

# حل مسائل پایتون

(حل ۶۵۰ برنامه - مرجع کامل)



مولفین: دکتر رمضان عباس نژادورزی  
مهندس یوسف عباس نژادورزی  
مهندس محمد نادعلی زاده چاری

## برخی از عناوین مهم

آشنایی با پایتون (دستورات ورودی و خروجی)  
ساختار تصمیم و حلقه تکرار  
متدها در پایتون  
آرایه‌ها و NumPy  
رشته‌ها  
لیست، چندتایی، دیکشنری  
کلاس‌ها و وراثت  
فایل‌های ورودی و خروجی

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/b3058">http://ktbr.ir/b3058</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/b3032">http://ktbr.ir/b3032</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/b2994">http://ktbr.ir/b2994</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/b2998">http://ktbr.ir/b2998</a>	<a href="http://ketabesabz.com/dl/52181">http://ketabesabz.com/dl/52181</a>
آزمایشگاه C++ (حل مسائل کامپیوتر مرجع کامل)	<a href="http://daneshnegar.com/book_392262.html">http://daneshnegar.com/book_392262.html</a>	<a href="http://ktbr.ir/b2998">http://ktbr.ir/b2998</a>	<a href="http://ketabesabz.com/dl/53546">http://ketabesabz.com/dl/53546</a>
C# با LINQ آموزش گام‌به‌گام	<a href="http://daneshnegar.com/book_369388.html">http://daneshnegar.com/book_369388.html</a>	<a href="http://ktbr.ir/b2845">http://ktbr.ir/b2845</a>	
C++ ساختمان داده‌ها با	<a href="http://daneshnegar.com/book_379094.html">http://daneshnegar.com/book_379094.html</a>	<a href="http://ktbr.ir/b2967">http://ktbr.ir/b2967</a>	<a href="http://ketabesabz.com/dl/52156">http://ketabesabz.com/dl/52156</a>
طراحی سیستم‌های شی گرا با C#	<a href="http://daneshnegar.com/book_374658.html">http://daneshnegar.com/book_374658.html</a>	<a href="http://ktbr.ir/b2962">http://ktbr.ir/b2962</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/b2977">http://ktbr.ir/b2977</a>	<a href="http://ketabesabz.com/dl/52125">http://ketabesabz.com/dl/52125</a>
گرافیک رایانه‌ای با زبان C# برنامه‌نویسی	<a href="http://daneshnegar.com/book_376021.html">http://daneshnegar.com/book_376021.html</a>	<a href="http://ktbr.ir/b2967">http://ktbr.ir/b2967</a>	<a href="http://ketabesabz.com/dl/51049">http://ketabesabz.com/dl/51049</a>
درس و کتکور پایگاه داده پیشرفته	<a href="http://daneshnegar.com/book_392578.html">http://daneshnegar.com/book_392578.html</a>	<a href="http://ktbr.ir/b2964">http://ktbr.ir/b2964</a>	<a href="http://ketabesabz.com/dl/52102">http://ketabesabz.com/dl/52102</a>
فیزیک الکتروسیسته	<a href="http://daneshnegar.com/book_379161.html">http://daneshnegar.com/book_379161.html</a>	<a href="http://ktbr.ir/b2968">http://ktbr.ir/b2968</a>	
تجارت الکترونیکی	<a href="http://daneshnegar.com/book_379188.html">http://daneshnegar.com/book_379188.html</a>	<a href="http://ktbr.ir/b2961">http://ktbr.ir/b2961</a>	<a href="http://ketabesabz.com/dl/51013">http://ketabesabz.com/dl/51013</a>
راهنمای کاربردی کاربری برای شبکه‌های OPNET شبیه‌سازی کامپیوتر	<a href="http://daneshnegar.com/book_392583.html">http://daneshnegar.com/book_392583.html</a>	<a href="http://ktbr.ir/b2850">http://ktbr.ir/b2850</a>	
درس و کتکور سیستم عامل پیشرفته	<a href="http://daneshnegar.com/book_392580.html">http://daneshnegar.com/book_392580.html</a>	<a href="http://ktbr.ir/b2850">http://ktbr.ir/b2850</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/b2852">http://ktbr.ir/b2852</a>	
آزمایشگاه پایگاه داده SQL Server 2012	<a href="http://daneshnegar.com/book_377301.html">http://daneshnegar.com/book_377301.html</a>	<a href="http://ktbr.ir/b2850">http://ktbr.ir/b2850</a>	<a href="http://ketabesabz.com/dl/53447">http://ketabesabz.com/dl/53447</a>

<a href="http://ketabesabz.com/dl/53446">http://ketabesabz.com/dl/53446</a>	<a href="http://ktbr.ir/b28450">http://ktbr.ir/b28450</a>	<a href="http://daneshnegar.com/book/375892.html">http://daneshnegar.com/book/375892.html</a>	کاربرد رایانه در مدیریت و حسابداری
<a href="http://ketabesabz.com/dl/53488">http://ketabesabz.com/dl/53488</a>	<a href="http://ktbr.ir/b28449">http://ktbr.ir/b28449</a>	<a href="http://daneshnegar.com/book/368929.html">http://daneshnegar.com/book/368929.html</a>	آموزش گام به گام برنامه نویسی بانک اطلاعاتی با ویژوال بیسیک نت
	<a href="http://ktbr.ir/b28450">http://ktbr.ir/b28450</a>	<a href="http://daneshnegar.com/book/380238.html">http://daneshnegar.com/book/380238.html</a>	آموزش گام به گام برنامه نویسی به زبان C++
<a href="http://ketabesabz.com/dl/51047">http://ketabesabz.com/dl/51047</a>	<a href="http://ktbr.ir/b28448">http://ktbr.ir/b28448</a>	<a href="http://daneshnegar.com/book/368486.html">http://daneshnegar.com/book/368486.html</a>	دانلود کتاب آموزش گام به گام برنامه نویسی بانک اطلاعاتی با C#
	<a href="http://ktbr.ir/b28399">http://ktbr.ir/b28399</a>		حل مسائل پاسکال
<a href="http://ketabesabz.com/dl/51048">http://ketabesabz.com/dl/51048</a>	<a href="http://ktbr.ir/b28440">http://ktbr.ir/b28440</a>	<a href="http://daneshnegar.com/book/392262.html">http://daneshnegar.com/book/392262.html</a>	حل مسائل C++
<a href="http://ketabesabz.com/dl/51011">http://ketabesabz.com/dl/51011</a>	<a href="http://ktbr.ir/b28399">http://ktbr.ir/b28399</a>	<a href="http://daneshnegar.com/book/374657.html">http://daneshnegar.com/book/374657.html</a>	دانلود کتاب حل مسائل C#
	<a href="http://ktbr.ir/b32672">http://ktbr.ir/b32672</a>	<a href="http://daneshnegar.com/book/392818.html">http://daneshnegar.com/book/392818.html</a>	یا حل کامل C++ برنامه 600 مرجع - C++ آنها (حل مسائل کامل)
	<a href="http://ktbr.ir/b32671">http://ktbr.ir/b32671</a>	<a href="http://daneshnegar.com/book/392817.html">http://daneshnegar.com/book/392817.html</a>	حل مسائل جاوا (حل 600 برنامه - مرجع کامل)
		<a href="http://daneshnegar.com/book/392815.html">http://daneshnegar.com/book/392815.html</a>	600 برنامه # C یا حل کامل آنها (حل مسائل # - مرجع).
		<a href="http://daneshnegar.com/book/392819.html">http://daneshnegar.com/book/392819.html</a>	حل مسائل پایتون (حل 650 برنامه - مرجع کامل)

		<a href="http://daneshnegar.com/book_۳۹۲۸۱۴.html">http://daneshnegar.com/book_۳۹۲۸۱۴.html</a>	دانش 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/b۲۸۳۹">http://ktbr.ir/b۲۸۳۹</a> <u>۷</u>		دانلود کتاب حل مسائل C

---

---

# حل مسائل پایتون

## (حل ۶۵۰ برنامه - مرجع کامل)

---

---

### تألیف:

دکتر رمضان عباس نژادورزی  
مهندس یوسف عباس نژادورزی  
مهندس محمد نادعلی زاده چاری



