout << "\n3. List full band numbers";
    cout << "\n4. Full band";
    cout << "\n5. Free band";
    cout << "\n6. Exit";
    cout << "\nSelect 1..6:";
    cin >> choose;
    return choose;
}
void list1()
{
    int i, count = 0;
    for (i=0; i< MAX; i++)
        if (band[i].full == 0)
            count ++;
    cout << "\nNumber of free band is " <<  count;
}
void list2()
{
    int i;
    cout << "\nList of free bunds ";
    for (i=0; i< MAX; i++)
        if (band[i].full == 0)
            cout << band[i].no << " ";
}
void list3()
{
    int i;
    for (i=0; i< MAX; i++)
        if (band[i].full )
            cout << "band " << band[i].no << " by full " <<
                band[i].name << endl;
}
void full()
{

```

### ۶۰۰ برنامه C++ با حل آن‌ها ۱۰۳

```
int num;
cout <<"Enter band no(1..20):";
cin >> num;
if (band[num -1 ].full)
{
    cout <<"Band is " << num << " full" << endl;
    getch();
    return;
}
else
{
    band[num -1 ].full = 1;
    cout <<"Enter bund full by:";
    cin >> band[num-1].name;
}
}
void free()
{
    int num;
    cout <<"Enter band no(1..20):";
    cin >> num;
    if (band[num -1 ].full == 0)
    {
        cout <<"Band is " << num << " free" << endl;
        return;
    }
    else
    {
        band[num -1 ].full = 0;
        band[num-1].name[0] = '\0';
    }
}
int main()
{
    int i;
    for(i = 0; i < MAX; i++ ) band[i].no = i+1;
    while(1)
    {
        switch(menu())
        {
            case 1:
                list1();
                break;
            case 2:
                list2();
                break;
            case 3:
                list3();
                break;
            case 4:
                full();
                break;
            case 5:
                free();
                break;
            case 6:
                return 0;
                break;
        }
    }
}
```

```

    }
    return 0;
}

```

```

D:\BookCSharp\cppnewPractice\6\3\bin\Debug\3.exe
1. List number of free b
2. List free band number:
3. List full band number:
4. Full band
5. Free band
6. Exit
Select 1..6:4
Enter band no(1..20):10
Enter bund full by:Ramzan
1. List number of free b
2. List free band number:
3. List full band number:
4. Full band
5. Free band
6. Exit
Select 1..6:4
Enter band no(1..20):14
Enter bund full by:Reza
1. List number of free b
2. List free band number:
3. List full band number:
4. Full band
5. Free band
6. Exit
Select 1..6:1

```

```

D:\BookCSharp\cppnewPractice\6\3\bin\Debug\3.exe
Number of free band is 18
1. List number of free band
2. List free band numbers
3. List full band numbers
4. Full band
5. Free band
6. Exit
Select 1..6:2
List of free bands 1 2 3 4 5 6 7 8 9 11 12 13 15 16 17 18 19 20
1. List number of free band
2. List free band numbers
3. List full band numbers
4. Full band
5. Free band
6. Exit
Select 1..6:3
band 10 by full Ramzan
band 14 by full Reza
1. List number of free band
2. List free band numbers
3. List full band numbers
4. Full band
5. Free band
6. Exit
Select 1..6:5
Enter band no(1..20):10
1. List number of free band
2. List free band numbers
3. List full band numbers
4. Full band
5. Free band
6. Exit
Select 1..6:3
band 14 by full Reza
1. List number of free band
2. List free band numbers
3. List full band numbers
4. Full band
5. Free band
6. Exit
Select 1..6:6

```

۴. برنامه‌ای که ضرایب و توان دو چندجمله‌ای را از ورودی خوانده، این دو چندجمله‌ای را با یکدیگر جمع می‌کند.

۵. برنامه‌ای که صورت و مخرج دو کسر را خوانده، حاصل ضرب، تقسیم، تفریق و حاصل جمع آن‌ها را محاسبه کرده به خروجی می‌برد.

### وظایف توابع:

🚩 تابع main: مقادیر کسری را خوانده، با فراخوانی توابع نتایج را چاپ می‌کند.

🚩 تابع add: دو کسر را با یکدیگر جمع کرده، در کسر سوم قرار می‌دهد.

🚩 تابع sub: دو کسر را از یکدیگر کم کرده و در کسر سوم قرار می‌دهد.

🚩 تابع div: دو کسر را بر یکدیگر تقسیم کرده، در کسر سوم قرار می‌دهد.

🚩 تابع mul: دو کسر را با یکدیگر ضرب کرده، در کسر سوم قرار می‌دهد.

| تابع | نام | هدف                                                 |
|------|-----|-----------------------------------------------------|
| main | a   | اولین کسر                                           |
|      | b   | دومین کسر                                           |
|      | c   | مجموع، تفاضل، ضرب و تقسیم دو کسر را نگهداری می‌کند. |
| add  | p3  | مجموع دو کسر                                        |
| sub  | p3  | تفاضل دو کسر                                        |
| div  | p3  | تقسیم دو کسر                                        |
| mul  | p3  | حاصل ضرب دو کسر                                     |

```
#include <iostream>
using namespace std;
struct rational {
    int a, b;
};
struct rational add(struct rational p1, struct rational p2 )
{
    struct rational p3;
    p3.a = p1.a*p2.b+p2.a* p1.b;
    p3.b = p2.b* p1.b;
    return p3;
}
struct rational sub(struct rational p1, struct rational p2 )
{
    struct rational p3;
    p3.a=p1.a*p2.b - p2.a* p1.b;
    p3.b = p2.b* p1.b;
    return p3;
}
struct rational mul(struct rational p1, struct rational p2 )
{
    struct rational p3;
    p3.a=p1.a* p2.a;
    p3.b = p2.b* p1.b;
    return p3;
}
struct rational div(struct rational p1, struct rational p2 )
{
    struct rational p3;
    p3.a =p1.a * p2.b;
    p3.b = p1.b * p2.a;
    return p3;
}
```