فن آوری نوین

---

---

سرشناسه	عباس نژاد ورزی، رمضان، ۱۳۴۸ -
عنوان و نام پدیدآور	حل مسایل پایتون (حل ۶۵۰ برنامه - مرجع کامل) / تالیف رمضان عباس نژاد ورزی، یوسف عباس نژاد ورزی، محمد نادعلی زاده چاری.
مشخصات نشر	بابل: فناوری نوین، ۱۳۹۷.
مشخصات ظاهری	۴۴۸ ص: مصور، جدول.
شابک	۴۲۵۰۰۰ ریال ۸-۱۳-۷۲۷۲-۶۰۰-۹۷۸:
وضعیت فهرست نویسی	فیا
موضوع	پیتون (زبان برنامه نویسی کامپیوتر) -- مسائل، تمرین ها و غیره
موضوع	Python (Computer program language) -- Problems, exercises, etc.
موضوع	پیتون (زبان برنامه نویسی کامپیوتر)
موضوع	Python (Computer program language)
شناسه افزوده	عباس نژاد ورزی، یوسف، ۱۳۶۴ -
شناسه افزوده	نادعلی زاده چاری، محمد، ۱۳۶۴ -
رده بندی کنگره	۱۳۹۷ ۲/۷۶/۷۳QA
رده بندی دیویی	۰۰۵/۱۳۳
شماره کتابشناسی ملی	۵۱۳۰۴۲۳



فن آوری نوین

[www.fanavari novin.net](http://www.fanavari novin.net)

بابل، کد پستی ۴۷۱۶۷-۷۳۴۴۸

تلفن: ۰۱۱-۳۲۲۵۶۶۸۷

حل مسائل پایتون (حل ۶۵۰ برنامه - مرجع کامل)

تالیف: رمضان عباس نژاد ورزی، یوسف عباس نژاد ورزی، محمد نادعلی زاده چاری

نوبت چاپ: چاپ اول

سال چاپ: بهار ۹۷

شمارگان: ۱۰۰۰

قیمت: ۴۲۵۰۰ تومان

نام چاپخانه و صحافی:

شابک: ۹۷۸-۶۰۰-۷۲۷۲-۱۳-۸

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، نبش وحید نظری، پلاک ۱۴۲ تلفکس: ۶۶۴۰۰۱۴۴-۶۶۴۰۰۲۲۰

## فهرست مطالب

فصل اول: آشنایی با پایتون (دستورات ورودی و خروجی)	۵
فصل دوم: ساختار تصمیم و حلقه تکرار	۳۰
فصل سوم: متدها در پایتون	۱۴۹
فصل چهارم: آرایه‌ها و NumPy	۲۳۲
فصل پنجم: رشته‌ها	۳۶۱
فصل ششم: لیست، چندتایی، دیکشنری	۳۹۵
فصل هفتم: کلاس‌ها و وراثت	۴۰۷
فصل هشتم: فایل‌های ورودی و خروجی	۴۲۶
منابع:	۴۴۳





## مقدمه

پایتون یک زبان برنامه‌نویسی همه منظوره، سطح بالا، شیء‌گرا و مفسری است که توسط فان روسوم در سال ۱۹۹۱ در کشور هلند طراحی گردید. فلسفه ایجاد این زبان بر دو هدف اصلی خوانایی بالایی برنامه‌ها نوشته شده، کوتاهی و بازدهی نسبی بالای آن است. کلمات کلیدی این زبان به صورت حداقلی تهیه شده‌اند و در مقابل کتابخانه‌هایی که در اختیار کاربر است، بسیار وسیع هستند. کتاب حاضر اولین کتاب حل مسائل پایتون است که شامل ۸ فصل است. فصل‌های این کتاب عبارت انداز:

فصل اول، مفاهیم اولیه پایتون و دستورات ورودی و خروجی را با ۸۰ مثال آموزش می‌دهد.

فصل دوم، ساختارهای تکرار و تصمیم را با ۲۳۵ مثال کاربردی آموزش می‌دهد.

فصل سوم، تعریف متدها (توابع)، فراخوانی آن‌ها و استفاده از توابع کتابخانه‌ای را با ۱۰۷ مثال کاربردی آموزش می‌دهد.

فصل چهارم، آرایه‌ها یک‌بعدی، دوبعدی، چندبعدی و NUMPY را با ۸۱ مثال کاربردی آموزش می‌دهد.

فصل پنجم، رشته‌ها را با ۵۲ مثال برنامه‌نویسی آموزش داده است.

فصل ششم، چندبایی، مجموعه‌ها، لیست‌ها و دیکشنری‌های را با ۳۹ مثال برنامه‌نویسی آموزش داده است.

فصل هفتم، شیء‌گرایی، کلاس‌ها، وراثت، چندریختی و تعریف مجدد عملگرها را با ۱۹ مثال کاربردی آموزش داده است.

فصل هشتم، مفاهیم پایه‌ای ورودی و خروجی فایل‌های متنی و باینری را با ۴۰ مثال کاربردی آموزش داده است.

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)

فصل  
۱

آشنایی با پایتون ( دستورات ورودی و خروجی)

۱. برنامه‌ای که قاعده و ارتفاع متوازی‌الاضلاع را خوانده، مساحت آن را نمایش می‌دهد (مساحت متوازی‌الاضلاع با برابر قاعده \* ارتفاع است).

```
base = float(input("length of base: "))
height = float(input('measurement of height: '))
area = base * height
print("Area is:", area)
```

خروجی برنامه:

length of base: ۱۴  
measurement of height: ۱۵,۵  
Area is: ۲۱۷,۰

۲. برنامه‌ای که شعاع و ارتفاع استوانه‌ای را خوانده، حجم و مساحت کل استوانه را محاسبه می‌کند. حجم و مساحت کل استوانه به صورت زیر محاسبه می‌شود:

$$\text{حجم استوانه} = \pi * (\text{شعاع})^2 * \text{ارتفاع}$$

$$(\text{شعاع})^2 * (\pi * \text{ارتفاع} + ۲ * \pi * \text{شعاع}) = ۲ * \text{حجم مساحت کل}$$

```
pi = 22 / 7
height = float(input('Height of cylinder: '))
radian = float(input('radius of cylinder: '))
volume = pi * radian * radian * height
sur_area = ((2* pi* radian)* height ) + (( pi * radian ** 2 ) * 2)
print("Volume is: ", volume)
print("surface Area is: ", sur_area)
```

خروجی برنامه:

Height of cylinder: ۲۰  
radius of cylinder: ۲  
Volum is: ۲۵۱,۴۲۸۵۷۱۴۲۸۵۷۱۴۲  
surface Area is: ۲۷۶,۵۷۱۴۲۸۵۷۱۴۲۸۵۶

۳. برنامه‌ای که شعاع کره‌ای را خوانده، مساحت و حجم کره را محاسبه می‌کند. مساحت واجم کره برابر است با:

$$\text{مساحت کره} = ۴ * \pi * (\text{شعاع})^2$$

$$\text{حجم کره} = \frac{۴}{۳} * \pi * (\text{شعاع})^3$$

```
pi=22/7
radian = float(input('Radius of sphere: '))
sur_area = 4 * pi * radian **2
volume = (4/3) * (pi * radian ** 3)
print("Surface Area is: ", sur_area)
print("Volume is: ", volume)
```

خروجی برنامه:

Radius of sphere: ۱۰  
Surface Area is: ۱۲۵۷,۱۴۲۸۵۷۱۴۲۸۵۷  
Volume is: ۴۱۹۰,۴۷۶۱۹۰۴۷۶۱۹

۴. برنامه‌ای که طول یک ضلع و تعداد اضلاع را خوانده، مساحت چندضلعی را با فرمول زیر محاسبه می‌کند:

$$\frac{(\text{طول ضلع})^2 * \text{تعداد اضلاع}}{4 + \tan(\pi / \text{تعداد اضلاع})}$$

```
from math import tan, pi
n_sides = int(input("Input number of sides: "))
s_length = float(input("Input the length of a side: "))
p_area = n_sides * (s_length ** 2) / (4 * tan(pi / n_sides))
print("The area of the polygon is: ",p_area)
```

خروجی برنامه:

Input number of sides: ۱۲  
 Input the length of a side: ۸  
 The area of the polygon is: ۷۱۶,۵۵۳۷۵۵,۵۳۲۲۴۴

۵. برنامه‌ای که سرعت باد در ارتفاع ۱۰ متری را به متر در ثانیه (v) و دمای هوا به درجه سانتی‌گراد (T) را خوانده، شاخص وزش باد (wind chill) را محاسبه کند و نمایش می‌دهد:

```
import math
v = float(input("Input wind speed in kilometers/hour: "))
t = float(input("Input air temperature in degrees Celsius: "))
wci = 13.12 + 0.6215*t - 11.37*math.pow(v, 0.16) +
0.3965*t*math.pow(v, 0.16)
print("The wind chill index is", int(round(wci, 0)))
```

خروجی برنامه:

Input wind speed in kilometers/hour: ۴۸  
 Input air temperature in degrees Celsius: ۱۴  
 The wind chill index is ۱۱

۶. برنامه‌ای که یک‌رشته و تعداد تکرار آن را خوانده، رشته را به تعداد عدد وارد شده تکرار می‌نماید و نمایش می‌دهد (هدف برنامه استفاده از عملگر \* برای تکرار رشته است).

```
s = input("Enter a string: ")
rep=int(input("Enter repeat: "))
print (s * rep)
```

متغیر	هدف
s	رشته ورودی
rep	تعداد تکرار رشته

خروجی برنامه:

Enter a string: Python  
 Enter repeat: ۵  
 PythonPythonPythonPythonPython

۷. برنامه‌ای که دو عدد را خوانده، معادل مختلط آن دو عدد را نمایش می‌دهد (عدد اول بخش real و عدد دوم بخش imag می‌باشد). هدف این برنامه ایجاد اعداد مختلط و نمایش آن‌ها است.

```
a = int(input("Enter real part: "))
b = int(input("Enter image part: "))
complex1 =complex(a, b)
print (complex1)
```

متغیر	هدف
a	بخش real
b	بخش imag
complex۱	عدد مختلط تولیدشده

خروجی برنامه:

Enter real part: ۱۲  
 Enter image part: -۴

۱۰ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

۱۲-۴j

۸. وزن یک مولکول آب  $3,0 \times 10^{-23}$  گرم و وزن یک لیتر آب در حدود ۹۵۰ گرم است. برنامه‌ای که وزن آب را برحسب لیتر از ورودی خوانده، تعداد مولکول‌های آن را محاسبه می‌کند.

```
w = input("Enter w: ")
w = int(w)
m = 3.0e-23
l = 950
tedad = (w * l) / m;
print("Tedad = ", tedad);
```

هدف	متغیر
وزن یک مولکول	M
وزن یک لیتر آب	L
وزن آب به لیتر از ورودی خوانده می‌شود.	W
تعداد مولکول‌های آب	tedad

خروجی برنامه :

Enter w: ۱۳۴

Tedad =  $4,2433333333333333 \times 10^{27}$

۹. هرسال برابر با  $3/106 \times 10^7$  ثانیه است. برنامه‌ای که سن شما را دریافت کرده، به ثانیه تبدیل می‌کند.

```
ss = 3.156e7
age = int(input("Enter age:"))
second = age * ss;
print("Second = ", second)
```

هدف	متغیر
هرسال برحسب ثانیه	Ss
سن به سال	Age
سن برحسب ثانیه	Second

خروجی برنامه :

Enter age: ۴۸

Second = ۱۵۱۴۸۸۰۰۰۰,۰

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.

```
a = float(input("Enter salary:"))
b = a * 0.07
m = a * 0.1
p = a - b - m
print("s = ", a, " b = ", b, " m = ", m, " p = ", p)
```

هدف	متغیر
حقوق	S
بیمه	B
مالیات	M
دریافتی	P

خروجی برنامه :

Enter salary: ۱۰۰۰۰۰۰۰

s = ۱۰۰۰۰۰۰۰,۰ b = ۷۰۰۰۰۰,۰۰۰۰۰۰۰۰۰۰۰۰۰۰۰ m = ۱۰۰۰۰۰۰,۰ p = ۸۳۰۰۰۰۰,۰

۱۱. برنامه‌ای که قیمت کالایی را در سال قبل و جاری خوانده، نرخ تورم و قیمت سال بعد آن را چاپ می‌کند. نرخ تورم باید به صورت درصد (مثلاً ۶/۵٪)

حساب گردد. (توضیح: برای محاسبه نرخ تورم و قیمت سال بعد به صورت زیر عمل کنید):

```
y1 = int(input("Enter price for first year:"))
y2 = int(input("Enter price for second year:"))
t = (float)(y2-y1) / y1 * 100;
y3 =y2 + y2 * t;
print("Extera=%", t, "\t\tPrice next year = " ,
y3)
```

متغیر	هدف
T	نرخ تورم
y <sup>۱</sup>	قیمت کالا در سال قبل
y <sup>۲</sup>	قیمت کالا در سال جاری
y <sup>۳</sup>	قیمت تخمینی کالا در سال بعد

خروجی برنامه :

Enter price for first year: ۱۰۰۰۰۰۰۰

Enter price for second year: ۱۲۰۰۰۰۰۰

Extera=% ۲۰,۰ Price next year = ۲۵۲۰۰۰۰۰۰,۰

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.)

۱۲ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

```

id1 = -1
max1 = -1
max2 = -1
id2 = -1
n = int(input("Enter n:"))
if n < 2:
    print("Please enter a number
greater then 1")
else:
    for i in range(1, n+1):
        id = int(input("Enter id:"))
        aver = float(input("Enter average:"))
        if aver > max1:
            id2=id1
            max2 = max1
            max1 = aver
            id1 = id
        else:
            if aver>max2:
                max2=aver
                id2=id
    print ("Max2 = ", max2, '\t', "Id2 =", id2)

```

متغیر	هدف
n	تعداد دانشجویان
max۱	اولین معدل
max۲	دومین معدل
i	شمارنده
id۱	شماره دانشجویی که اولین معدل را دارد.
id۲	شماره دانشجویی که دومین معدل را دارد.
aver	معدل دانشجو
id	شماره دانشجو

خروجی برنامه:

```

Enter n:۳
Enter id:\
Enter average:\۷,۵
Enter id:۲
Enter average:\۲
Enter id:۳
Enter average:\۵
Max۲ = \۵,۰      Id۲ = ۳

```

۲. برنامه‌ای که اعدادی 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.

متغیر	هدف
Num	اعداد موردنظر
I	شمارنده از ۱ تا num
Sum	مجموع مقسوم‌علیه‌ها

```
while True:
    num = int(input("Enter a number:"))
    sum = 0
    for i in range(1, num):
        if (num % i == 0):
            sum += i
    if (sum == num):
        print('\t', "Perfected")
    else:
        print('\t'," Not perfect")
    yes=input("Continue ?")
    if (yes[0] == 'N' or yes[0] == 'n'):
        break
```

تعیین ادامه کار	Yes
-----------------	-----

خروجی برنامه:

```
Enter a number: ۱
        Perfected
Continue ?y
Enter a number: ۱۲
        Not perfect
Continue ?n
```

۳. برنامه ای که تعداد n جمله از سری فیبوناچی را تولید می کند.

۲      ۳      ۵      ۸      ۱۳      ...

سری فیبوناچی = ۱      ۱

```
f1 = 1
f2 = 1
n = int(input("Enter a number:"))
if n == 1:
    print(f1)
    exit(0)
elif n == 2:
    print(f1)
    print(f2)
else:
    print(f1)
    print(f2)
    i = 3
    while i <= n:
        f3 = f1 + f2
        print(f3, " ")
        f1 = f2
        f2 = f3
        i = i + 1
```

متغیر	هدف
I	شمارنده
f۱	جملات سری
f۲	جملات سری
f۳	جملات سری
N	تعداد جملات

خروجی برنامه:

```
Enter a number: ۱
۱
۱
۲
۳
۵
۸
```

## ۱۴ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

۴. برنامه‌ای که رشته‌ای را خوانده، کاراکترهای رشته را با یک فاصله از یکدیگر نمایش می‌دهد (هدف برنامه پیمایش رشته با for می‌باشد).

```
s = input("Enter a string:")
for i in s:
    print(i, end = ' ')
```

تغییر	هدف
S	رشته خوانده شده
I	هر کاراکتر رشته

خروجی برنامه:

Enter a string:Fanavarienovin

F a n a v a r i e n o v i n

۵. فرض کنید بخواهید از بانک ۱۰۰۰۰۰۰۰۰ ریال وام با بهره ۱۵٪ با مدت بازپرداخت ۱۸ ماهه دریافت کنید، بهره وام به صورت زیر محاسبه می‌شود:

$$\text{پرداختی وام} = \frac{\text{بهره وام}}{\dots} = ۲۲۵۰۰۰$$

مبلغ بهره (۱۲۲۵۰۰۰۰) ریال به متقاضی پرداخت می‌گردد. حال چنانچه متقاضی ۱۰۰۰۰۰۰۰۰ ریال نیاز داشته باشد، چقدر وام باید به او پرداخت شود. برنامه‌ای که مبلغ موردنیاز متقاضی، تعداد اقساط و مبلغ بهره را دریافت می‌کند، سپس وامی که باید به متقاضی پرداخت شود و قسط هر ماه را چاپ می‌کند. (توضیح: مبلغ وام پرداختنی به صورت زیر محاسبه می‌گردد).