```

    }
int main()
{
    struct rational a = {0}, b = {0}, c={0};
    cout << "Enter a/b for Fraction a:";
    cin >> a.a >> a.b;
    cout << "Enter a/b for Fraction b:";
    cin >> b.a >> b.b;
    c = add(a, b);
    cout<<"\n" << a.a << "/" << a.b << " + " << b.a << "/" <<
        b.b << " = " << c.a << "/" << c.b;
    c = sub(a, b);
    cout << "\n" << a.a << "/" << a.b << " - " << b.a << "/" <<
        b.b << " = " << c.a << "/" << c.b;
    c = mul(a, b);
    cout << "\n" << a.a << "/" << a.b << " * " << b.a << "/" <<
        b.b << " = " << c.a << "/" << c.b;
    c = div(a, b);
    cout << "\n" << a.a << "/" << a.b << " / " << b.a << "/" <<
        b.b << " = " << c.a << "/" << c.b;
    return 0;
}

```

The screenshot shows a Windows command prompt window titled "D:\BookCSharp\cppnewPractice\6\5\bin\Debug\5.exe". The program prompts the user to enter two fractions. The first fraction is 1/5 and the second is 6/7. The program then displays the results of four operations: addition (1/5 + 6/7 = 37/35), subtraction (1/5 - 6/7 = -23/35), multiplication (1/5 \* 6/7 = 6/35), and division (1/5 / 6/7 = 7/30). At the bottom, it shows "Process returned 0 (0x0) execution time : 5.048 s" and "Press any key to continue.".

۶. برنامه‌ای که اطلاعات دانشجویان از قبیل شماره دانشجویی، نام و معدل را در رکورد قرار داده اعمال زیر را انجام می‌دهد:

۱. اطلاعات تعدادی دانشجو را می‌خواند
۲. لیست دانشجویان را به‌صورت مرتب‌شده بر اساس شماره دانشجویی نمایش می‌دهد.
۳. لیست دانشجویان را به‌صورت مرتب‌شده بر اساس نام نمایش می‌دهد.
۴. لیست دانشجویان را به‌صورت مرتب‌شده بر اساس معدل نمایش می‌دهد.
۵. دانشجوی خاصی را بر اساس شماره دانشجویی جست‌وجو می‌کند.
۶. دانشجوی خاصی را بر اساس نام پیدا می‌کند.

۷. برنامه‌ای که ضلع مربع را خوانده، محیط و مساحت مربع را نمایش می‌دهد (هدف از این برنامه ایجاد کلاس، نمونه‌سازی و فراخوانی متدهای آن است).

۸. برنامه‌ای که کلاسی برای مثلث به نام triangle پیاده‌سازی می‌کند و با استفاده از فرمول زیر مساحت مثلث را نمایش می‌دهد:

$$p = \frac{(a+b+c)}{2} \quad , \quad s = \sqrt{p * (p-a) * (p-b) * (p-c)}$$

## ۶۰۰ برنامه ++C با حل آن‌ها ۱۰۷

a, b و c اضلاع مثلث و s مساحت آن است (هدف این برنامه آشنایی بیش‌تر با اعضای داده‌ای و تابعی کلاس و ایجاد سازنده‌های مختلف و استفاده از آن‌ها می‌باشد).



۹. برنامه‌ای که استفاده از کپی سازنده را نمایش می‌دهد.  
۱۰. برنامه‌ای که نوشتن کد برای سازنده کپی را نمایش می‌دهد (سازنده کپی فقط در هنگام تعریف کلاس به همراه مقداردهی اولیه فراخوانی می‌شود).

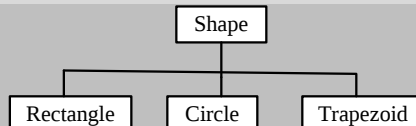
۱۱. برنامه‌ای که کلاسی به نام *Number* ایجاد می‌کند. این کلاس دارای اعضای زیر است:  
• عضو داده‌ای *Num* از نوع *static* است. این عضو تعداد نمونه‌های (اشیاء) ایجادشده را نگه‌داری می‌کند.  
• عضو تابعی *getNum()* از نوع *static* است. این عضو مقدار داده‌ای *Num* را برمی‌گرداند.  
• سازنده *Number*، به عضو داده‌ای *Num* (تعداد اشیاء تولیدشده) یک واحد اضافه می‌کند.  
• تابع مخرب *~Number*، یک واحد از عضو داده‌ای *Num* کم می‌کند.  
هدف این برنامه آشنایی با اعضای داده‌ای و تابعی *static* است. این برنامه تعداد نمونه‌های ایجادشده و موجود کلاس را می‌شمارد.

۱۲. برنامه‌ای که کلاسی به نام *Balance* ایجاد می‌کند. این کلاس برای برداشت و واریز مبلغ به حساب به کار می‌رود (هدف این برنامه آشنایی با پیاده‌سازی مجدد عملگرها است).



۱۳. برنامه‌ای که پیاده‌سازی سازنده‌ها و مخرب‌های کلاس پایه و مشتق را نشان می‌دهد.  
• ی a و b را به ترتیب از کلاس‌های A و B به ارث می‌برد.

۱۴. برنامه‌ای که اطلاعات اشکالی از قبیل مستطیل، دایره و ذوزنقه را دریافت می‌کند. اشکال موردنظر را ایجاد کرده، مساحت آن‌ها را نمایش می‌دهد. اشکال دارای سلسله‌مراتب وراثتی زیر هستند:



هدف این برنامه آشنایی با وراثت، سازنده‌های کلاس پایه و مشتق، متدهای مجازی (virtual) و تعریف مجدد متدها (override) می‌باشد.

۱۵. برنامه‌ای که عددهای تصادفی تا ۱۰۰۰۰ ایجاد کرده، نمایش می‌دهد.  
۱۶. برنامه‌ای که با استفاده از کلاسی با نوع الگو (Template) داده‌های با انواع مختلف را برای نمونه‌های متفاوت ایجاد می‌کند و نمایش می‌دهد.  
۱۷. برنامه‌ای که n را خوانده، n علامت \* (ستاره) را به‌طور تصادفی بر روی صفحه نمایش می‌دهد (هدف این برنامه آشنایی با سازنده و مخرب است).  
۱۸. کلاسی که طول و عرض خانه‌ای را خوانده، زمان لازم برای چمن‌زنی خانه را نمایش می‌دهد. سرعت ماشین چمن‌زنی ۱۴۵/۵ متر در دقیقه است.  
۱۹. کلاسی که وزن جسمی را به پوند خوانده، معادل گرم و کیلوگرم آن را چاپ می‌کند (هر پوند معادل ۰/۴۵۳۵۹۵ کیلوگرم و معادل ۴۵۳/۵۹۲۳۷ گرم است).

۲۰. طبق قضیه فیثاغورث، مجموع مربعات اضلاع قائمه یک مثلث قائم‌الزاویه برابر است با مربع وتر. چنانچه m و n داشته باشیم، با استفاده از فرمول زیر می‌توان مثلث قائم‌الزاویه را تشکیل داد.

کلاسی که  $m$ ,  $n$  را خوانده، مقادیر اضلاع را محاسبه و چاپ می کند.

$c = m^2 + n^2$  (وتر)  $b = 2 * m * n$  (ضلع دوم)  $a = m^2 - n^2$  (ضلع اول)

۲۱. برنامه‌ای که کلاسی به نام triangle ایجاد می کند. این کلاس صفاتی مانند base و height دارد که مقادیر اولیه آن‌ها یک است. تابع عضو (`input()`) اعضای base و height را می خواند. تابع عضو `calculate`، مساحت مثلث را محاسبه می کند و تابع `print`، مساحت مثلث را به خروجی می برد.

۲۲. برنامه‌ای که با تعریف یک کلاس  $n$  را می خواند و فاکتوریل  $n$  را چاپ می کند (این کلاس دو بار فراخوانی می گردد یکبار برای  $n = 0$  و بار دیگر  $n$  را می خواند).

۲۳. کلاسی که دو عدد ده‌دهی حداکثر ۵۰ رقمی را با یکدیگر جمع می نماید و نتیجه را چاپ می کند.

۲۴. کلاسی که مقداری را به فوت و اینچ می خواند و به متر تبدیل می کند. هر فوت برابر ۰٫۳۰۴۸ متر و هر اینچ ۱۲ متر است.

۲۵. میزان حقوق کارگران به صورت هفتگی محاسبه می شود. روش محاسبه به صورت زیر است:

- نام، ساعت کارکرد، تعداد فرزندان و میزان پرداختی به ازای هر ساعت کارکرد کارگران را دریافت می کند.
- کارکرد بیش تر از ۴۰ ساعت اضافه کار محسوب می شود و دستمزد آن ۱/۵ برابر دستمزد عادی محاسبه می شود.
- ۷٪ از حقوق کارگران به عنوان بیمه تأمین اجتماعی کسر می گردد.
- اگر کارگران بیش از ۲ فرزند داشته باشند، از حقوقشان ۲۰۰۰ تومان به عنوان بیمه درمان کسر می گردد.
- اگر حقوق کارگران بیش از ۳۵۰۰۰ تومان باشد، ۱۰٪ مازاد بر ۳۵۰۰۰ تومان به عنوان مالیات حقوق کسر می گردد.
- کلاسی که اطلاعات کارگران را دریافت می کند و حقوق خالص آن‌ها را نمایش می دهد. این برنامه با آرایه‌ای از کلاس‌ها پیاده سازی گردید.

۲۶. برنامه‌ای که کلاسی برای اعداد کسری با قابلیت‌های زیر را می نویسد:

- تابع عضوی دارد که از مخرج صفر جلوگیری می کند، کسرها را ساده می نماید و صورت کسر منفی را نمی پذیرد.
- اعمال جمع، تفریق، ضرب و تقسیم را برای این کلاس مجدداً تعریف می کند.
- عملگرهای رابطه‌ای و تساوی را برای این کلاس مجدداً تعریف می کند.
- عمل جمع، تفریق، ضرب و تقسیم کسرها را انجام می دهد و خروجی را به صورت کسری و اعشاری نمایش می دهد.

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۰۹

۲۷. الحاق رشته‌ها به عملوند نیاز دارد (رشته‌هایی که باید با هم الحاق شوند). بدین ترتیب نتیجه حاصل، در یکی از این دو عملوند قرار می‌گیرد، مانند  $s^3 = s^1 + s^2$ . تابع `operator +()` را طوری می‌نویسد که بتواند به این صورت عمل کند:

$$s^1 = s^2 + s^3$$

۲۸. برنامه‌ای که کلاسی به نام `complex` برای انجام محاسبات روی اعداد موهومی تشکیل می‌دهد. سپس، عملگرهای `+`، `-`، `*`، و `/` را برای اعداد موهومی مجدداً تعریف می‌کند. اعداد موهومی به صورت  $x + yi$  هستند که در آن  $i$  برابر با  $\sqrt{-1}$  است.

۲۹. برنامه‌ای که کلاسی به نام `integerest` ایجاد می‌کند. هر شیء از این کلاس می‌تواند اعدادی صحیح در بازه ۰ تا ۱۰۰ را بپذیرد. مجموعه‌ای به صورت آرایه‌ای از یک و صفر نمایش داده می‌شود. اگر عدد صحیح  $i$  در مجموعه باشد، `a[i]` برابر با یک است. عنصر `a[i]` در صورتی صفر است که  $i$  در مجموعه نباشد. سازنده، مجموعه را طوری مقدار می‌دهد که تمام عناصر آرایه صفر باشند. این کلاس دارای اعضای زیر است:

- تابع عضوی که اشتراک دو مجموعه را می‌یابد.
- تابع عضوی که اجتماع دو مجموعه را می‌یابد.
- تابع عضوی که عنصری مثل  $k$  را در آرایه درج می‌کند، به طوری که `a[k]` برابر با صفر باشد.
- تابع عضوی که عنصری از مجموعه را حذف می‌کند.
- تابع عضوی که عناصر مجموعه را به عنوان اعداد صحیح به خروجی می‌برد.
- تابع عضوی که مشخص می‌کند آیا دو مجموعه برابرند یا خیر؟

۳۰. برنامه‌ای که کلاس `point` را برای نقاط سه‌بعدی ( $x, y, z$ ) پیاده‌سازی می‌کند. در این کلاس اعمال زیر انجام می‌شود:

- یک سازنده دارد که مقادیر پیش فرض را به اعضای کلاس نسبت می‌دهد.
- سازنده دیگری دارد که مقادیر اعضا را به عنوان پارامتر گرفته به اعضاء تخصیص می‌دهد.
- متدی به نام `negative` دارد که کلیه نقاط را منفی می‌نماید.
- متدی به نام `norm` دارد که فاصله از مبدأ  $(0, 0, 0)$  را حساب می‌کند.
- یک متد `print` دارد که مختصات  $(x, y, z)$  را چاپ می‌کند.
- عملگرهای حسابی `+`، `=` و `++` را برای این کلاس مجدداً تعریف می‌کند.
- کلاس `point`: این کلاس دارای اعضای زیر است:
  - عضو `point`، متدهای سازنده کلاس هستند.
  - عضو `operator=`، عملگر `=` را مجدداً پیاده‌سازی می‌کند.
  - عضو `operator++`، عملگر `++` را مجدداً تعریف می‌کند.
  - عضو `negative`، متدی است که اعضای داده‌ای را منفی می‌نماید.
  - عضو `norm`، متدی است که فاصله نقطه تا نقطه  $(0, 0, 0)$  را نمایش می‌دهد.
  - عضو `print`، متدی است که محتویات نقطه را نمایش می‌دهد.
  - اعضای `x, y, z` مختصات نقطه را نگه‌داری می‌کنند.

۳۱. برنامه‌ای که کلاسی به نام `time` را پیاده‌سازی می‌کند که دارای امکانات زیر است:

۱. اعضای کلاس ساعت، دقیقه و ثانیه است.
۲. متدهای دستیابی به اعضای کلاس را دارد.
۳. متدی به نام `advance` دارد که زمان را به جلو می‌برد.

۴. متدی به نام reset دارد که زمان جدیدی به اعضای کلاس تخصیص می‌دهد.
۵. عملگر - را برای این کلاس مجدداً تعریف می‌کند.
۶. عملگرهای = و == را برای این کلاس مجدداً پیاده‌سازی می‌کند.
۷. متدی به نام print دارد که محتویات کلاس را چاپ می‌کند.
۸. متدی به نام printAmPm دارد که زمان را به صورت am یا pm چاپ می‌نماید.

**پروژه برنامه‌نویسی ۱ :** برنامه‌ای که کلاسی برای چندجمله‌ای ایجاد می‌کند. نمایش داخلی چندجمله‌ای (polynomial) آرایه‌ای از جملات است. هر جمله حاوی یک ضریب و یک توان است. به عنوان مثال، چند جمله  $۲x^۴$  دارای ضریب ۲ و توان ۴ است. این برنامه، توابعی برای دریافت چندجمله‌ای دارد. کلاس دارای قابلیت‌های زیر است:

- عملگر + را برای جمع دو چندجمله‌ای مجدداً تعریف می‌کند.
- عملگر - را برای تفریق دو چندجمله‌ای مجدداً تعریف می‌کند.
- عملگر انتساب را برای انتساب یک چندجمله‌ای به چندجمله‌ای دیگر تعریف می‌کند.
- عملگر \* را برای ضرب دو چندجمله‌ای مجدداً تعریف می‌کند.
- عملگرهای +=، -= و == را برای چندجمله‌ای مجدداً تعریف می‌کند.

## ورودی و خروجی فایل

۱. برنامه‌ای که نام یک فایل متنی را خوانده، سپس نام چند کتاب را می‌خواند و در آن فایل ذخیره می‌کند (چنانچه کاربر در هنگام ورود نام کتاب چیزی وارد نکند و کلید Enter را فشار دهد، برنامه خاتمه می‌یابد).

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    char bookName[50], fileName[50];
    cout << "Enter a file name:";
    cin.getline(fileName, 50);
    ofstream f;
    f.open(fileName);
    for(;;)
    {
        cout << "Enter book name:";
        cin.getline(bookName, 50);
        if(bookName[0] == '\0') break;
        f << bookName << endl;
    }
    f.close();
    return 0;
}
```

در این برنامه اعمال زیر انجام می‌شود:

متغیرهای `fileName` (نام فایل را نگهداری می‌کند) و `bookName` (نام کتاب را برای هر بار نگهداری می‌کند) را تعریف می‌کند.

با پیغام مناسب نام فایل را خوانده، در `fileName` قرار می‌دهد.

با پیغام مناسب نام فایل را خوانده، در `fileName` قرار می‌دهد.

اشاره گر `f` را با نوع `ofstream` تعریف می‌کند تا بتواند در آن بنویسد و سپس فایلی که نام آن در `fileName` است، باز می‌کند.

در یک حلقه تکرار بی‌نهایت اعمال زیر را انجام می‌دهد:

۱. با پیغام مناسب نام کتاب را می‌خواند.

حلقه خاتمه می‌یابد. `break`. ۲. اگر نام کتاب خالی باشد، با دستور

۳. نام کتاب را در فایل می‌نویسد.

فایل را می‌بندد.

```
D:\BookCSharp\cppnewPractice\CodeBlack\7\1\bin\Debug\1.exe
Enter a file name:D:\Data\c1.txt
Enter book name:C++
Enter book name:Java
Enter book name:Python
Enter book name:C#
Enter book name:

Process returned 0 (0x0)   execution time : 36.767 s
Press any key to continue.
```

۲. برنامه‌ای که نام یک فایل متنی را خوانده، اطلاعات آن فایل را بر روی صفحه نمایش نشان می‌دهد.

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    char line[80], fileName[50];
    cout << "Enter a file name:";
    cin.getline(fileName, 50);
    ifstream f;
    f.open(fileName);
    do
    {
        f >> line;
        if( f.eof() ) break;
        cout << line << endl;
    } while(!f.eof());
    f.close();
    return 0;
}
```

در این برنامه اعمال زیر انجام می‌شود:

- متغیرهای line (هر خط فایل) و fileName (نام فایل ورودی) را می‌خواند.
- با یک پیغام مناسب نام فایل ورودی را خوانده، در fileName قرار می‌دهد.
- اشاره گر f را با نوع ifstream تعریف می‌کند تا بتواند فایل را باز کرده، اشاره گر آن را در این متغیر قرار دهد.
- در یک حلقه تکرار اعمال زیر را انجام می‌دهد:
  ۱. یک خط از فایل f را می‌خواند.
  ۲. اگر به انتهای فایل رسیده باشد (if(f.eof())), با دستور break حلقه را خاتمه می‌دهد.
  ۳. اطلاعات خط خوانده شده (line) را در خروجی نمایش می‌دهد.
- فایل f را می‌بندد.

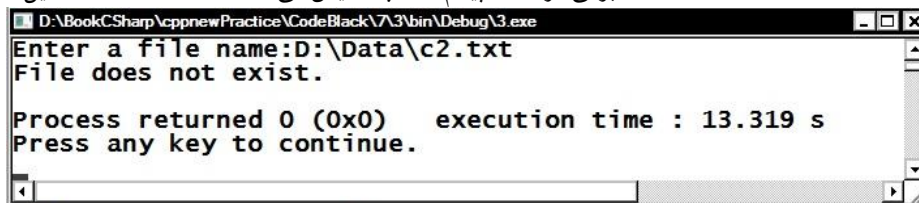
```
D:\BookCSharp\cppnewPractice\CodeBlack\7\2\bin\Debug\2.exe
Enter a file name:D:\Data\c1.txt
C++
Java
Python
C#

Process returned 0 (0x0)   execution time : 68.873 s
Press any key to continue.
```

۳. برنامه‌ای که نام یک فایل متنی را خوانده، چنانچه فایل وجود نداشته باشد، پیغام مناسب نمایش می‌دهد، وگرنه اطلاعات فایل خوانده‌شده را بر روی صفحه نمایش نشان می‌دهد.

```
#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    char line[80], fileName[50];
    cout << "Enter a file name:";
    cin.getline(fileName, 50);
    ifstream f;
    f.open(fileName);
    if (f.fail())
    {
        cout << "File does not exist." << endl;
        return 0;
    }
    do
    {
        f >> line;
        if( f.eof() ) break;
        cout << line << endl;
    } while(!f.eof());
    f.close();
    return 0;
}
```

این برنامه مانند برنامه ۲ عمل می‌کند، با این تفاوت که چنانچه فایل وجود نداشته باشد، بعد از باز کردن را برمی‌گرداند و پیغام مناسب نمایش می‌دهد. true مقدار f.fail() قابل متد



۴. برنامه‌ای که ابتدا n را خوانده، سپس n نام و سن را می‌خواند و در فایل "out.dat" در پوشه data درایو D می‌نویسد. در پایان، اطلاعات فایل "out.dat" را نمایش می‌دهد.