$$\text{مبلغ متقاضی} = \text{مبلغ}$$

$$\text{مبلغ وام پرداختی} = \frac{\text{بهره نرخ درصد} \times \text{تعداد اقساط} \times \text{مبلغ وام پرداختی}}{۱۲} - \text{مبلغ وام پرداختی}$$

$$\text{مبلغ متقاضی وام}$$

$$\text{مبلغ وام} - \text{مبلغ وام} \times ۱۲ = \dots$$

تذکر: برنامه تا هر زمان که کاربر بخواهد ادامه می‌یابد.

```
while True :
    m = int(input("Enter m:"))
    tedad = int(input("Enter tedad:"))
    s = int (input("Enter s:"))
    k = (12 * m) / (12 - tedad * s / 100)
    p = k / tedad
    print("k = ", k, '\t', p)
    ansi=input("Do you want to continue(y/n):")
    if ansi[0] == 'n':
        break
```

متغیر	هدف
m	مبلغ موردنیاز متقاضی
s	نرخ بهره
k	کل وام پرداختی
tedad	تعداد اقساط
p	پرداختی هر قسط
ansi	شرط ادامه

خروجی برنامه:

Enter m:۸

Enter tedad:۷

Enter s:۱۲

k = ۸,۶۰۲۱۵۰۰۳۷۶۳۴۴۰۸      ۱,۲۲۸۸۷۸۶۴۸۲۳۳۴۸۶۸

Do you want to continue(y/n):y

Enter m:۵

Enter tedad:۹

Enter s:۶

k = ۵,۲۳۵۶۰۲۰۹۴۲۴۰۸۳۷۴      ۰,۵۸۱۷۳۳۵۶۶۰۲۶۷۵۹۷

Do you want to continue(y/n):n



۶. 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.

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

```

byy =int(input("Enter birth date(year):"))
bmm =int(input("Enter birth date(month):"))
bdd =int(input("Enter birth date(day):"))
cyy =int(input("Enter current date(year):"))
cmm =int(input("Enter current date(month):"))
cdd =int(input("Enter current date(day):"))
if cdd < bdd:
    cmm -= 1
    cdd += 30
day = cdd - bdd
if cmm < bmm:
    cyy -= 1
    cmm += 12
month = cmm - bmm
year = cyy - byy
days = day + month * 30 + year * 36
hh = days * 24
mm = hh * 60
ss = mm * 60
print("Old is: {0}/{1}/{2}", year, month, day)
print("Hour is(hh:mm:ss): {0}:{1}:{2}", hh, mm, ss)
    
```

خروجی برنامه:

```

Enter birth date(year): ۱۳۴۸
Enter birth date(month): ۱۰
    
```

```
Enter birth date(day): ۰۴
Enter current date(year): ۱۳۹۶
Enter current date(month): ۰۹
Enter current date(day): ۳۰
Old is: {۰}/{۱}/{۲} ۴۷ ۱۱ ۲۶
Hour is(hh:mm:ss): {۰}:{۱}:{۲} ۴۹۱۵۲ ۲۹۴۹۱۲۰ ۱۷۶۹۴۷۲۰۰۰:
```

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

```
a = int(input("Enter a:"))
b = int(input("Enter b:"))
c = int(input("Enter c:"))
if a > b:
    temp = a
    a = b
    b = temp
if a > c:
    temp = a
    a = c
    c = temp
if a > c:
    temp = a
    a = c
    c = temp
if b > c:
    temp = b
    b = c
    c = temp
print("Sorted is ", a, b, c)
```

هدف	متغیر
اعداد خوانده شده	a, b, c
متغیر کمکی	Temp

خروجی برنامه:

```
Enter a: ۱۲
Enter b: ۸
Enter c: ۹
Sorted is ۸ ۹ ۱۲
```

۸. 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.

هدف	متغیر
تعداد کارکنان	n
شماره کارمندی	id
ساعت کارکرد	h
دستمزد ساعتی	hp
اضافه کاری	ov
شمارنده	i

حقوق	p
------	---

```
n = int(input("Enter n:"))
i = 1
while i <= n:
    id = int(input("Enter id:"))
    h = int(input("Enter h:"))
    hp = int(input("Enter hp:"))
    ov = 0
    if h > 40:
        ov = (3 / 2.0 - 1) * (h - 40) * hp
    p = ov + hp * h
    print("id = ",id," ov = ",ov, " p = ", p)
    i = i + 1
```

خروجی برنامه:

```
Enter n: ۲
Enter id: ۱
Enter h: ۱۰۰
Enter hp: ۹۰۰۰
id = ۱ ov = ۲۷۰۰۰۰,۰ p = ۱۱۷۰۰۰۰,۰
Enter id: ۲
Enter h: ۵۰
Enter hp: ۱۴۰۰۰
id = ۲ ov = ۷۰۰۰۰,۰ p = ۷۷۰۰۰۰,۰
```

۹. 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).

```
p=int(input("Enter p:"))
n = int(input("Enter n:"))
inc = int(input("Enter inc:"))
print("Year Price")
for i in range(1, n+1):
    p = p + (p * inc / 100)
    print( i," ", p)
```

متغیر	هدف
n	تعداد سال پیش‌بینی بودجه
p	ارزش کالا در سال جاری و مهرموم‌های بعد
i	شمارنده
inc	نرخ تورم (به درصد)

خروجی برنامه:

```
Enter p: ۱۰۰۰۰۰
Enter n: ۵
Enter inc: ۲۰
Year Price
۱ ۱۲۰۰۰۰,۰
۲ ۱۴۴۰۰۰,۰
۳ ۱۷۲۸۰۰,۰
۴ ۲۰۷۳۶۰,۰
۵ ۲۴۸۸۳۲,۰
```

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

$$s = \frac{1}{x} - \frac{1}{x+2x}$$

۱۸ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

```
s = 1
pow = 1.0
sum = 0.0
sum1 = 0.0
x = int(input ("Enter x:"))
for i in range(1, 11):
    pow = pow * x
    sum1 =sum1 + i * pow
    sum = sum + s * 1.0 / sum1
    s = -s
print("Sum is ", sum)
```

متغیر	هدف
x	عدد خوانده شده
i	شمارنده
pow	توان های x
sum	مجموع
sum۱	مجموع مخرج
s	علامت یک در میان منفی

خروجی برنامه:

```
Enter x: ۵
Sum is ۰,۱۸۳۸۴۸۷۴۶۹۷۸۹۶۲۴۷
```

۱۱. برنامه ای که مجموعه ای از اعداد دوتایی صحیح را خوانده، حاصل ضرب آنها را بدون استفاده از عملگر ضرب نمایش می دهد. چنانچه کاربر به جای دو عدد، صفر وارد کند، برنامه خاتمه یابد.

```
while True:
    sum = 0
    x = int(input("Enter x:"))
    y = int (input ("Enter y:"))
    if (x == 0 and y == 0):
        break
    temp = y
    if (y < 0):
        temp = -y;
    for i in range(1, temp+1):
        sum = sum + x
    if (y < 0):
        sum = -sum
    print(x, " * ", y, " = ", sum)
```

متغیر	هدف
x	مقدار اول
y	مقدار دوم
i	شمارنده
sum	مجموع (حاصل ضرب)
temp	کمکی

خروجی برنامه:

```
Enter x: ۱۰
Enter y: -۸
۱۰ * -۸ = -۸۰
Enter x: -۸
Enter y: -۱۰
-۸ * -۱۰ = ۸۰
Enter x: ۰
Enter y: ۰
```

۱۲. برنامه ای که عددی را خوانده، مشخص می کند آیا عدد مقارن است یا خیر. نمونه های از اعداد مقارن عبارتند از: ۱۲۴۲۱ و ۳۲۴۲۳. (توضیح: چنانچه عددی برابر با مغلوبش باشد، مقارن است. پس، ابتدا مغلوب عدد را محاسبه کرده، با خودش مقایسه می کنیم.)

متغیر	هدف
Num	عدد وارد شده

```
pow = 10
sum = 0
num = int(input("Enter a number:"))
temp = num
while temp > 0:
    sum = (pow * sum) + temp % 10
    temp = temp // 10
if (sum == num):
    print("Yes")
else:
    print("No")
```

ثابت ۱۰	Pow
کمکی برای محاسبه مغلوب	Temp
مجموع (مغلوب عدد)	Sum

خروجی برنامه:

Enter a number: ۱۲۲۲۱

Yes

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

```
m = int(input("Enter n:"))
n = int(input("Enter m:"))
sum = 0
temp = m
for i in range(1, n):
    sum = 0
    for j in range(1, m + 1):
        sum = sum + temp
    temp = sum
print(m, " ^ ", n, " = ", sum)
```

متغیر	هدف
m	پایه
n	توان
i	شماره ۱ تا n-۱
j	شمارنده ۱ تا m
sum	مجموع (در پایان مقدار توان قرار می‌گیرد)
temp	متغیر کمکی برای نگهداری مجموع هر دوره

خروجی برنامه:

Enter n: ۲

Enter m: ۱۰

۲ ^ ۱۰ = ۱۰۲۴

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

متد **main**: این متد  $n$  را خوانده، متد **fibonacci** را فراخوانی می‌کند.  
 متد **fibonacci**: این متد  $i$  امین عدد فیبوناچی را تولید می‌کند.  $i$  را به‌عنوان پارامتر می‌پذیرد.

```
def fibo (n):
    if (n == 1):
        return (1)
    if (n == 2):
        return (1)
    return (fibo (n - 2) + fibo (n - 1))
def main():
    n = int(input("Enter n:"))
    print("Result is:")
    for i in range(1, n+1):
        print (fibo(i), end = " ")
main()
```

متد	متغیر	هدف
main	n	تعداد اعداد فیبوناچی
	i	شمارنده حلقه

خروجی برنامه :

Enter n:۱۰

Result is:

۱ ۱ ۲ ۳ ۵ ۸ ۱۳ ۲۱ ۳۴ ۵۵

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

متد main: این متد، متدها read، write و calculate را فراخوانی می‌کند.  
 متد read: این متد فاصله را به فوت و اینچ می‌خواند.  
 متد calculate: این متد، هر یک از فواصل برحسب فوت و اینچ را به متر و سانتی‌متر تبدیل می‌کند.  
 متد write: این متد برای نمایش فواصل تبدیل‌شده به کار می‌رود.

متد	متغیر	هدف
عمومی	Foot	ورودی به فوت
	Inch	ورودی به اینچ
	foot_to_meter	فوت به متر
	inch_to_meter	اینچ به متر
	centi_meter	سانتی‌متر
	foot_to_centimeter	فوت به سانتی‌متر
	inch_to_centimeter	اینچ به سانتی‌متر

```

foot = inch = 0
foot_to_meter=foot_to_centimeter=inch_to_centimeter=0
def read():
    foot = int(input("Foot?"))
    inch = int(input("Inch?"))
def calculate():
    foot_to_meter = 0.3048 * foot
    foot_to_centimeter = 100 * foot_to_meter
    inch_to_meter = (1.0 / 12)*0.3048* inch
    inch_to_centimeter = 100 * inch_to_meter
def write():
    print ("The ", foot, " foot is ", foot_to_meter, " meter" )
    print("The ", foot, " foot is ", foot_to_centimeter, "
centiMeter")
    print("The ", inch, ", inch is ", inch_to_meter , " meter")
    print ("The ", inch, ", inch is ",inch_to_centimeter, "
centiMeter")
def main():
    read ()
    calculate ()
    write ()
main()
    
```

خروجی برنامه :

Foot?۱۰۰

Inch?۲۰۰

The ۱۰۰ foot is ۳۰,۴۸ meter  
 The ۱۰۰ foot is ۳۰۴۸,۰ centiMeter  
 The ۲۰۰, inch is ۰ meter  
 The ۲۰۰, inch is ۵۰۸,۰ centiMeter

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

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

m۱ جرم جسم اول، m۲ جرم جسم دوم، d فاصله بین دو جسم و g ثابت جهانی جاذبه با مقدار  $۶,۶۹۳ * ۱۰^{-۸}$  cm(g,sec) است. برنامه‌ای که با استفاده از دو متد مقدار نیروی جاذبه محاسبه می‌کند.

```
def calc_f (m1, m2, d):
    g = 6.673e-8;
    return (g * m1 * m2 / pow (d))
def pow (d):
    return (d * d)
def main():
    m1 = float(input("Enter m1:"))
    m2 = float(input("Enter m2:"))
    d = float(input("Enter d:"))
    print("F is ", calc_f (m1, m2, d))
main()
```

متد	متغیر	هدف
سراسری	g	نیروی جاذبه جهانی
main	m۱	جرم جسم اول
	m۲	جرم جسم دوم
	d	فاصله بین دو جسم

خروجی برنامه:

Enter m۱:۱۲  
 Enter m۲:۱۷  
 Enter d:۶  
 F is ۳,۷۸۱۳۶۶۶۶۶۶۶۶۶۶۶۶۶۶۶۶e-۰۷

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 برحسب متر بر ثانیه، t دما برحسب درجه سانتی‌گراد، (t <= ۱۰) و W شاخص سردی باد است. برنامه‌ای که با استفاده از یک متد، سرعت باد و دما را می‌خواند، شاخص سردی باد را نمایش می‌دهد.

```
import math
def calc_cold (v, t):
    return (33 - ((10 * math.sqrt (v) - v + 10.5) * (33 - t)) / 23.1)
def main():
    v = float(input("Enter v:"))
    t = float (input ("Enter v:"))
    print("W is ", calc_cold (v, t))
main()
```

متد	متغیر	هدف
main	v	سرعت باد
	t	دما

خروجی برنامه:

Enter v:۱۰  
 Enter v:۲۰  
 W is ۱۴,۹۲۲۲۴۶۹۳۴۱۱۷۳۴۶

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

```
def repeat(n, m):
    count = 0
    while n > 0:
        if (n % 10) == m:
            count = count + 1
        n //= 10
    return count
def main():
    n = int(input("Enter n:"))
    m = int(input("Enter m:"))
    print(m, " Repeat ", repeat(n, m), " times")
main()
```

هدف	متغیر	متد
عدد مورد نظر	n	main
رقمی که باید در عدد جستجو گردد	m	
تعداد تکرار رقم در عدد	count	repeat

خروجی برنامه:

```
Enter n: ۱۴۳۱۳۱
Enter m: ۳
۳ Repeat ۲ times
```

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 متدها:

متد main: این متد  $x$  و  $n$  را خوانده، با فراخوانی متدها `fact` و `pow` حاصل عبارت  $\sin(x)$  را چاپ می‌کند.  
 متد fact: این فاکتوریل عددی را برمی‌گرداند.  
 متد pow: این متد توان  $n$  عددی را برمی‌گرداند.

```
def fact(n):
    if n == 1:
        return 1.0;
    else:
        return n * fact(n - 1)
def pow(x, n):
    if n == 0:
        return 1.0
    else:
        return x * pow(x, n - 1)
def main():
    sign = 1
    sum = 0
    x = float(input("Enter x:"))
    n = int(input("Enter n:"))
    for i in range(1, n+1, 2):
```

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



```
sum += sign * pow (x, i) / fact (i)
sign = -sign
print("Sin(", x, ") = ", sum)
main()
```

خروجی برنامه :

Enter x: ۳

Enter n: ۶

Sin( ۳, ۰ ) = ۰,۵۲۴۹۹۹۹۹۹۹۹۹۹۹۹۹۹۹۹۹۹۹

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 متد main تعریف شده‌اند). با اجرای متد cout مقدار x=۰ چاپ می‌گردد. سپس متد f1 فراخوانی می‌گردد. با فراخوانی متد f1 متغیر y به صورت محلی تعریف گردیده، مقدار اولیه ۱۰ به آن تخصیص می‌یابد. سپس با اجرای cout، مقدار x=۱۰ چاپ خواهد شد و مقدار متغیر عمومی x به ۱۰ تغییر می‌یابد ( x += ۱۰). با اجرای cout، مقدار y = ۱۰ چاپ خواهد شد و سپس مقدار ۱۱ در x قرار می‌گیرد (x++). پس از برگشت از متد f1 با اجرای cout، ابتدا یک واحد به متغیر عمومی y اضافه می‌گردد، مقدار ۱ در y قرار می‌گیرد و مقدار ۱ y=۱ چاپ می‌شود. با اجرای آخرین cout مقدار x = ۱۱ چاپ می‌شود. سپس، یک واحد به متغیر x اضافه می‌گردد.

```
def f1():
    global x
    y = 10
    print("y = ", y)
    x += 10
    print("x = ", x)
    x = x + 1
def main():
    global x, y
    print ("x = ", x)
    f1 ()
    y = y + 1
    print ("y = ", y)
    print ("x = ", x)
    x = x + 1
x = y = 0
main()
```

خروجی برنامه :

```
x = ۰
y = ۱۰
x = ۱۰
y = ۱
x = ۱۱
```

۸. برنامه‌ای که اعداد سه‌رقمی را چاپ می‌کند که مجموع فاکتوریل ارقام آن اعداد برابر با خود عدد باشد. به عنوان مثال، اگر n1، n2 و n3 به ترتیب رقم یکان، دهگان و صدگان باشند، فرمول زیر برقرار باشد:  
 $n^3n^2n^1 = n^1! + n^2! + n^3!$

وظایف متدها:

متد main: این متد، کلیه ارقام اعداد سه‌رقمی را جدا کرده، با فراخوانی متد fact مجموع ارقام آن را محاسبه می‌کند. چنانچه مجموع فاکتوریل ارقام هر عدد برابر با خود آن عدد باشد، چاپ می‌شود.  
 متد fact: فاکتوریل n را به روش بازگشتی حساب می‌کند.