```
#include <fstream>
#include <iostream>
using namespace std;
int main ()
{
    char data[100];
    int n;
    cout << "Enter n:";
    cin >> n;
    cin.ignore();
    ofstream outfile;
    outfile.open("D:\\Data\\out.dat");
    cout << "Writing to the file" << endl;
    for (int i = 1; i <= n; i++)
    {
        cout << "Enter " << i << "th name: ";
```

```

        cin.getline(data, 100);
        outfile << data << endl;
        cout << "Enter " << i << "th age: ";
        cin >> data;
        cin.ignore();
        outfile << data << endl;
    }
    outfile.close();
    ifstream infile;
    infile.open("D:\\Data\\out.dat");
    cout << "Reading from the file" << endl;
    for(;;)
    {
        infile >> data;
        if(infile.eof()) break;
        cout << data << endl;
        infile >> data;
        if(infile.eof()) break;
        cout << data << endl;
        cout << "======" << endl;
    }
    infile.close();
    return 0;
}

```

در این برنامه اعمال زیر انجام می‌شود:

- متغیرهای data (آرایه‌ای از کاراکترها) و n (تعداد نام و سن) را تعریف می‌کند.
- با پیغام مناسب n را خوانده و enter را نادیده می‌گیرد تا داده بعدی را بخواند.
- فایل "out.dat" را به صورت نوشتنی باز می‌کند.
- در یک حلقه تکرار از ۱ تا n نام و n سن را می‌خواند و در فایل outFile می‌نویسد.
- فایل outFile را می‌بندد.
- فایل "out.dat" را به صورت خواندنی باز می‌کند و اشاره گر آن در inFile قرار می‌دهد.
- در یک حلقه تکرار بی‌نهایت تا زمانی که به انتهای فایل نرسیده باشد، از inFile می‌خواند و در خروجی می‌نویسد.
- فایل inFile را می‌بندد.

```

D:\BookCSharp\cppnewPractice\CodeBlack\7\4\bin\Debug\4.exe
Enter n:3
Writing to the file
Enter 1th name: Ahmadi
Enter 1th age: 12
Enter 2th name: Abbasnejad
Enter 2th age: 18
Enter 3th name: Rezvani
Enter 3th age: 22
Reading from the file
Ahmadi
12
=====
Abbasnejad
18
=====
Rezvani
22
=====
Process returned 0 (0x0)   execution time : 35.486 s
Press any key to continue.

```

۵. برنامه‌ای که نام فایلی را خوانده، اندازه فایل را نمایش می‌دهد.

```
#include <iostream>
#include <fstream>
using namespace std;
int main ()
{
    long begin,end;
    char fileName[50];
    cout << "Enter a file name:";
    cin.getline(fileName, 50);
    ifstream myfile (fileName);
    begin = myfile.tellg();
    myfile.seekg (0, ios::end);
    end = myfile.tellg();
    myfile.close();
    cout << "size is: " << (end-begin) << " bytes.\n";
    return 0;
}
```

در این برنامه اعمال زیر انجام می‌شود:

متغیرها begin (شماره کاراکتر شروع فایل) و end (شماره کاراکتر انتهای فایل) را با نوع long تعریف می‌کند.

متغیر fileName را تعریف کرده، با پیغام مناسبی نام فایل را خوانده، در این متغیر قرار می‌دهد.

فایلی که نام آن در fileName است را باز کرده، در اشاره‌گر myfile قرار می‌دهد.

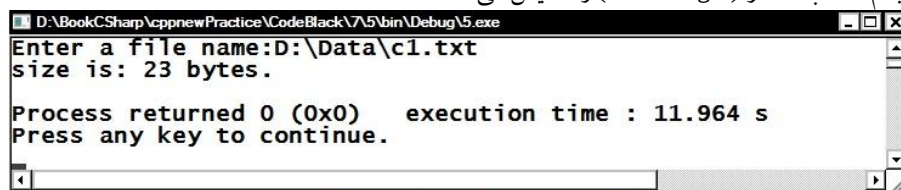
شماره بایت فعلی فایل را با تابع tellg() در begin قرار می‌دهد.

با تابع seekg(0, ios::end) (دستور myfile.seekg(0, ios::end)).

با تابع tellg() موقعیت فعلی فایل را در end قرار می‌دهد.

فایل را می‌بندد.

با پیغام مناسب مقدار (end – begin) را نمایش می‌دهد.

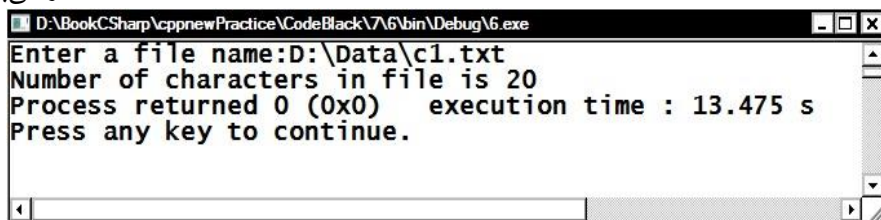


۶. برنامه‌ای که نام یک فایل متنی را خوانده، تعداد کاراکترهای آن را شمارش می‌کند.

```
#include <iostream>
#include <fstream>
using namespace std;
int main ()
{
    char fileName[50];
    cout << "Enter a file name:";
    cin.getline(fileName, 50);
    ifstream myfile (fileName);
    char ch;
    int count=0;
    while(!myfile.eof())
```