۲۴ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

متد	متغیر	هدف
main	i	شمارنده
	n۱	رقم اول
	n۲	رقم دوم
	n۳	رقم سوم
	temp	متغیر کمکی
	sum	مجموع فاکتوریل ارقام
fact	n	عددی که باید فاکتوریل آن حساب گردد

```
def fact(n):
    if n == 0:
        return 1
    else:
        return n * fact (n - 1)
def main():
    print ("Result is:", end='')
    for i in range(100, 1000):
        n1 = i % 10
        temp = i // 10
        n2 = temp % 10
        temp //= 10
        n3 = temp % 10
        sum = fact (n1) + fact (n2) + fact (n3)
        if (sum == i):
            print ("    ", i)
main()
```

خروجی برنامه :

Result is: ۱۴۵

۹. برنامه ای 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.

$$۲۱۴۱ = ۲^۴ + ۱^۱ = ۱۶ + ۱ = ۱۷$$

وظایف متدها:

متد main: این متد، کلیه ارقام اعداد چهاررقمی را جدا کرده، با فراخوانی pow توان ۴ رقم چهارم و توان ۳ رقم سوم را محاسبه می‌کند. چنانچه مجموع توان ۴ رقم چهارم و توان ۳ رقم اول برابر توان ۲ رقم دوم و توان ۳ رقم سوم باشد، عدد را نمایش می‌دهد.

متد pow: این متد توان n عددی را برمی‌گرداند.

```
def pow(x, n):
    if n == 0:
        return 1
    else:
        return x * pow(x, n - 1)
def main():
    print("Result is:", end='')
    for i in range(1000, 10000):
        n1 = i % 10
        temp = i // 10
        n2 = temp % 10
        temp //= 10
        n3 = temp % 10
        temp //= 10
        n4 = temp % 10
        if (n1 + pow(n4, 4)) ==
(n2 * n2 + pow(n3, 3)):
            print(" ", i, end='')
main()
```

متد	متغیر	هدف
main	i	شمارنده
	n1	رقم اول
	n2	رقم دوم
	n3	رقم سوم
	n4	رقم چهارم
pow	temp	متغیرهای کمکی
	x	پایه که باید به توان n برسد
	n	توان

خروجی برنامه:

Result is: ۱۰۱۰ ۱۰۲۳ ۱۰۳۸ ۱۱۰۰ ۱۱۱۱ ۱۱۲۴ ۱۱۳۹ ۱۲۰۷ ۱۲۱۸  
۲۰۴۰ ۲۰۵۹ ۲۱۴۱ ۲۲۳۱ ۲۲۴۸ ۳۰۹۰ ۳۱۹۱ ۳۲۹۸ ۳۴۵۸ ۴۶۷۹

۱۰. برنامه‌ای که دو عدد را خوانده، کوچکترین مضرب مشترک آن‌ها (LCM) را نمایش می‌دهد.

```
def lcm(x, y):
    if x > y:
        z = x
    else:
        z = y
    while(True):
        if((z % x == 0) and (z % y == 0)):
            lcm = z
            break
        z += 1
    return lcm
def main():
    x = int(input("Enter x:"))
    y = int(input("Enter y:"))
    print("lcm is ", lcm(x, y))
main()
```

متد	متغیر	هدف
Main	x	عدد صحیح اول
	y	عدد صحیح دوم
Lcm	x	پارامتر اول
	y	پارامتر دوم

خروجی برنامه:

Enter x:۲۴  
Enter y:۲۸  
lcm is ۱۶۸

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 متدها  
متد main: آرایه را تعریف کرده، n را می‌خواند، سپس در یک حلقه و فراخوانی متد prim آرایه را مقداردهی می‌کند و در پایان مقدار عناصر آرایه را نمایش می‌دهد.

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

```
from array import *
def prim(a, tedad, num):
    for i in range(0, tedad):
        if num % a[i] == 0:
            return 0
    a.append(num)
    return 1
def main():
    tedad = 0
    p = array('i', [])
    n = int(input("Enter n:"))
    for i in range(2, n+1):
        if prim(p, tedad, i) == 1:
            tedad = tedad + 1
    print("Primary is ", end='')
    for i in range(0, tedad):
        print(" ", p[i], end='')
main()
```

متد	متغیر	هدف
main	n	عدد خوانده شده
	p	آرایه‌ای از اعداد اول
	i	شمارنده
prim	tedad	تعداد اعداد اول
	i	شمارنده

خروجی برنامه:

Enter n: ۲۹  
Primary is ۲ ۳ ۵ ۷ ۱۱ ۱۳ ۱۷ ۱۹ ۲۳ ۲۹

۲. 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 متد دارد که عبارتند از: متدی برای خواندن عناصر آرایه، متدی برای مرتب‌سازی و متدی برای چاپ عناصر آرایه. وظایف متدها:

متد main: تعداد دانشجویان را می‌خواند و متدهای readarray، select\_sort و writearray را فراخوانی می‌کند.

متد readarray: شماره‌های دانشجویی را می‌خواند و در آرایه قرار می‌دهد.

متد select\_sort: شماره‌های دانشجویی موجود در آرایه را مرتب می‌کند.

متد writearray: شماره‌های دانشجویی موجود در آرایه را چاپ می‌کند.

متد	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

بزرگ‌ترین عدد	max	
شمارنده	i	writearray

```

from array import *
def readArray (id, n):
    for i in range(0, n):
        id.append(int(input("Enter id["+str( i + 1)+ " ]")))
def select_sort (id, n):
    for i in range(0, n):
        max = i
        for j in range(i+1, n):
            if id[max] > id[j]:
                max = j
        temp = id[i]
        id[i] = id[max]
        id[max] = temp;
def writeArray (id, n):
    print("Sorted output ", end='')
    for i in range(0, n):
        print (" ", id[i], end='')
def main():
    id = array('i', [])
    n = int(input("Enter n:"))
    readArray (id, n)
    select_sort (id, n)
    writeArray (id, n)
main()
    
```

خروجی برنامه :

```

Enter n: ۶
Enter id[۱] ۱
Enter id[۲] ۲
Enter id[۳] ۳
Enter id[۴] ۴
Enter id[۵] ۵
Enter id[۶] ۶
Sorted output  ۱ ۲ ۳ ۴ ۵ ۶
    
```

۳. 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.

```

from array import *
n = int(input("Enter n:"))
a = array ('i', [])
j = 0
for i in range(0, n):
    find = 0
    num = int(input("Enter a["+str(i+1)+ "]:"))
    for j in range(0, i):
        if a[j] > num:
    
```

هدف	متغیر
آرایه‌ای از اعداد	a
باید درج شود یا نه؟	find
تعداد اعداد	n
شمارنده‌ها	i, j
عدد خوانده‌شده	num

```

        find = 1
        break
    if find == 1:
        a.insert(j, num)
    else:
        a.append(num)
print("Result is : ", end='')
for i in range(0, n):
    print(" ", a[i], end='')

```

خروجی برنامه :

```

Enter n:۷
Enter a[۱]:۱
Enter a[۲]:-۱
Enter a[۳]:۳
Enter a[۴]:-۲
Enter a[۵]:۷
Enter a[۶]:۴
Enter a[۷]:۸
Result is : -۲ -۱ ۱ ۳ ۴ ۷ ۸

```

۴. برنامه‌ای که با خواندن تعدادی عدد از ورودی، آن‌ها را در آرایه‌ای قرار می‌دهد. سپس، کلید عناصر آرایه را بر عنصر وسط تقسیم می‌کند. اگر عنصر وسط صفر باشد، بر عنصر بعد از عنصر وسط تقسیم می‌نماید. اگر این عنصر 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 پیدا شود، action فراخوانی می‌شود تا تمام عناصر آرایه را بر آن عنصر تقسیم کند. اگر همه عناصر صفر باشند، برنامه پیام مناسبی صادر می‌کند.

هدف	متغیر	متد
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

```

from array import *
def action (a, loc, n):
    for i in range(0, n):
        print(a[i], " / ", a[loc], " = ", a[i] / a[loc])
def main():
    sign = 1
    a = array ('i', [])
    n = int(input("Enter n:"))

```

```

mid = n // 2
for i in range(0, n):
    a.append(int(input("Enter a["+str( i + 1) + "]:")))
j = 0
for i in range(0, mid + 1):
    if a[mid + (sign * j)] != 0:
        action(a, mid + (sign * j), n)
        return
    if sign == 1:
        j = j + 1
    sign = -sign;
print("All element are zero")
main()

```

خروجی برنامه :

```

Enter n:۷
Enter a[۱]:۱
Enter a[۲]:۰
Enter a[۳]:۰
Enter a[۴]:۰
Enter a[۵]:۴
Enter a[۶]:۲
Enter a[۷]:۶
۱ / ۴ = ۰,۲۵
۰ / ۴ = ۰,۰
۰ / ۴ = ۰,۰
۰ / ۴ = ۰,۰
۴ / ۴ = ۱,۰
۲ / ۴ = ۰,۵
۶ / ۴ = ۱,۵

```

۵. برنامه‌ای که یک آرایه  $3 \times 3$  تعریف کرده، به صورت تصادفی به اعضای آن مقادیر بین صفر تا ۲۰ را تخصیص می‌دهد، اعمال زیر را انجام می‌دهد.

۱. حاصل جمع عناصر ماتریس را نمایش می‌دهد.
۲. میانگین جمع عناصر ماتریس را نمایش می‌دهد.
۳. بزرگترین عنصر هر سطر را نمایش می‌دهد.
۴. کوچکترین عنصر هر ستون را نمایش می‌دهد.

هدف	متغیر	تابع
تعداد سطرهای آرایه	n	main
تعداد ستون‌های آرایه	m	
آرایه $3 \times 3$	a	
آرایه‌ای که باید با اعداد تصادفی صفر تا ۲۰ پر شود یا چاپ شود.	arr	createrandom یا printarray()
تعداد سطرهای	n	
تعداد ستون‌های	m	
شمارنده سطرها از صفر تا $n - 1$	i	

شمارنده ستون از صفر تا $m-1$	j	
------------------------------	---	--

```
import numpy as np
import random
import numpy.matlib
def createRandom(arr, n, m):
    for i in range(0, n):
        for j in range(0, m):
            arr[i, j] = random.randint(0, 20)
    return arr
def printArray(arr, n, m):
    for i in range(0, n):
        for j in range(0, m):
            print(arr[i, j], end = '\t')
        print()
def main():
    n = m = 3
    a = np.matlib.empty((n, m), dtype = 'i')
    a = createRandom(a, n, m)
    printArray(a, n, m)
    print("=====")
    print("Sum is ", np.sum(a))
    print("Average is ", np.average(a))
    print("Max rows are ", np.amax(a, axis = 1))
    print("Min cloums are ", np.amin(a, axis = 0))
main()
```

خروجی برنامه :

```
۱۳    ۵    ۱۴
۱۲    ۱۶    ۲
۴     ۳    ۱۵
```

```
=====
Sum is ۸۴
Average is ۹,۳۳۳۳۳۳۳۳۳۳۳
Max rows are [[۱۴]
[۱۶]
[۱۵]]
Min cloums are [[۴ ۳ ۲]]
```

۶. 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).  
وظایف متدها:

- 🔥 **متد main:** ابتدا تعداد عدد را می‌خواند، با فراخوانی متد `input()` تعدادی عدد را از ورودی خوانده، متد `convert` آن‌ها را معکوس می‌کند و با فراخوانی `print()` اعداد معکوس شده را چاپ می‌کند.
- 🔥 **متد input:** تعدادی عدد را خوانده و در آرایه قرار می‌دهد.
- 🔥 **متد convert:** اعداد موجود در آرایه را معکوس می‌کند.
- 🔥 **متد print:** عناصر موجود در آرایه را چاپ می‌کند.



متد	نام	هدف
main	n	تعداد اعداد
	num	آرایه‌ای از اعداد
input\	i	شمارنده
convert	temp	متغیر کمکی
	i	شمارنده
print\	i	شمارنده

```

from array import *
def input1( a, n):
    for i in range(0, n):
        a.append (int (input ("Enter a[" + str (i + 1) + "]:")))
def convert(a, n):
    for i in range(0, n // 2):
        temp = a[i]
        a[i] = a[n - i - 1]
        a[n - i - 1] = temp
def print1 (a, n):
    for i in range(0, n):
        print("    ", a[i], end='')
def main():
    n = int(input("Enter n:"))
    num = array ('i', [])
    input1 (num, n)
    print("Input is ", end='')
    print1 (num, n)
    convert (num, n)
    print("\nConvert is ", end='')
    print1(num, n);
main()

```

خروجی برنامه:

```

Enter n:^
Enter a[1]:۱
Enter a[2]:۲
Enter a[3]:۳
Enter a[4]:۷
Enter a[5]:۸
Enter a[6]:۱
Enter a[7]:۵
Enter a[8]:۳
Input is  ۱  ۲  ۳  ۷  ۸  ۱  ۵  ۳
Convert is  ۳  ۵  ۱  ۸  ۷  ۳  ۲  ۱

```

۷. 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 به‌جای شماره ماشین ۹۹۹- را وارد می‌کند.

۳۲ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

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

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

```

from array import *
def main():
    table = array ('i', [1000, 2000,
3000, 4000, 5000,8000,\ 10000, 20000, 30000, 50000 ])
    sum = 0
    while True:
        number = int(input("Enter number:"))
        if number == -999:
            break
        n = int(input("Enter n:"))
        sum = 0.0
        i = 1
        while i <= n:
            code =int(input("Enter code "+ str(i)+":"))
            if code >= 0 and code < 10:
                sum += table[code]
            else:
                print("Enter code between 0 to 9")
                continue
            i = i + 1
        print("*** Number is ",number, "    Sum is ", sum)
main()

```

خروجی برنامه :

```

Enter number: ۱
Enter n: ۵
Enter code ۱: ۱
Enter code ۲: ۳
Enter code ۳: ۱۱
Enter code between ۰ to ۹
Enter code ۴: ۸
Enter code ۵: ۶
Enter code ۵: ۷
*** Number is ۱    Sum is ۶۶۰۰۰,۰
Enter number: ۲
Enter n: ۳
Enter code ۱: ۱
Enter code ۲: ۵

```

Enter code ۳:۷

\*\*\* Number is ۲ Sum is ۳۰۰۰۰,۰

Enter number:-۹۹۹

۸. برنامه‌ای که عناصر آرایه‌ای ۴×۴ 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.

```
import numpy.matlib
import numpy as np
def readArray (a):
    for i in range(0, 4):
        print("Enter row(", i
+ 1, "):", end='')
        s = input().split ('
')
        for j in range(0, 4):
            a[i, j] = int(s[j])
def findmaxrow (a):
    for i in range(0, 4):
        a[i, 4] = a[i, 0]
        for j in range(1, 4):
            if a[i, 4] < a[i, j]:
                a[i, 4] = a[i, j];
def writearray(a):
    for i in range(0, 4):
        print ()
        for j in range(0,4):
            print (" ", a[i, j], end='')
        print(" ", a[i, 4])
def main():
    a = np.matlib.zeros ((4, 5))
    readArray (a)
    findmaxrow (a)
    print("*****result*****")
    writearray (a)
main()
```

هدف	متغیر	متد
آرایه‌ای با ۴ سطر و ۵ ستون	a	main
شمارنده‌ها	i, j	readarray
شمارنده‌ها	i, j	findmax
شمارنده‌ها	i, j	writearray

خروجی برنامه :

Enter row( ۱ ):۱ ۲ ۳ ۴

Enter row( ۲ ):۶ ۷ ۸ ۹

Enter row( ۳ ):۱ ۴ ۶ ۷

Enter row( ۴ ):۲ ۶ ۸ ۱

\*\*\*\*\*result\*\*\*\*\*

```
۱,۰    ۲,۰    ۳,۰    ۴,۰    ۴,۰
۶,۰    ۷,۰    ۸,۰    ۹,۰    ۹,۰
۱,۰    ۴,۰    ۶,۰    ۷,۰    ۷,۰
۲,۰    ۶,۰    ۸,۰    ۱,۰    ۸,۰
```

۹. 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، یکخانه به طرف پائین حرکت می‌کنیم و عدد بعدی را در آن مکان می‌نویسیم.  
 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، یکخانه به سمت پائین حرکت می‌کنیم.  
 برنامه‌ای که ماتریس جادویی را تولید کرده و چاپ می‌نماید (ماتریس  $5 \times 5$  زیر را در نظر بگیرید):