```
{
    myfile.get(ch);
    count++;
}
cout<<"Number of characters in file is " << count;
myfile.close();
return 0;
}
```

را تعریف کرده و با پیغام مناسب نام فایل را می‌خواند و در این `fileName` در این برنامه ابتدا، متغیر قرار می‌دهد. در ادامه `myfile` را باز کرده و اشاره گر آن را در `fileName` متغیر قرار می‌دهد. سپس فایل (تعداد کاراکترهای فایل) را تعریف می‌کند و در یک حلقه `count` (هر کاراکتر فایل) و `ch` متغیرهای اضافه می‌کند. `count` تکرار کاراکتر به کاراکتر از فایل می‌خواند و با خواندن هر کاراکتر یک واحد به `myfile` را نمایش داده فایل (`count` این عمل را تا انتهای فایل انجام می‌دهد و در پایان، با پیغام مناسب را می‌بندد.



۷. برنامه‌ای که نام یک فایل متنی را خوانده و تعداد کلمات آن را شمارش کرده و نمایش می‌دهد.

۸. برنامه‌ای که نام یک فایل متنی را خوانده، تعداد خطوط آن را شمارش کرده و نمایش می‌دهد.

۹. برنامه‌ای که نام دو فایل را از ورودی خوانده، اطلاعات فایل اول را در فایل دوم کپی می‌کند.

۱۰. برنامه‌ای که اطلاعات یک دانشجو از قبیل شماره دانشجویی، نام، معدل را به فایل `student.dat` اضافه می‌کند. این برنامه امکاناتی از قبیل اضافه کردن دانشجوی جدید، حذف دانشجوی موجود، جستجوی دانشجو، تغییر اطلاعات دانشجو و نمایش کلیه اطلاعات دانشجویان موجود در فایل را دارد.

۱۱. برنامه‌ای که ۵ عدد اعشاری را خوانده، در فایل‌های متنی `1.txt` و فایل باینری `1.dat` می‌نویسد.

۱۲. برنامه‌ای که نام چند فایل را از ورودی خوانده، اطلاعات فایل‌ها را بر روی صفحه نمایش نشان می‌دهد (اگر کاربر به جای نام فایل، خالی وارد کند `(fileName[0] == '\0')`، برنامه خاتمه می‌یابد).

۱۳. برنامه‌ای که نام فایلی را خوانده، سپس یک کلمه را می‌خواند و خطوطی از فایل که کلمه خوانده شده در آن‌ها باشد، آن خطوط را نمایش می‌دهد.

۱۴. برنامه‌ای که نام یک فایل متنی را خوانده، تعداد کاراکتر، تعداد خطوط، تعداد جاهای خالی و تعداد ارقام موجود در آن را شمارش کرده و نمایش می‌دهد.

۱۵. برنامه‌ای که نام دو فایل متنی را خوانده، یک سطر از فایل اول و سپس یک سطر دیگر از فایل دوم بر روی صفحه نمایش نشان می‌دهد و این روند تا انتهای فایل اول و دوم ادامه می‌دهد.

۱۶. برنامه‌ای که نام دو فایل را خوانده، فایل اول را به فایل دوم انتقال می‌دهد.

## الکترونیکی (فصل اول)

### پیوست

➤ برنامه‌ای که نرخ حقوق به ازای هر ساعت و تعداد ساعت کارکرد یک کارمند را خوانده، میزان حقوق او را محاسبه می‌کند و نمایش می‌دهد (از حقوق ۱۰ درصد به عنوان مالیات کسر می‌شود. حقوق خالص برابر با حقوق منهای مالیات است).

➤ برنامه‌ای که مختصات دو نقطه را دریافت کرده، فاصله بین آن دو نقطه را محاسبه می‌نماید و نمایش می‌دهد.

$$\text{فاصله بین دو نقطه} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

➤ زمینی به ابعاد  $16 \times 13$  وجود دارد. موزاییک‌هایی به ابعاد  $3 \times 2$ ، برای فرش کردن زمین نیاز است. اولاً تعیین کنید چند موزاییک نیاز است. آیا قسمتی از زمین می‌ماند که با موزاییک فرش نشود؟

➤ برنامه‌ای که نقطه تقاطع دو خط  $y = 3x + c$  و  $y = ax + b$  را نمایش می‌دهد (برنامه a، b و c را از ورودی می‌خواند). مختصات نقطه تقاطع به صورت زیر به دست می‌آید:

$$x = (b - c) / (3 - a) \quad y = 3 * x + c$$

➤ برنامه‌ای که a و b را خوانده، مقدار  $y = b * \sin(a)$  را نمایش می‌دهد.

➤ برنامه‌ای که Four را نمایش داده، چهار بار بوق کامپیوتر را به صدا در می‌آورد، سپس به خط بعد می‌رود، Five را نمایش داده، ۵ بار بوق کامپیوتر را به صدا در می‌آورد.

➤ برنامه‌ای که عدد یک و یک کاراکتر قلب، سپس عدد ۲، دو کاراکتر قلب و عدد ۳، سه کاراکتر قلب، ۴ و کاراکتر قلب را نمایش می‌دهد.

➤ برنامه‌ای که تعداد بایت‌هایی را نمایش می‌دهد که متغیرهای با انواع long، int، char، float و double اشغال می‌کنند.

- برنامه‌ای که یک کاراکتر را خوانده، کد اسکی آن را نمایش می‌دهد (در این برنامه متغیر ch کاراکتر خوانده‌شده را نگه‌داری می‌کند).
- برنامه‌ای که عددی را خوانده، با استفاده از عملگر یا انحصاری تشخیص می‌دهد، عدد خوانده‌شده زوج است یا فرد؟
- برنامه‌ای که ابتدا عددی را خوانده، سپس n را می‌خواند و مقدار n امین بیت عدد اول خوانده شده را نمایش می‌دهد.
- برنامه‌ای که ابتدا عددی را خواند، سپس n را می‌خواند و n امین بیت عدد را برابر ۱ (یک) قرار می‌دهد.
- برنامه‌ای که ابتدا عددی را خوانده، سپس n را می‌خواند و n امین بیت عدد خوانده‌شده را صفر می‌نماید. به عنوان مثال، شکل زیر ۴ امین بیت عدد ۵۵ را صفر می‌نماید.

|          |   |                                 |
|----------|---|---------------------------------|
| 55       | ← | num(in decimal)                 |
| 4        | ← | n(in decimal)                   |
| 00000001 | ← | 1(in binary)                    |
| 00010000 | ← | $1 \ll n$                       |
| 11101111 | ← | $\sim(1 \ll n)$                 |
| 00110111 | ← | num(in binary)                  |
| 00100111 | ← | $\text{Num} \& (\sim(1 \ll n))$ |

- برنامه‌ای که ابتدا عددی را خوانده، سپس عدد n را می‌خواند و n امین بیت عدد خوانده شده را معکوس می‌کند. یعنی، چنانچه این بیت یک باشد، به صفر و اگر صفر باشد، به یک تبدیل می‌نماید.

- برنامه‌ای که دو عدد را خوانده، با استفاده از عملگر یا انحصاری بیتی آن‌ها را باهم تعویض می‌نماید و نمایش می‌دهد. مراحل انجام این کار در شکل زیر آمده است:

|       |          |     |          |     |          |
|-------|----------|-----|----------|-----|----------|
| a     | 00011010 | b   | 01000001 | a   | 00011010 |
| b ^   | 01000001 | x ^ | 01011011 | x ^ | 01011011 |
| <hr/> |          |     |          |     |          |
| x     | 01011011 | a   | 00011010 | b   | 01000001 |

- برنامه‌ای که سه عدد را به عنوان قاعده کوچک، قاعده بزرگ و ارتفاع یک دوزنقه خوانده، مساحت دوزنقه را نمایش می‌دهد. مساحت دوزنقه برابر با:  
$$\left( \text{قاعده بزرگ} + \text{قاعده کوچک} \right) * \text{ارتفاع} * \frac{1}{2}$$
- برنامه‌ای که عددی را خوانده، تمام بیت‌های عدد خوانده شده را معکوس می‌کند. یعنی، چنانچه بیت یک باشد به صفر و اگر صفر باشد، به یک تبدیل می‌نماید.

## پیوست

## الکترونیکی (فصل دوم)

➤ برنامه‌ای که عددی را خوانده، تعیین می‌کند که مربع کامل است یا خیر؟ عددی مربع کامل است که مجموع تعدادی اعداد فرد پشت سر هم از ۱ و بعد آن برابر آن عدد شود. به عنوان مثال، اعداد زیر مربع کامل هستند:

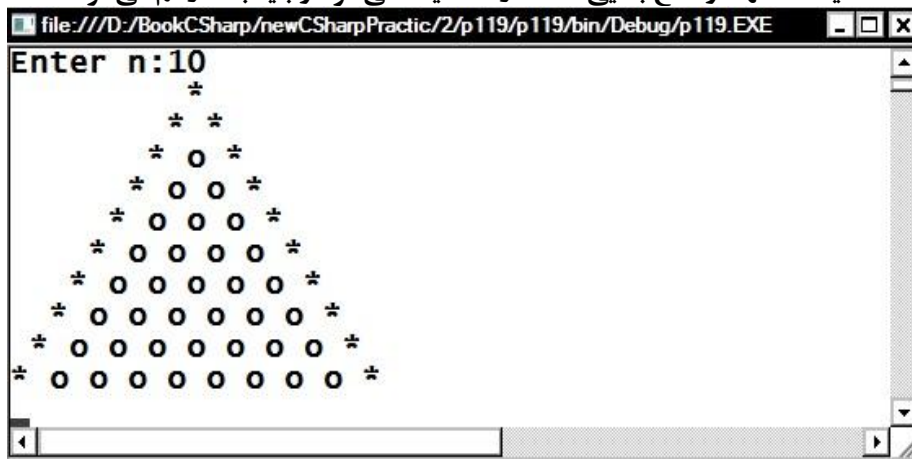
$$\begin{aligned}\sqrt{9} &= 1+3+5 & \sqrt{1} &= 1 \\ \sqrt{16} &= 1+3+5+7 & \sqrt{4} &= 1+3\end{aligned}$$

➤ برنامه‌ای که  $n$  و  $x$  را خوانده، حاصل  $n$  جمله سری زیر را نمایش می‌دهد:

$$\frac{x + \frac{x}{2}}{1!} + \frac{x^3 - \frac{x}{4}}{2!} + \frac{x^5 + \frac{x}{8}}{3!} + \dots$$

➤ برنامه‌ای که عددی را خوانده، ارقام زوج آن را نمایش می‌دهد. به عنوان مثال، برای عدد ۱۳۵۲۷۴، عدد ۲۴ را نمایش خواهد داد.

➤ برنامه‌ای که  $n$  را خوانده، یک مثلث به شکل زیر با  $n$  سطر در خروجی نمایش می‌دهد. دقت کنید که تنها دو ضلع بالایی مثلث از \* کشیده می‌شود و بقیه با O رسم می‌گردند.



➤ ۱۲۰. برنامه‌ای که یک عدد را خوانده، در خروجی عددی را نمایش دهد که هر رقم آن یک واحد اضافه شده است. به عنوان مثال، اگر کاربر عدد ۴۵۹۳۴ را وارد کند، در خروجی ۵۶۰۴۵ چاپ شود (رقم ۹ به صفر تبدیل گردد).

➤ ۱۲۱. برای محاسبه  $\pi$  می‌توان از رابطه زیر استفاده کرد:

$$\frac{\pi}{2} = \frac{2}{1} + \frac{2}{3} + \frac{4}{3} + \frac{4}{5} + \frac{6}{5} + \frac{6}{7} + \dots$$

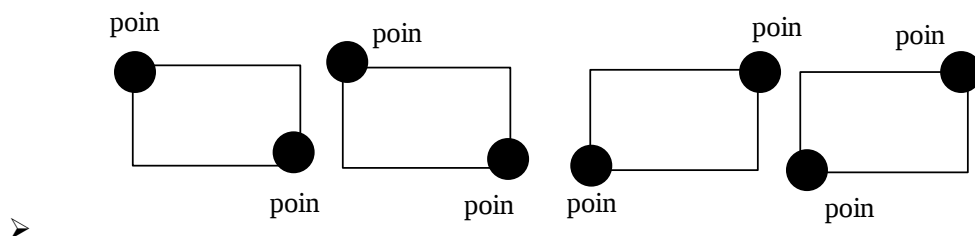
## ۶۰۰ برنامه C++ با حل آن‌ها ۱۲۱

➤ برنامه‌ای که تعداد جملات سری فوق (n) را به عنوان یک عدد صحیح دریافت کرده و مقدار  $\pi$  را با استفاده از این رابطه محاسبه کرده و نمایش می‌دهد.

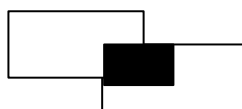
➤ برنامه‌ای که مختصات سه نقطه را از ورودی می‌گیرد و تعیین می‌کند که آن سه نقطه هم خط هستند یا خیر؟

➤ اگر سه نقطه هم خط بودند، خروجی برنامه، یک معادله به صورت  $ax + by + c = 0$  خواهد بود، وگرنه خروجی "No" خواهد بود. ورودی به صورت شش عدد اعشاری خواهد بود که مختصات سه نقطه A و B و C را بیان می‌کند. سه نقطه  $A(x_1, y_1)$ ،  $B(x_2, y_2)$  و  $C(x_3, y_3)$  را هم خط گویند، هرگاه شیب دو خط AB و BC باهم برابر باشند. شیب دو خط AB با X فرمول  $m = \frac{y_2 - y_1}{x_2 - x_1}$  قابل محاسبه است. همچنین، معادله خط AB به صورت  $y - y_1 = m(x - x_1)$  نوشته می‌شود.

➤ برنامه‌ای که مختصات دو مستطیل را از ورودی گرفته و مساحت ناحیه‌ی مشترک بین دو مستطیل را حساب می‌نماید. به ازاء هر مستطیل دو نقطه (جمعاً چهار عدد اعشاری) باید گرفته شود. توجه کنید که هر مستطیل با استفاده از دو نقطه می‌تواند به یکی از چهار فرم زیر نمایش داده شود.



➤



➤ به عنوان نمونه، هدف این سؤال پیدا کردن ناحیه پررنگ بین دو مستطیل زیر می‌باشد. البته این احتمال وجود دارد که ناحیه مشترک وجود نداشته باشد که در این صورت مساحت ناحیه مشترک صفر خواهد بود.

➤ برای هر کسر به صورت  $1/k$  (k عدد صحیح بزرگ‌تر از صفر و کوچک‌تر از ۱۰۰۰)، دو

$$\frac{1}{k} = \frac{1}{x} + \frac{1}{y}$$

عدد صحیح x و y وجود دارد که عبارت مقابل برقرار باشد:

برنامه‌ای که مقدار  $k$  را از دریافت می‌کند و تمام  $x$  و  $y$ هایی که در این عبارت صدق می‌کنند را نمایش می‌دهد. توجه کنید اگر  $k$  در بازه درست نباشد، برنامه باید در خروجی هشدار خطا دهد.

نمونه ورودی: ۱۲ خروجی

$$\begin{array}{lll} \frac{1}{12} = \frac{1}{13} + \frac{1}{156} & \frac{1}{12} = \frac{1}{14} + \frac{1}{84} & \frac{1}{12} = \frac{1}{15} + \frac{1}{60} \\ \frac{1}{12} = \frac{1}{16} + \frac{1}{48} & \frac{1}{12} = \frac{1}{18} + \frac{1}{36} & \frac{1}{12} = \frac{1}{20} + \frac{1}{30} \\ \frac{1}{12} = \frac{1}{24} + \frac{1}{24} & \frac{1}{12} = \frac{1}{21} + \frac{1}{28} & \end{array}$$

➤ برنامه‌ای که به روش تنصیف معادله زیر را تا چهار رقم اعشار حل می‌کند.

$$x + e^x = 5 \quad F(x) = x + e^x - 5 = 0$$

برای حل این معادله دو تخمین اولیه نیاز داریم. یکی عددی که به ازای آن مقدار معادله مثبت باشد ( $x_p$ ) و دیگری که مقدار منفی به معادله بدهد ( $x_n$ ). تخمین جدید در هر مرحله ( $x_m$ ) میانگین دو تخمین قبل است.

➤ برنامه‌ای که  $n$  را خوانده، حاصل  $n$  جمله سری زیر را نمایش می‌دهد.

$$1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots$$

➤ برنامه‌ای که همه حروف الفبای انگلیسی را به ترتیب چاپ می‌کند به‌طوری که هر حرف بزرگ، مشابه کوچک آن هم نمایش داده شود.

➤ برنامه‌ای که تعدادی عدد که به‌صورت مرتب‌شده صعودی وارد می‌شوند، را از ورودی خوانده، حاصل جمع آن‌ها را نمایش می‌دهد. اگر عدد بعدی از عدد قبلی کمتر وارد شود، برنامه خاتمه می‌یابد.

$$\circ \quad y = 3 * t * t + 5;$$

$$y = -3 * t * t + 5;$$

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۲۳

➤ برنامه‌ای که x و n را خوانده، حاصل n جمله عبارت زیر را نمایش می‌دهد:

$$\frac{-x^1}{1+2} + \frac{x^3}{3+4} - \frac{x^5}{5+6} + \dots$$

➤ برنامه‌ای که x و n را خوانده، حاصل n جمله عبارت زیر را حساب می‌کند:

$$-\frac{x^2}{1*3} + \frac{x^4}{3*5} - \frac{x^6}{5*7} + \dots$$

➤ برنامه‌ای که x و n را خوانده، حاصل n جمله سری زیر را محاسبه کرده، نمایش می‌دهد.

$$X / 3 + x^2 / (2 * 3^2) + x^2 / (3 * 3^2) + \dots$$

➤ برنامه‌ای که x و n را خوانده، حاصل n جمله سری زیر را محاسبه کرده و نمایش می‌دهد:

$$\frac{2x^1}{1+2} - \frac{4x^2}{2+4} + \frac{8x^3}{3+8} - \dots$$

➤ برنامه‌ای که x و n را خوانده، حاصل n جمله سری زیر را محاسبه کرده، نمایش می‌دهد:

$$x + \frac{x^2}{5} + \frac{x^3}{25} + \frac{x^4}{125} + \dots$$

➤ برنامه‌ای که ابتدا n را خوانده، سپس به تعداد n نقطه را به صورت اعشاری (x, y) می‌خواند و معادله  $y = mx + b$  را به دست آورد. m و b صورت‌های زیر محاسبه می‌شوند:

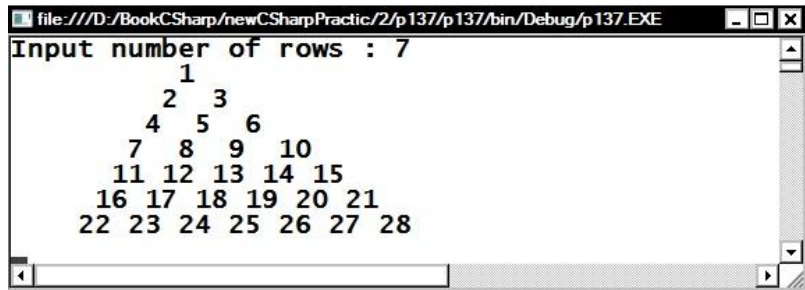
$$m = \frac{(\sum xy) - \bar{y}(\sum x)}{(\sum xx) - \bar{x}(\sum x)} \quad b = \bar{y} - m\bar{x}$$

$\bar{X}$  و  $\bar{Y}$  به ترتیب میانگین Y و X هستند. در این برنامه متغیرهای n (تعداد نقاط)، X و Y (مختصات هر نقطه در هر مرحله)، sumX (مجموع مقادیر x)، sumY (مجموع مقادیر y)، sumXY (مجموع  $x*y$ )، meanX (میانگین x)، meanY (میانگین y)، m (ضریب خط)، b (مقدار ثابت خط) و i (شمارنده از ۱ تا n) را داریم.

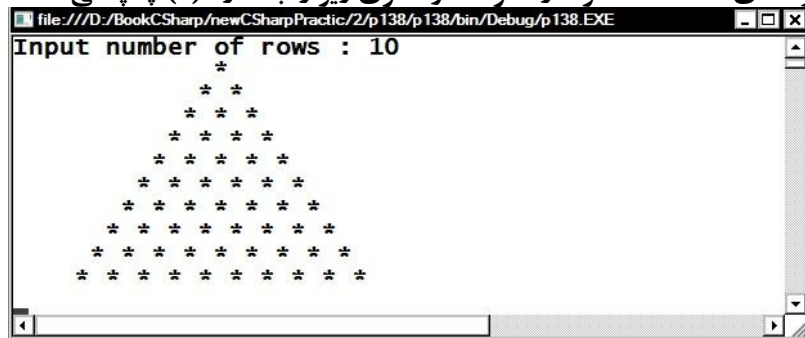
➤ برنامه‌ای که تعداد سطرها را خوانده و حاصل خروجی زیر را تولید می‌کند (برای تعداد سرهای ۶ خروجی زیر را نمایش می‌دهد):

|    |    |    |    |    |    |
|----|----|----|----|----|----|
| 1  |    |    |    |    |    |
| 2  | 3  |    |    |    |    |
| 4  | 5  | 6  |    |    |    |
| 7  | 8  | 9  | 10 |    |    |
| 11 | 12 | 13 | 14 | 15 |    |
| 16 | 17 | 18 | 19 | 20 | 21 |

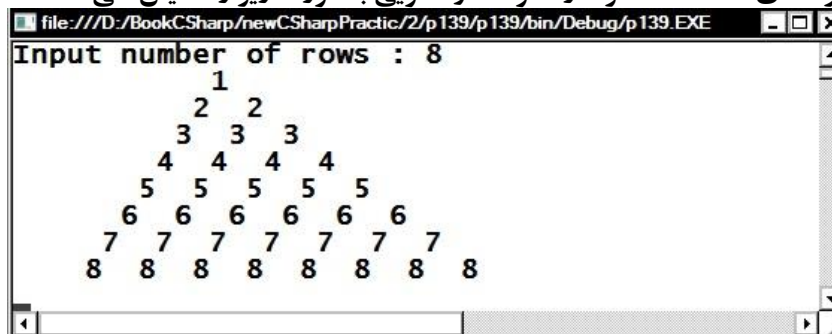
➤ برنامه‌ای که تعداد سطرها را خوانده و خروجی را با الگوی زیر نمایش می‌دهد



➤ برنامه‌ای که تعداد سطرها را خوانده و الگوی زیر را با ستاره (\*) چاپ می‌کند:



➤ برنامه‌ای که تعداد سطرها را خوانده و الگویی به صورت زیر را نمایش می‌دهد:



➤ برنامه‌ای که  $n$  و  $k$  را از ورودی خوانده و سپس  $n$  جمله سری زیر را چاپ می‌کند:

➤ 
$$K + KK + KKK + \dots + KKK \dots K$$
  
 $K$  عددی بین ۱ تا ۹ است. در پایان، مجموع این سری را نمایش می‌دهد.

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل عبارت زیر را محاسبه می‌کند:

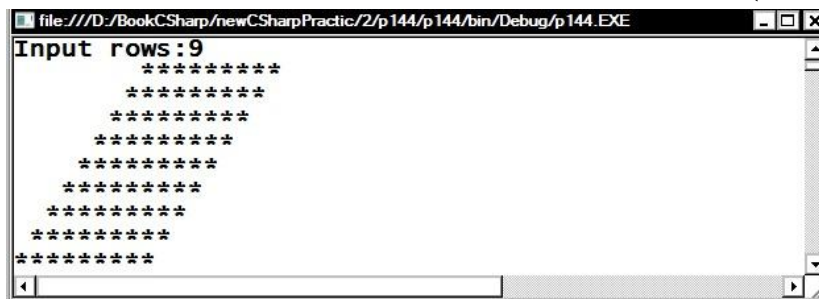
➤ 
$$\frac{x^n n^x x^{(n+n)} n^{(x+x)}}{x! n! (x+n)!} \sum_{i=1}^n \frac{x^n n^x (x+n)^i}{(x+2i)! (n+2i)!}$$

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۲۵

➤ . برنامه‌ای که تعیین می‌کند کم‌ارزش‌ترین بیت ۱ است یا ۰ (کم‌ارزش‌ترین بیت، سمت راست‌ترین بیت می‌باشد). به عنوان مثال، برای عدد ۱۵ (۰۰۰۰۱۱۱۱)، سمت راست-ترین بیت ۱ است، ولی برای عدد ۱۲ (۰۰۰۰۱۱۰۰) سمت راست‌ترین بیت ۰ است. اگر این اعداد را با ۱ (یک)، و بیتی نماییم، چنانچه نتیجه صفر شود، سمت راست‌ترین بیت صفر، ولی اگر نتیجه یک شود سمت راست‌ترین بیت یک است:

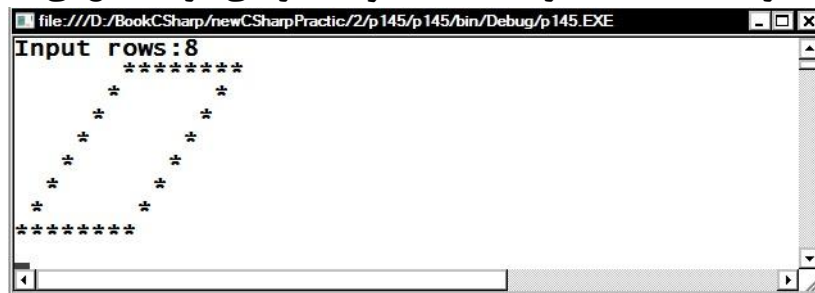
➤ . برنامه‌ای که تعیین می‌کند با ارزش‌ترین بیت یک عدد یک (۱) است یا صفر (۰)؟ با ارزش‌ترین بیت عدد، سمت چپ‌ترین بیت آن است.

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی مانند زیر را نمایش می‌دهد (rows=۹).



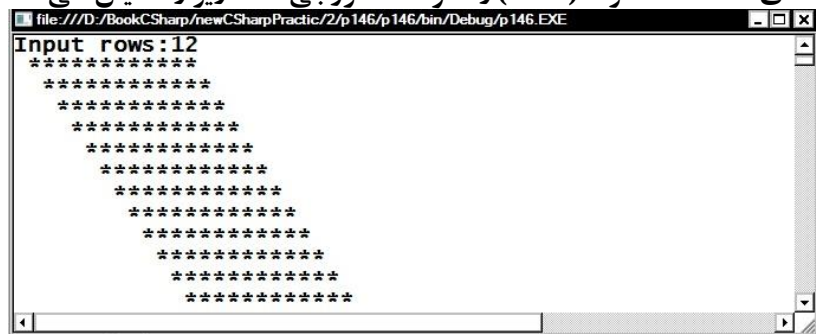
```
file:///D:/BookCSharp/newCSharpPractic/2/p144/p144/bin/Debug/p144.EXE
Input rows:9
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

➤ ۱۴۵. برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی زیر را نمایش می‌دهد:



```
file:///D:/BookCSharp/newCSharpPractic/2/p145/p145/bin/Debug/p145.EXE
Input rows:8
*****
*
*
*
*
*
*
*****
```

➤ . برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد :



```
file:///D:/BookCSharp/newCSharpPractic/2/p146/p146/bin/Debug/p146.EXE
Input rows:12
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی زیر را نمایش می‌دهد:

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد:

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد:

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی مانند زیر را نمایش می‌دهد:

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۲۷

```
file:///D:/BookCSharp/newCSharpPractic/2/p 150/p 150/bin/Debug/p 150.EXE
Input rows:8
*****
*****
*****
*****
****
***
**
*
**
***
****
*****
*****
*****
*****
```

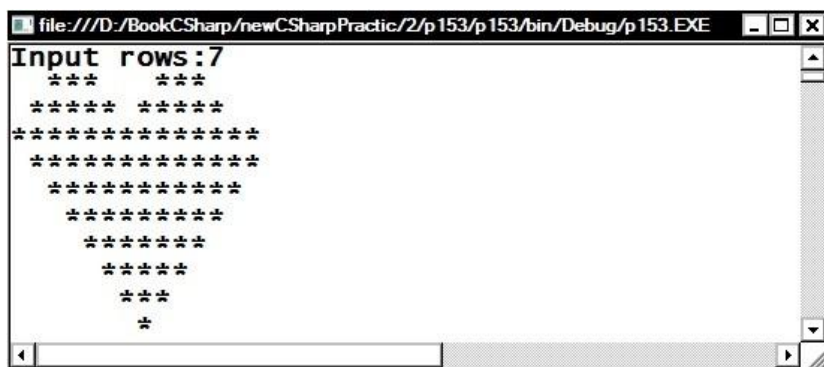
➤ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی مانند زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p 151/p 151/bin/Debug/p 151.EXE
Input rows:5
+
+
+
+
+++++++
+
+
+
+
```

➤ برنامه‌ای که تعداد سطرها را خوانده (rows) و خروجی مانند زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p 152/p 152/bin/Debug/p 152.EXE
Input rows:8
*****
*      *
*      *
*      *
*      *
*      *
*****
*      *
*      *
*      *
*****
```

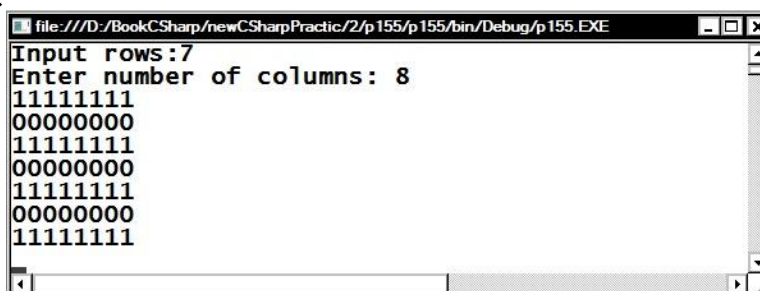
➤ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد:



➤ برنامهای که تعداد سطرها (rows) و یک نام را خوانده، خروجی مانند زیر را نمایش می‌دهد:



➤ برنامه‌ای که تعداد سطرها (rows) و ستون‌ها (cols) را خوانده، خروجی مانند زیر را نمایش



برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی نظیر زیر را نمایش می‌دهد:

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۲۹

```
file:///D:/BookCSharp/newCSharpPractic/2/p156/p156/bin/Debug/p156.EXE
Input rows:6
Enter number of columns: 10
0101010101
0101010101
0101010101
0101010101
0101010101
0101010101
0101010101
0101010101
```

➤ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را از ورودی خوانده، خروجی نظیر زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p157/p157/bin/Debug/p157.EXE
Input rows:6
Enter number of columns: 10
1111111111
1000000001
1000000001
1000000001
1000000001
1111111111
```

➤ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی مانند زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p158/p158/bin/Debug/p158.EXE
Input rows:6
Enter number of columns: 12
101010101010
010101010101
101010101010
010101010101
101010101010
010101010101
```

➤ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی نظیر زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p159/p159/bin/Debug/p159.EXE
Input rows:7
Enter number of columns: 9
111101111
111101111
111101111
000000000
111101111
111101111
111101111
```

➤ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی نظیر زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p160/p160/bin/Debug/p160.EXE
Input rows:7
Enter number of columns: 15
1000000000000001
0100000000000010
0010000000000100
0001000000001000
000010000010000
000001000100000
000000101000000
```

➤ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده، خروجی مانند زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p161/p161/bin/Debug/p161.EXE
Input rows:8
Enter number of columns: 9
123456789
234567899
345678999
456789999
567899999
678999999
789999999
899999999
```

➤ برنامه‌ای که با خواندن تعداد سطرها (rows) و تعداد ستون‌ها (cols) خروجی نظیر زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p162/p162/bin/Debug/p162.EXE
Input rows:5
Input columns: 7
1234567
2345671
3456721
4567321
5674321
```

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند الگوی زیر را نمایش می‌دهد"

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۳۱

```
file:///D:/BookCSharp/newCSharpPractic/2/p163/p163/bin/Debug/p163.EXE
Input rows:8
8888888888888888
8777777777777778
8766666666666678
87655555555555678
8765444444445678
8765433333345678
876543222345678
876543212345678
876543222345678
8765433333345678
876544444445678
8765555555555678
876666666666678
877777777777778
8888888888888888
```

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده و الگوی نظیر خروجی زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p164/p164/bin/Debug/p164.EXE
Input rows:6
1
23
4567
89123456
7891234567891234
56789123456789123456789123456789
```

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده و الگوی مانند خروجی زیر را نمایش می‌دهد "در این

```
file:///D:/BookCSharp/newCSharpPractic/2/p165/p165/bin/Debug/p165.EXE
Input rows:8
1                1
12              21
123            321
1234          4321
12345        54321
123456      654321
1234567    7654321
1234567887654321
```

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده، الگوی نظیر زیر را چاپ می‌نماید:

```
file:///D:/BookCSharp/newCSharpPractic/2/p166/p166/bin/Debug/p166.EXE
Input rows:7
1
2  8
3  9 14
4 10 15 19
5 11 16 20 23
6 12 17 21 24 26
7 13 18 22 25 27 28
```

➤ برنامه‌ای که تعداد سطرها را خوانده (rows) و الگوی مانند خروجی زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p167/p167/bin/Debug/p167.EXE
Input rows:8
1
2 4
7 11 16
22 29 37 46
56 67 79 92 106
121 137 154 172 191 211
232 254 277 301 326 352 379
407 436 466 497 529 562 596 631
```

➤ برنامه‌ای که تعداد سطرها را خوانده، الگوی نظیر زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p168/p168/bin/Debug/p168.EXE
Input rows:7
1
3 2
4 5 6
10 9 8 7
11 12 13 14 15
21 20 19 18 17 16
22 23 24 25 26 27 28
```

➤ برنامه‌ای که تعداد سطرها (rows) را خوانده، الگوی نظیر خروجی زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p169/p169/bin/Debug/p169.EXE
Input rows:7
1
12
123
4444
33333
222222
1111111
```

➤ برنامه‌ای که عددی را خوانده و بر اساس عدد خوانده‌شده، خروجی طبق الگوی زیر را نمایش می‌دهد:

```
file:///D:/BookCSharp/newCSharpPractic/2/p170/p170/bin/Debug/p170.EXE
Enter n:1236754
1236754
123675
12367
1236
123
12
1
```

➤ برنامه‌ای که اگر عدد 24165 را از ورودی بخواند، خروجی با الگوی زیر را نمایش می‌دهد:

### ۶۰۰ برنامه C++ با حل آن‌ها ۱۳۳

```
file:///D:/BookCSharp/newCSharpPractic/2/p171/p171/bin/Debug/p171.EXE
Enter a number:4525262
4525262
525262
25262
5262
262
62
2
```

➤ برنامه‌ای که ابتدا n را خوانده و سپس الگوی مانند خروجی زیر را نمایش می‌دهد

```
file:///D:/BookCSharp/newCSharpPractic/2/p172/p172/bin/Debug/p172.EXE
Enter n:8
1
12
123
1234
12345
123456
1234567
12345678
1234567
123456
12345
1234
123
12
1
```

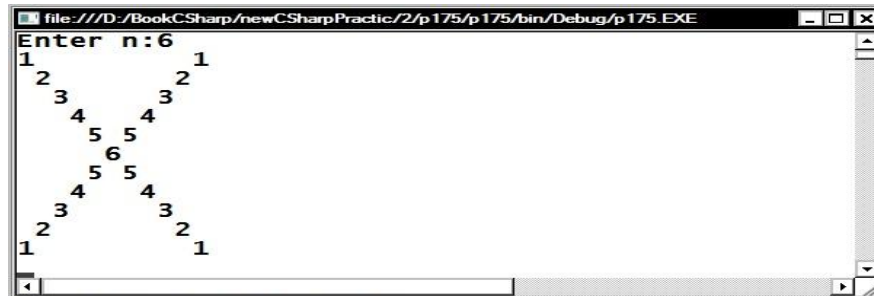
➤ برنامه‌ای که n را خوانده و الگوی خروجی نظیر زیر را نمایش می‌دهد

```
file:///D:/BookCSharp/newCSharpPractic/2/p173/p173/bin/Debug/p173.EXE
Enter n:4
1
123
12345
1234567
12345
123
1
```

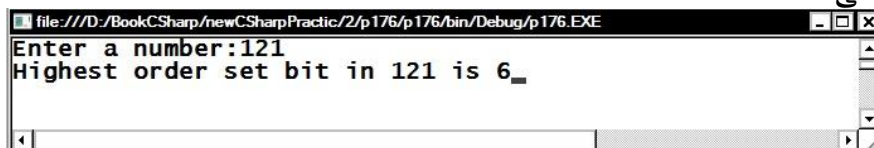
➤ برنامه‌ای که n را خوانده و الگوی مانند خروجی زیر را نمایش می‌دهد

```
file:///D:/BookCSharp/newCSharpPractic/2/p174/p174/bin/Debug/p174.EXE
Enter n:5
*1*
*123*
*12345*
*1234567*
*123456789*
*1234567*
*12345*
*123*
*1*
```

➤ برنامه‌ای که ابتدا n را خوانده و الگوی نظیر خروجی زیر را نمایش می‌دهد:



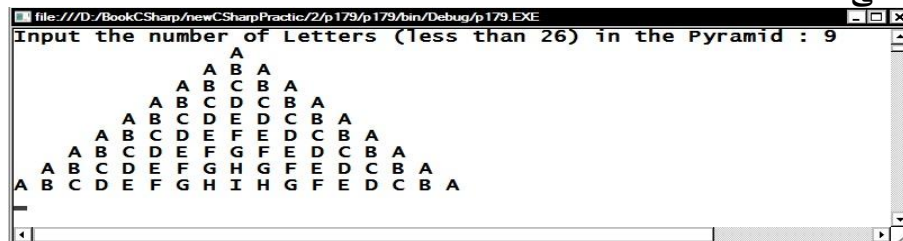
➤ برنامه‌ای که عددی را خوانده، مکان باارزش‌ترین بیت آن را برمی‌گرداند و نمایش می‌دهد.



➤ برنامه‌ای که عددی را خوانده، مکان سمت راست‌ترین بیتی که ۱ باشد را نمایش می‌دهد. به عنوان مثال، اگر عدد ۲۲ در n بیت نمایش داده شود (یعنی ۰۰۰۱۰۱۱۰) مکان سمت راست‌ترین بیت یک آن، ۱ است.

➤ برنامه‌ای که عددی را خوانده، تعداد بیت‌های صفر (۰) و یک (۱) را شمارش می‌کند.

➤ برنامه‌ای که تعداد حروف الفبا را خوانده و الگوی نظیر خروجی زیر را نمایش می‌دهد:



➤ برنامه‌ای که عددی را خوانده، تشخیص می‌دهد عددی زشت (ugly) است یا خیر؟ عددی زشت است که عامل‌های اول آن اعداد ۲، ۳ یا ۵ باشند. برخی از اعداد زشت عبارت‌اند از:

▪ ۱ ۲ ۳ ۵ ۶ ۸ ۹ ۱۰ ۱۲.....

➤ برنامه‌ای که ابتدا n را خوانده، سپس n عدد را خوانده و تشخیص می‌دهد که این اعداد تشکیل تصاعد حسابی را می‌دهند یا خیر.

دو سری از اعداد که تشکیل تصاعد حسابی را می‌دهند در زیر آمده‌اند:

➤ -۱ ۳ ۷ ۱۱ ۱۵ ۱۹ .....

➤ ۵ ۱۰ ۱۵ ۲۰ ۲۵ ۳۰ .....

## ۶۰۰ برنامه C++ با حل آن ها ۱۳۵

- برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  عدد را می‌خواند. و تشخیص می‌دهد این اعداد تشکیل تصاعد هندسی را می‌دهند یا خیر. سری‌های زیر تشکیل تصاعد هندسی را می‌دهند:

○ نرخ ۳ ..... ۵۴ ۱۸ ۶

▪ نرخ  $\frac{1}{2}$  ..... ۱۰ ۲,۵ ۵

- برنامه‌ای که عددی را خواند، سپس اعدادی بین ۱ تا  $n$  را نمایش می‌دهد. که مجموع مکعبات دو یا بیش‌تر عدد برابر با این عدد باشد را نمایش می‌دهد. (نمونه‌ای از این اعداد در شکل زیر آمده است):



$$\begin{aligned} 1729 &= 1^3 + 12^3 = 9^3 + 10^3 \\ 4104 &= 2^3 + 16^3 = 9^3 + 15^3 \\ 13832 &= 2^3 + 24^3 = 18^3 + 20^3 \\ 39312 &= 2^3 + 34^3 = 15^3 + 33^3 \\ 46683 &= 3^3 + 36^3 = 27^3 + 30^3 \\ 32832 &= 4^3 + 32^3 = 18^3 + 30^3 \\ 40033 &= 9^3 + 34^3 = 16^3 + 33^3 \\ 20683 &= 10^3 + 27^3 = 19^3 + 24^3 \\ 65728 &= 12^3 + 40^3 = 31^3 + 33^3 \\ 64232 &= 17^3 + 39^3 = 26^3 + 36^3 \end{aligned}$$

- اگر اضلاع مثلث اعداد تکریمی باشند، تعداد مثلث‌هایی که می‌توان تشکیل داد را نمایش می‌دهد. سپس اضلاع این مثلث‌ها را نمایش می‌دهد (زمانی اضلاع تشکیل مثلث را می‌دهند که مجموع هر دو ضلع بیش‌تر از ضلع سوم باشد).

- اگر اضلاع مثلث اعداد ۱ تا ۹ (اعداد تکریمی) باشند، برنامه‌ای که تعداد مثلث متساوی‌الساقین که با این اعداد می‌توان تشکیل داد را نمایش می‌دهد.

- اگر اضلاع مثلث اعداد تکریمی باشند، تعداد مثلث‌های قائم‌الزاویه که می‌توان تشکیل داد را نمایش می‌دهد. سپس اضلاع این مثلث را نمایش می‌دهد (زمانی که سه ضلع  $a$ ,  $b$  و  $c$  تشکیل مثلث قائم‌الزاویه را می‌دهند که  $a^2 + b^2 = c^2$  یا  $a^2 + c^2 = b^2$  یا  $b^2 + c^2 = a^2$  باشد).

- برنامه‌ای که مقدار سه زاویه را برحسب درجه دریافت کرده، تشخیص می‌دهد که آیا زاویه تشکیل مثلث را می‌دهند یا نه؟

➤ برنامه‌ای که حرفی را خوانده، با استفاده از Switch تشخیص می‌دهد حروف صدادار است یا بی‌صدا؟

➤ برنامه‌ای که دو عدد را خوانده، با استفاده از Swltch عدد بزرگ‌تر را پیدا کرده و نمایش می‌دهد.

➤ برنامه‌ای که ضرایب یک معادله درجه ۲ را خوانده، با استفاده از سوئیچ ریشه‌های آن را محاسبه کرده و نمایش می‌دهد.

➤  $\Delta = b^2 - 4ac$   $ax^2 + bx + c = 0$  اگر  $\Delta > 0$  باشد،  $\frac{-b + \sqrt{\Delta}}{2a}$  و  $\frac{-b - \sqrt{\Delta}}{2a}$  دو ریشه هستند.

اگر  $\Delta = 0$  باشد، معادله دارای یک ریشه  $\frac{-b}{2a}$  است وگرنه، معادله ریشه ندارد یا معادله

دارای ریشه‌های  $\frac{-b + i\sqrt{-\Delta}}{2a}$  و  $\frac{-b - i\sqrt{-\Delta}}{2a}$  است.

➤ برنامه‌ای که عددی را خوانده، تشخیص می‌دهد آیا عددی Strong است یا خیر؟ عددی Strong است که حاصل جمع فاکتوریل ارقام آن برابر خود آن عدد باشد. به عنوان مثال، (عدد ۱۴۵،  $1! + 4! + 5! = 145$ ) یک عدد Strong است.

➤ برنامه‌ای که عددی را خوانده تشخیص می‌دهد عددی Armstrong است یا خیر؟ عددی Armstrong است که مجموع ارقام به توان تعداد ارقام برابر آن عدد باشد. عدد زیر Armstrong است:

$$371 = 3^3 + 7^3 + 1^3 = 371$$

➤ برنامه‌ای که درجه حرارت را به سانتی گراد خوانده و بر اساس جدول زیر پیغام مناسب چاپ می‌کند.

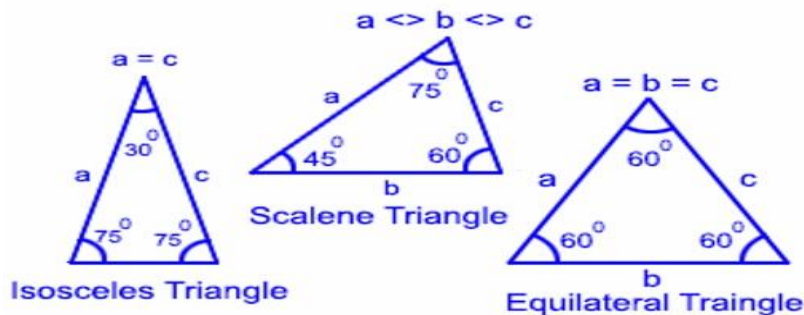
| پیغام             | درجه حرارت |
|-------------------|------------|
| Freezing Weather  | $< 0$      |
| Very Cold Weather | $0 - 10$   |
| Cold Weather      | $10 - 20$  |
| Normal            | $20 - 30$  |
| Hot               | $30 - 40$  |
| Very Hot          | $> 40$     |

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۳۷

- برنامه‌ای یک عدد در مبنای ۲ را خوانده به مبنای ۱۰ تبدیل می‌کند و نمایش می‌دهد. به عنوان مثال عدد زیر از مبنای ۲ به مبنای ۱۰ تبدیل شده است:

$$1100101 = 1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 0 \times 2^3 + 0 \times 2^4 + 1 \times 2^5 + 1 \times 2^6 = 1 + 0 + 4 + 0 + 0 + 32 + 64 = 10^3$$

- برنامه‌ای که اندازه زوایای یک مثلث را گرفته و نوع مثلث را تشخیص می‌دهد. اگر مجموع زوایای مثلث ۱۸۰ درجه نباشد، این زوایا تشکیل مثلث را نمی‌دهند. اما، اگر سه زاویه مثلث برابر باشند، مثلث متساوی‌الاضلاع است، ولی اگر اندازه دو زاویه مثلث برابر باشد، مثلث متساوی‌الساقین است، وگرنه مثلث مختلف‌الاضلاع است (مانند شکل زیر):



- برنامه‌ای که تمام اعداد سه‌رقمی که می‌توان با ارقام ۳ و ۴ و ۵ تولید کرد را نمایش می‌دهد (اعداد را به صورت صعودی مرتب‌شده نمایش می‌دهد).
- برنامه‌ای که تمام اعداد چهاررقمی که با ارقام ۵، ۶، ۷ و ۸ می‌توان تولید کرد را نمایش می‌دهد (در این برنامه در تولید اعداد از ارقام تکراری استفاده نمی‌کند).
- برنامه‌ای که اعداد سه‌رقمی با ارقام فرد می‌توان تولید کرد را نمایش می‌دهد که ارقام آن تکراری نیستند.
- برنامه‌ای که تمام اعداد سه‌رقمی را تولید می‌کند که رقم یکان آن اعداد زوج ۲ تا ۸، رقم دهگان آن اعداد فرد ۵ تا ۹ و رقم یکان آن اعداد مضرب ۳ تا ۹ می‌باشند را نمایش می‌دهد.
- برنامه‌ای که قیمت خرید و فروش یک کالا را خوانده، میزان سود یا زیان را محاسبه کرده، نمایش می‌دهد.
- یک اسب‌سوار به یک پیرزن برخورد کرده، تمام تخم مرغ‌های پیرزن از سبد می‌افتد و می‌شکند. صاحب اسب از پیرزن عذرخواهی می‌کند و به او می‌گوید خسارت تخم

مرغ هایتان را می‌دهم. پیرزن می‌گوید، تخم مرغ‌ها را دو تا دو تا برداشتم، یکی ماند، سه تا سه برداشتم، نیز یکی ماند، چهار تا چهار تا برداشتم، همچنین یکی ماند، پنج تا پنج تا برداشتم، یکی ماند، شش تا شش تا برداشتم، نیز یکی ماند، ولی هفت تا هفت تا برداشتم، هیچ تخم مرغی نماند. برنامه‌ای که تعیین می‌کند، حداقل چند تخم مرغ در سبد پیرزن وجود داشت.

➤ برنامه‌ای که تا زمانی که کاربر کلید ESC را فشار ندهد، کلید از کاربر می‌گیرد. به محض این که کاربر کلید ESC را فشار دهد، از برنامه خارج می‌شود. این برنامه کلید و کد اسکی کاراکتر فشار داده شده را نمایش می‌دهد. کد اسکی کلید ESC، ۲۷ است.

➤ یک رابطه ریاضی خیلی خوب برای تخمین جذر اعداد به این صورت است که ابتدا یک متغیر  $t$  را برابر ۱ قرار می‌دهیم، سپس هر بار  $t$  جدید را از روی  $t$  قدیم این گونه محاسبه می‌کنیم:

$$t_{\text{قدیم}} + \frac{x}{t_{\text{قدیم}}} * 0.5 = t_{\text{جدید}}$$

به این ترتیب اگر این کار را به صورت متوالی انجام دهیم، به تعمیم خوبی برای  $\sqrt{x}$  خواهیم رسید. برنامه‌ای که  $x$  و  $n$  را خواند و تا  $n$  مرحله تخمین  $\sqrt{x}$  را پیش می‌برد (تخمین نهایی  $\sqrt{x}$  همان جدید  $t$  آخرین مرحله خواهد بود).

➤ برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه‌های تکرار تو در تو و کاراکتر '\*' حرف 'A' را نمایش می‌دهد.

➤ برنامه‌ای که تعداد سطر (عددی زوج) را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' حرف 'B' را نمایش می‌دهد.

➤ برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تو در تو و کاراکتر '\*' حرف 'C' را نمایش می‌دهد.

➤ برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'D' را نمایش می‌دهد.

➤ برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'E' را نمایش می‌دهد.

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۳۹

- برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و '\*' کاراکتر 'F' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'G' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'H' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'I' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'J' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*' کاراکتر 'K' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده و با استفاده از حلقه تو در تو و کاراکتر '\*'، کاراکتر 'L' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، و با استفاده از حلقه تکرار تو در تو و کاراکتر '\*'، کاراکتر 'M' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*'، کاراکتر 'N' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*'، کاراکتر 'O' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*'، کاراکتر 'P' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*'، کاراکتر 'Q' را نمایش می‌دهد.

- برنامه‌ای که تعداد و سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'R' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'S' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'T' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'U' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'W' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'V' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از حلقه تکرار تو در تو و کاراکتر '\*', کاراکتر 'X' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطر را خوانده، با استفاده از کاراکتر '\*', کاراکتر 'Y' را نمایش می‌دهد.
- برنامه‌ای که تعداد سطرها را خوانده، با استفاده از کاراکتر '\*', کاراکتر 'Z' را نمایش می‌دهد.

➤ برنامه‌ای که اعداد  $x$  و  $n$  را خوانده، حاصل سری زیر را محاسبه کرده و نمایش می‌دهد:

$$\frac{x^n 0^n}{-n} + \frac{x^1 (1)^{n-1}}{-(n-1)} + \dots + \frac{x^{(n-1)} (n-1)^1}{-1} +$$

$$\frac{x^n n^1}{1} + \frac{x^{(n+1)} (n+1)^2}{2} + \dots + \frac{x^{(2n-1)} (2n-1)^n}{n}$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\frac{1}{1^2+1} + \frac{2}{2^2+2} + \frac{3}{3^2+3} + \dots + \frac{n}{n^2+n}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\frac{\tan x}{\sin(x^n)} + \frac{\tan^2(x)}{\sin(x^{n-1})} + \frac{\tan^3 x}{\sin(x^{n-2})} + \dots + \frac{\tan^n x}{\sin(x)}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد:

$$\leftarrow \frac{\log(\log(\log \dots \log(\log(x)))}{n} \rightarrow$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\frac{1!}{1^2} + \frac{2!}{1^2+2^3} + \frac{3!}{1^2+2^3+3^4} + \dots + \frac{n!}{1^2+2^3+\dots+n^{n+1}}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots \pm \frac{(-1)^n x^{2n}}{(2n)!}$$

➤ برنامه‌ای که  $x$  (بین ۱- تا ۱) و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\arcsin x = x + \frac{2!x^3}{4^1(1!)^2(3)} + \frac{4!x^5}{4^2(2!)^3(5)} + \dots + \frac{(2^n)x^{2n+1}}{4^n(n!)^2(2n+1)}$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\frac{\sqrt{1}}{1} - \frac{1!}{1^2} + \frac{\sqrt{2}}{2} - \frac{2!}{2^2} + \frac{\sqrt{3}}{3} - \frac{3!}{3^2} + \frac{\sqrt{n}}{n} - \frac{n!}{n^2}$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد:

$$\frac{\sqrt{3}-\sqrt{2}}{\sqrt{2^2+2}} + \frac{\sqrt{4}-\sqrt{3}}{\sqrt{3^2+3}} + \frac{\sqrt{n+1}-\sqrt{n}}{\sqrt{n^2+n}}$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد:

$$\frac{1}{2^1(\log 2)^2} + \frac{1}{3^2(\log 3)^3} + \dots + \frac{1}{n^{n-1}(\log n)^n}$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد:

$$\log \frac{2}{2^2-1} + \log \frac{3^2}{3^3-1} + \dots + \log \left( \frac{n^{n-1}}{n^n-1} \right)$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد:

$$\frac{\sqrt{5}}{(2^3+2)^2} + \frac{\sqrt{7}}{(3^3+3)^2} + \dots + \frac{\sqrt{2n+1}}{(n^3+n)^2}$$

➤ برنامه‌ای که  $n$  را خوانده، حاصل عبارت زیر را محاسبه می‌کند و نمایش می‌دهد:

$$\frac{\sin(\frac{1}{2})}{\cos(\frac{1}{1}) \cdot \cos(\frac{1}{2})} + \frac{\sin(\frac{1}{6})}{\cos(\frac{1}{2}) \cdot \cos(\frac{1}{3})} + \dots + \frac{\sin\left(\frac{1}{n(n+1)}\right)}{\cos(\frac{1}{n}) \cdot \cos\left(\frac{1}{n+1}\right)}$$

➤ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\frac{\sqrt{\sin(n) \cdot \log(2!)}}{\tan^n(1)} + \frac{\sqrt{\sin(n-1) \cdot \log(3!)}}{\tan^{n-1}(2)} + \dots + \frac{\sqrt{\sin(0) \cdot \log(n+2)!}}{\tan^n(n+1)}$$

➤ متدی که  $x$  و  $n$  را به‌عنوان پارامتر دریافت کرده، حاصل سری زیر را برمی‌گرداند.

این برنامه از این متد استفاده می‌کند.

$$\frac{\tan(x^1) + \tan(x^2) + \dots + \tan(x^n)}{\sin^1(x) + \sin^2(x) + \dots + \sin^n(x)}$$

➤ متدی  $n$  را به‌عنوان دریافت کرده، سپس حاصل سری زیر را محاسبه و نمایش می‌دهد.

برنامه‌ای که از این متد استفاده می‌کند.

$$(n-1)! \cdot 1^{(n-1)} + (n-2)! \cdot 2^{(n-2)} + \dots + 0! \cdot n^{(n-n)}$$

### ۶۰۰ برنامه C++ با حل آن ها ۱۴۳

- متدی که عدد صحیح  $n$  را به عنوان پارامتر دریافت کرده، حاصل سری زیر را محاسبه می کند و برنامه اصلی آن را نمایش می دهد:

$$\frac{1}{n-1}x + \frac{n-2}{2}x + \dots + \frac{n-1}{1}x$$

- برنامه ای که  $n$  را خوانده با استفاده از متدی حاصل سری زیر را محاسبه و نمایش می دهد:

$$\frac{1*2*3}{(n+1)(n+2)(n+3)} + \frac{2*3*4}{(n+2)(n+3)(n+4)} + \dots + \frac{n(n+1)(n+2)}{(n+n)(n+n+1)(n+n+2)}$$

- برنامه ای که  $x$  و  $n$  را خوانده و توسط متدهای حاصل سری زیر را محاسبه کرده، نمایش می دهد:

$$1 + \frac{nx^1}{1!} + \frac{n(n-1)x^2}{2!} + \dots + \frac{n!x^n}{n!}$$

- **برنامه ای که  $n$  را خوانده و با استفاده از متدی حاصل سری زیر را محاسبه کرده و نمایش می دهد:**

$$\phi = \left\{ \begin{array}{l} \frac{1}{1 + \frac{1}{1+1}} \\ \dots \\ 1 + \frac{1}{1+1} \end{array} \right.$$

جمله  $n$

- برنامه ای که عدد  $n$  را خوانده، کلیه اعداد مضرب  $\gamma$  زیر  $n$  که یکی از ارقام آن  $\gamma$  باشد را نمایش می دهد. در این برنامه برای نمایش اعداد مضرب  $\gamma$  یک متد و برای تعیین این که یک رقم عدد  $\gamma$  است از متد دیگری استفاده می شود.

- برنامه ای که عدد  $n$  را خوانده، کلیه اعداد مضرب  $\gamma$  زیر  $n$  که هیچ یک از ارقام آن  $\gamma$  نباشد را نمایش می دهد.

➤ برنامه‌ای که عددی را خوانده، اگر ارقام از سمت راست به چپ دوتا دوتا کم شدند، Yes، وگرنه No را نمایش می‌دهد. برای تعیین این که اعداد از راست به چپ دو تا کم شدند از یک متد استفاده می‌کند. به‌عنوان مثال، برای اعداد ۲۴۶، ۳۵۷ و ۱۳۵، "Yes" و برای اعداد ۱۲۴، ۳۵۶، 'No' را نمایش می‌دهد.

- برنامه‌ای که رشته‌ای را خوانده، با استفاده از یک متد فضای خالی سمت چپ (ابتدا رشته) را حذف می‌کند و نمایش می‌دهد (کد اسکی کاراکترهای خالی ۳۲ است).
- برنامه‌ای که رشته‌ای را خوانده، با استفاده از اشاره‌گر و بازگشتی، طول رشته را حساب کرده، نمایش می‌دهد.
- برنامه‌ای که رشته‌ای را خوانده، با استفاده از اشاره‌گر و متد بازگشتی معکوس رشته را نمایش می‌دهد.
- برنامه‌ای دو نام را خوانده، تعیین می‌کند که آیا دو نام با هم برابر هستند یا خیر؟
- برنامه‌ای که رشته‌ای را از ورودی خوانده و بین حروف آن فاصله (Blank) ایجاد می‌کند. به عنوان مثال، اگر کاربر رشته Test را وارد کند، برنامه رشته T e s t را نمایش می‌دهد. در این برنامه متغیرهای  $S_1$  (رشته ورودی) و  $S_2$  (رشته خروجی) را داریم.
- برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  رشته را می‌خواند و طول هر یک را نمایش می‌دهد.
- برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  رشته را می‌خواند و رشته‌ای که بیش‌ترین طول را دارد، نمایش می‌دهد.
- برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  رشته را می‌خواند و رشته‌هایی که حرف اول و آخر آن‌ها یکسان باشد را نمایش می‌دهد.
- برنامه‌ای که ابتدا  $n$  را خوانده، سپس  $n$  رشته را خوانده و در پایان یک کاراکتر را می‌خواند و رشته‌هایی که با این کاراکتر شروع می‌شوند یا پایان می‌یابند را نمایش می‌دهد.
- برنامه‌ای که یک رشته را خوانده، سپس یک کاراکتر را می‌خواند و بین حروف رشته آن کاراکتر را اضافه می‌کند.

- برنامه‌ای که رشته‌های  $S_1$  و  $S_2$  را خوانده و کاراکترها را یک‌درمیان در رشته  $S_3$  قرار می‌دهد و رشته  $S_3$  را نمایش می‌دهد.
- برنامه‌ای که رشته‌ای را خوانده، کاراکتری از رشته که بیش‌ترین کد اسکی را دارد، نمایش می‌دهد.
- برنامه‌ای که رشته‌ای را از ورودی خوانده و تمام کاراکترهای تکراری آن را حذف می‌نماید.
- برنامه‌ای که دو رشته را خوانده، حروف مشترک بین آن‌ها را نمایش می‌دهد.
- برنامه‌ای که رشته‌ای را خوانده، سپس دو عدد را می‌خواند و با توجه به اعداد خوانده‌شده، حروف آن مکان‌ها را جابجا نماید. در این برنامه از متد `Change` به‌صورت زیر استفاده می‌شود:
  - `String Change (String s, int n1, int n2),`  
 به‌عنوان مثال، اگر متد `Change` به‌صورت زیر فراخوانی شود:  
`Change ("Fanavarienovin", ۱, ۳)`  
 متد، رشته `"naFavarienovin"` را برمی‌گرداند.
- برنامه‌ای که تعداد فاصله اضافی به سمت چپ رشته اضافه می‌کند تا اندازه رشته به  $n$  تغییر یابد. اگر اندازه رشته بزرگ‌تر یا مساوی  $n$  باشد، رشته هیچ تغییر نمی‌یابد.
- برنامه‌ای که حروف 'a' تا 'z' و 'A' تا 'Z' را به همراه کد اسکی آن‌ها نمایش می‌دهد.
- برنامه‌ای که رشته‌ای را خوانده و کاراکتری که بیش‌ترین تکرار را دارد، نمایش می‌دهد.
- برنامه‌ای که دو رشته را خوانده، رشته دوم را در وسط رشته اول اضافه می‌کند.
- برنامه‌ای که رشته‌ای را خوانده و بعد از هر یک از حروف کوچک، حروف بزرگ آن کاراکتر را اضافه می‌کند. به‌عنوان مثال، رشته `"Tree"` به رشته `"TrReEeE"` تبدیل می‌شود.
- برنامه‌ای که دو رشته را دریافت کرده و رشته دوم را بعد از  $n$  امین کاراکتر رشته اول درج می‌نماید. در این برنامه متد `insert` استفاده‌شده که دارای **اعضای** زیر است:
  - `string insert (string s1, string s2, int n);`

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۴۷

➤ برنامه‌ای که تعدادی رشته را خوانده (برای خاتمه برنامه رشته پایانی Finish وارد می‌گردد)، تعداد رشته‌هایی که در آن‌ها کاراکترهای رقمی ('۰' تا '۹') وجود دارد را شمارش می‌کند.

➤ برنامه‌ای که یک ماتریس را خوانده، تشخیص می‌دهد که آیا این ماتریس یک ماتریسی اسپارس است یا نه؟ ماتریسی اسپارس است که حداقل نصف عناصر آن صفر باشند. یعنی، اگر ماتریس  $n \times m$  باشد، حداقل  $(n*m)/2$  عناصر آن صفر باشد.

➤ برنامه‌ای که یک ماتریس را می‌خواند و تعیین می‌کند که آیا متقارن است یا خیر؟ ماتریسی متقارن است که برابر ترانواده خودش باشد. یعنی  $A=A^T$  به عنوان مثال، ماتریس زیر متقارن است:

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 8 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 8 \end{bmatrix}^T$$

➤ برنامه‌ای که یک عدد را خوانده، تعداد هر رقم استفاده شده در آن را نمایش می‌دهد.

➤ برنامه‌ای که یک عدد مبنای ۲ را به صورت رشته خوانده، متمم یک آن را نمایش می‌دهد.

➤ برنامه‌ای که یک عدد باینری را دریافت کرده، به مبنای ۸ تبدیل می‌کند و نمایش می‌دهد. برای تبدیل عدد از باینری به مبنای ۸ عدد را سه رقم سه رقم جدا کرده و معادل مبنای ۸ را جایگزین هر سه رقم می‌کند (مطابق جدول زیر):

| معادل مبنای ۸ | معادل باینری | معادل مبنای ۸ | معادل مبنای ۲ |
|---------------|--------------|---------------|---------------|
| ۰             | ۰۰۰          | ۱             | ۰۰۱           |
| ۲             | ۰۱۰          | ۳             | ۰۱۱           |
| ۴             | ۱۰۰          | ۵             | ۱۰۱           |
| ۶             | ۱۱۰          | ۷             | ۱۱۱           |

اگر عدد باینری ۰۱۱۰۱۱۰۰ باشد، معادل آن ۶(۱۰۱)، ۵(۱۰۱) و ۱(۰۱) است (یعنی معادل ۱۵۶)

➤ برنامه‌ای که عددی در مبنای ۲ خوانده، به مبنای ۱۶ تبدیل می‌نماید و نمایش می‌دهد. هر چهار بیت مبنای ۲ معادل یک رقم مبنای ۱۶ است (معادل جدول زیر):

| مبنا | مبنا | مبنا | مبنا | مبنا | مبنا | مبنا | مبنا |
|------|------|------|------|------|------|------|------|
|      |      |      |      |      |      |      |      |

|      |      |      |      |      |      |      |      |      |      |
|------|------|------|------|------|------|------|------|------|------|
| ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ | ی ۱۶ |
| ۰    | ۰    | ۰    | ۰    | ۰    | ۰    | ۰    | ۰    | ۰    | ۰    |
| ۴    | ۵    | ۶    | ۷    | ۸    | ۹    | ۱۰   | ۱۱   | ۱۲   | ۱۳   |
| ۸    | ۹    | ۱۰   | ۱۱   | ۱۲   | ۱۳   | ۱۴   | ۱۵   | ۱۶   | ۱۷   |
| C    | D    | E    | F    | G    | H    | I    | J    | K    | L    |

به عنوان مثال، عدد باینری ۱۰۱۱۱۱۱۰۱۰۱۰۱ معادل ۱۷۰۵ (۱=۰۰۰۱, ۷=۰۱۱۱, E=۱۱۰۱, ۵=۰۱۰۱) است.

➤ برنامه‌ای که یک آرایه با n عنصر و سه عدد x, y و z را خواند و مقدار ماکزیمم  $(x*a[i])+(y*a[j])+(z*a[k])$  را برای  $i \leq j \leq k$  پیدا کرده و نمایش می‌دهد.

➤ برنامه‌ای که رشته‌ای را خوانده، کلمات آن را از انتها به ابتدای رشته نمایش می‌دهد. به عنوان مثال، اگر ورودی به صورت زیر باشد:

• I Like Cat Programming

خروجی به صورت زیر خواهد بود.

• Programming Cat Like I

➤ برنامه‌ای که رشته‌ای را خوانده (TXT) و سپس یک الگو (Pat) را می‌خواند و در رشته جست‌وجو می‌نماید. مکان‌های وقوع Pat در TXT را نمایش می‌دهد.

به عنوان مثال، اگر

`TXT= "AABAACAADAABAABA"`

`Pat= "AAB"`

باشد، خروجی به صورت زیر نمایش داده شود:

Pattern Found at index ۰

Pattern Found at index ۹

Pattern Found at index ۱۲

➤ برنامه‌ای که آرایه‌ای با n عنصر که مقادیر آن‌ها ۰ یا یک به ترتیب تصادفی هستند را خوانده، ۰های را به سمت چپ و ۱ها را به سمت راست آرایه منتقل کرده، نمایش می‌دهد. به عنوان مثال، اگر آرایه ورودی به صورت زیر باشد:

[۰, ۱, ۰, ۱, ۰, ۱, ۱, ۰, ۱, ۰, ۱, ۰, ۱, ۰, ۱]

خروجی به صورت زیر خواهد بود:



خروجی به صورت زیر است:

➤ ۱ ۲ ۳ ۴ ۸ ۱۲ ۱۶ ۱۵ ۱۴ ۱۳ ۹ ۵ ۶ ۷ ۱۱ ۱۰

➤ برنامه‌ای که یک آرایه دوبعدی را تعریف کرده، مقداردهی می‌کند (تمام عناصر آرایه (False z) یا (True) هستند)، اگر عنصر  $a[i][j]$  برابر یک (۱) باشد، تمام عنصر سطر  $i$  ام و ستون  $j$  ام را برابر یک قرار می‌دهد. به عنوان مثال، اگر آرایه به صورت زیر باشد:

```
0 1 0
0 0 0
0 0 1
```

خروجی به صورت زیر است:

```
1 1 1
0 1 1
0 1 1
```

➤ برنامه‌ای که مقادیر چند آرایه را مقداردهی اولیه می‌کند و یک نقطه ثابت شده (عنصری از آرایه که مقدار آن عنصر برابر اندیس آن است) را نمایش می‌دهد. چنانچه نقطه ثابت نباشد، ۱- را نمایش خواهد داد. به عنوان مثال، آرایه‌های زیر را در نظر بگیرید:

```
arr۱ = {-۱۰, -۵, ۰, ۳, ۷}
arr۲ = {۱۰, ۲, ۵, ۸, ۱۷}
```

برای  $arr_1$  خروجی ۳ (چون  $arr_1[3]=3$  است) و برای  $arr_2$  خروجی ۱- است.

➤ برنامه‌ای که آرایه‌ای را تعریف کرده و خوانده، به طوری که مقدار عناصر آن ابتدا افزایشی است (یعنی، عنصر قبلی کم‌تر از عنصر بعدی است) و سپس عناصر آن کاهشی است. این برنامه بزرگ‌ترین مقدار این نوع آرایه را نمایش می‌دهد. به عنوان مثال، اگر عناصر آرایه به صورت زیر باشد:

۷ ۴۱ ۱ ۳ ۵۰ ۱۰

خروجی ۵۰ خواهد بود.

➤ برنامه‌ای که یک آرایه را تعریف کرده، سپس می‌خواند و در پایان، آرایه‌ای را ایجاد می‌کند که هر عنصر آن، تعداد عناصری کوچک‌تر از سمت راست خودش را شمارش می‌کند. به عنوان مثال، اگر آرایه به صورت زیر باشد:

## ۶۰۰ برنامه C++ با حل آن‌ها ۱۵۱

arr={۱۲, ۱, ۲, ۳, ۰, ۱۱, ۴}

خروجی به صورت زیر خواهد بود:

۶, ۱, ۱, ۱, ۰, ۱, ۰

➤ برنامه‌ای که آرایه‌ای را تعریف کرده، و می‌خواند. سپس یک عدد را خوانده و آرایه فرعی را نمایش می‌دهد که مجموع عناصر آن برابر عدد وارد شده می‌باشد. اگر آرایه‌ای وجود نداشته باشد که مجموع عناصر آن برابر عدد وارد شده نباشد، پیغام "No SubArrary Found" را نمایش می‌دهد. به عنوان مثال، اگر آرایه‌های ورودی به صورت زیر باشد:

۵ ۱ ۴ ۲۰ ۰ ۳ ۱۰

و اگر sum=۳۰ باشد، آنگاه خروجی به صورت زیر است:

➤ Sum found between indexes ۱ and ۴

➤ برنامه‌ای که یک آرایه را تعریف کرده و می‌خواند، سپس یک عدد را می‌خواند، توسط متدی اگر مجموع سه عنصر آرایه برابر با عدد خوانده شده باشد، آن سه عدد را نمایش می‌دهد و True را برمی‌گرداند.

➤ برنامه‌ای که کوچک ترین عدد مثبت (غیر منفی) که در آرایه غیر مرتب حذف شده ست را می‌یابد و نمایش می‌دهد. به عنوان مثال، اگر آرایه ورودی به صورت زیر باشد:

{۲, ۳, -۷, ۶, ۸, ۱, -۱۰, ۱۵}

خروجی عدد ۴ خواهد بود. ولی برای آرایه زیر خروجی ۲ خواهد شد:

{۱, ۱, ۰, -۱, -۲}

➤ برنامه‌ای که آرایه‌ای را تعریف کرده، خوانده، سپس یک عدد را می‌خواند و زوج مقادیر از عناصر آرایه که اختلاف آن‌ها عدد وارد شده باشد، را نمایش می‌دهد. به عنوان مثال، اگر عناصر آرایه به صورت زیر باشد:

{۲, ۲۰, ۳, ۲, ۵۰, ۸۰}

و عدد ورودی ۷۸ باشد، خروجی به صورت زیر است:

Pair Found: (۲, ۸۰)

➤ برنامه‌ای که آرایه‌ای را تعریف کرده، می‌خواند، سپس هر عنصر را با بزرگ ترین عنصر بعد از خودش (سمت راست خودش) جایگزین می‌کند، به جای آخرین عدد ۱- قرار می‌گیرد و آرایه جدید را نمایش می‌دهد. به عنوان مثال، اگر آرایه ورودی به صورت زیر باشد:

{۱۶, ۱۷, ۴, ۳, ۵, ۲}

آرایه خروجی به صورت زیر خواهد شد:

{۱۷, ۵, ۵, ۵, ۲, -}

➤ برنامه‌ای که آرایه‌ای را تعریف کرده، می‌خواند، سپس یک عدد را خوانده و مجموع ۴ عنصر آرایه که برابر با عدد خوانده شده باشد را نمایش می‌دهد. به عنوان مثال، اگر آرایه ورودی به صورت زیر باشد:

{۱۰, ۲, ۳, ۴, ۵, ۹, ۷, ۸}

و عدد ورودی برابر ۲۳ باشد، آنگاه خروجی به صورت زیر است:

➤ ۳ ۵ ۷ ۸

➤ برنامه‌ای که آرایه‌ای را تعریف کرده و عناصر آن را مقداردهی اولیه می‌کند. سپس عناصری از آرایه که مقادیر آن‌ها تشکیل مثلث را می‌دهد را شمارش می‌کند (سه مقدار از آرایه که مجموع هر دو مقدار بیش تر از مقدار سوم باشد، تشکیل مثلث را می‌دهند).

## منابع

۱. جعفر نژاد قمی، عین اله، عباس نژادورزی، رمضان، برنامه نویسی به زبان C# (ویراست سوم)، بابل انتشارات علوم رایانه، ۱۳۸۸.
۲. عباس نژادورزی، رمضان، حل مسائل C++ (مرجع کامل)، بابل فن آوری نوین، ۱۳۸۸.
۳. عباس نژادورزی، رمضان، آموزش گام به گام برنامه نویسی بانک اطلاعاتی با C#، بابل فن آوری نوین، ۱۳۸۸.
۴. عباس نژادورزی، رمضان، رحیم پور کامی، باقر، هاشمیان، ابراهیم، طراحی سیستم‌های شی گرا به زبان C#، بابل فن آوری نوین، ۱۳۹۱.
۵. عباس نژادورزی، رمضان، یونسی، عمران، عباس نژادورزی، یوسف، حل C++ (آزمایشگاه کامپیوتر مرجع کامل)، بابل فن آوری نوین، ۱۳۹۶.
۶. Andrew Troelsen, "Pro C# ۲۰۰۸ and the .NET ۳,۵ Platform, Fourth Edition", Apress, ۲۰۰۷
۷. Daniel Solis, "Illustrated C# ۲۰۰۸", Apress, ۲۰۰۸
۸. JON SKEET, "C# in Depth", Manning, ۲۰۰۸
۹. Jesse Liberty & Alex Horovitz "Programming .NET ۳,۵", O'Reilly, ۲۰۰۸
۱۰. by Joe Duffy, "Professional .NET Framework ۲,۰", Wrox Press, ۲۰۰۶
۱۱. Donis Marshall, "Microsoft Visual C# ۲۰۰۸: The Language", Microsoft Press, ۲۰۰۸
۱۲. Richard Blum, "C# Network Programming", Sybex, ۲۰۰۳
۱۳. <http://barnamenevis.org>
۱۴. <http://www.codeproject.com/>

۱۵. <http://www.srco.ir/>