۱۵	۸	۱	۲۴	۱۷
۱۶	۱۴	۷	۵	۲۳
۲۲	۲۰	۱۳	۶	۴
۳	۲۱	۱۹	۱۲	۱۰
۹	۲	۲۵	۱۸	۱۱

متد	متغیر	هدف
Main	matrix	ماتریس
	n	تعداد سطر و ستون
fill_matrix	j, i	شماره سطر و ستون
	j', i'	شماره سطر و ستون خانه بعدی
	n	۱ تا شماره n
disp_matrix	j, i	شماره n برای نمایش ماتریس

```
import numpy.matlib
import numpy as np
def rangeXY(i1, j1, n):
    if i1 > n:
        i1 = 0
    if i1 < 0:
        i1 = n
    if j1 > n:
        j1 = 0
    if j1 < 0:
        j1 = n
    return [i1, j1]
def fill_matrix(matrix, n):
    i = 0
    n = n - 1
    j = n // 2
    matrix[i, j] = 1
    for m in range(1, (n + 1) * (n + 1)):
        i1 = i - 1
        j1 = j - 1
        [i1, j1] = rangeXY (i1, j1, n)
        if matrix[i1, j1] != 0:
            i1 = i + 1
            j1 = j
            [i1, j1] = rangeXY (i1, j1, n)
        matrix[i1, j1] = m + 1
        i = i1; j = j1
def disp_matrix (matrix, n):
    for i in range(0, n):
```

```

for j in range(0, n):
    if matrix[i, j] < 10:
        print("  ", matrix[i, j], end='')
    elif matrix[i, j] < 100:
        print("   ", matrix[i, j], end='')
    else:
        print("    ", matrix[i, j], end='')
    print()
def main():
    n = int(input("Enter n:"))
    matrix = np.matlib.zeros ((n, n))
    fill_matrix (matrix, n)
    disp_matrix (matrix, n)
main()

```

خروجی برنامه:

Enter n:۷

```

۲۸,۰  ۱۹,۰  ۱۰,۰  ۱,۰  ۴۸,۰  ۳۹,۰  ۳۰,۰
۲۹,۰  ۲۷,۰  ۱۸,۰  ۹,۰  ۷,۰  ۴۷,۰  ۳۸,۰
۳۷,۰  ۳۵,۰  ۲۶,۰  ۱۷,۰  ۸,۰  ۶,۰  ۴۶,۰
۴۵,۰  ۳۶,۰  ۳۴,۰  ۲۵,۰  ۱۶,۰  ۱۴,۰  ۵,۰
۴,۰  ۴۴,۰  ۴۲,۰  ۳۳,۰  ۲۴,۰  ۱۵,۰  ۱۳,۰
۱۲,۰  ۳,۰  ۴۳,۰  ۴۱,۰  ۳۲,۰  ۲۳,۰  ۲۱,۰
۲۰,۰  ۱۱,۰  ۲,۰  ۴۹,۰  ۴۰,۰  ۳۱,۰  ۲۲,۰

```

۱۰. برنامه‌ای که دو آرایه را با بسته **numpy** ایجاد کرده، اعداد زوج صفر تا ۱۰ را در آرایه اول و اعداد فرد ۱ تا ۱۰ را در آرایه دوم قرار می‌دهد. سپس اعمال زیر را انجام می‌دهد:

+ حاصل جمع، تفریق، ضرب و تقسیم دو آرایه را با استفاده از عملگرهای +، -، \* و / محاسبه کرده، نمایش می‌دهد.

+ آرایه دوم را به انتهای آرایه اول اضافه می‌کند.

+ آرایه اول را به ابتدای آرایه دوم اضافه می‌کند.

+ عناصر آرایه دوم را منحصر به فرد (یکتا یا **unique**) می‌کند.

+ عنصر ۲ را از آرایه دوم حذف می‌کند.

+ کمترین مقدار آرایه اول و دوم را نمایش می‌دهد.

+ بیشترین مقدار آرایه اول و دوم را نمایش می‌دهد.

+ میانگین مقادیر آرایه اول و دوم را نمایش می‌دهد.

+ واریانس آرایه اول و دوم را نمایش می‌دهد.

+ آرایه اول را در آرایه سوم با متد **np.copy()** کپی می‌کند، سپس تعیین می‌کند که آرایه سوم همان آرایه اول است یا نه (عملگر **is**)؟

+ با عملگر = آرایه دوم را در آرایه سوم قرار می‌دهد و تعیین می‌کند که آیا آرایه سوم همان آرایه دوم است (عملگر **is**)؟

+ آرایه اول را مرتب می‌کند.

+ تعیین می‌کند آیا هر عضو آرایه اول بزرگتر از ه هستند یا خیر؟

+ جذر عناصر آرایه دوم را محاسبه کرده، نمایش می‌دهد.

هدف این برنامه آشنایی با ماژول **numpy** و متدهای **append()**، **insert()**، **unique()**، **arrange()**، **delete()**، **amin()**، **amax()**، **average()**، **median()**، **var()**، **sort()**، **copy()** و عملگرهای +، -، \*، /، > و = است.

هدف

متغیر

متد

```
import numpy as np
a = np.arange(0, 10, 2)
print("A = ", a)
b = np.arange(1, 10, 2)
print("B = ", b)
print("A + B= ", a + b)
print("A - B = ", a - b)
print("A*B = ", a * b)
print("A / B = ", a / b)
a= np.append(a, b)
print("A = ", a)
b = np.insert(b, 0, a)
print("B = ", b)
b = np.unique(a)
print("B = ", b)
b = np.delete(b, 2)
print("B = ", b)
print("Min(B) = ", np.amin(a), "\tMin(b) = ", np.amin(b) )
print("Max (A) =", np.amax(a), "\tMax (b) =", np.amax(b))
print("Average(A) = ", np.average(a), "\tAverage(B) = ",
np.average(b))
print("Median(A) = ", np.median(a), "\tMedian(B) = ", np.median(b))
print("Var (A) = ", np.var(a), "\t Var(B) = ", np.var(b))
a = np.sort(a)
print("A = ", a)
c = np.copy(a)
print("C = ", c)
print("C is A = ", c is a)
c = b
print("C = ", c)
print("C is B = ", c is b)
print("A > 5 =", a > 5)
print("Sqrt(B) = ", np.sqrt(b))
```

آرایه اعداد فرد	b	
آرایه نتیجه محاسبات	c	

خروجی برنامه:

```
A = [۰ ۲ ۴ ۶ ۸]
B = [۱ ۳ ۵ ۷ ۹]
A + B = [۱ ۵ ۹ ۱۳ ۱۷]
A - B = [-۱ -۱ -۱ -۱ -۱]
A*B = [۰ ۶ ۲۰ ۴۲ ۷۲]
A / B = [۰. ۰.۶۶۶۶۶۶۷ ۰.۸ ۰.۸۵۷۱۴۲۸۶ ۰.۸۸۸۸۸۸۸۹]
A = [۰ ۲ ۴ ۶ ۸ ۱۳ ۵ ۷ ۹]
B = [۰ ۲ ۴ ۶ ۸ ۱۳ ۵ ۷ ۹ ۱۳ ۵ ۷ ۹]
B = [۰ ۱ ۲ ۳ ۴ ۵ ۶ ۷ ۸ ۹]
B = [۰ ۱ ۳ ۴ ۵ ۶ ۷ ۸ ۹]
Min(B) = ۰      Min(b) = ۰
Max (A) = ۹      Max (b) = ۹
Average(A) = ۴.۵      Average(B) = ۴.۷۷۷۷۷۷۷۷۷۷۸
Median(A) = ۴.۵      Median(B) = ۵.۰
Var (A) = ۸.۲۵      Var(B) = ۸.۳۹۵۰۶۱۷۲۸۴
```

```
A = [۰ ۱ ۲ ۳ ۴ ۵ ۶ ۷ ۸ ۹]
C = [۰ ۱ ۲ ۳ ۴ ۵ ۶ ۷ ۸ ۹]
C is A = False
C = [۰ ۱ ۳ ۴ ۵ ۶ ۷ ۸ ۹]
C is B = True
A > ۰ = [False False False False False True True True True]
Sqrt(B) = [۰. ۱. ۱,۷۳۲۰۵۰۸۱ ۲. ۲,۲۳۶۰۶۷۹۸ ۲,۴۴۹۴۸۹۷۴
۲,۶۴۵۷۵۱۳۱ ۲,۸۲۸۴۲۷۱۲ ۳. ]
```

۱۱. 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).  
وظایف متدها:

متد `prim`: اعداد اول کوچکتر از عدد خوانده شده، را در آرایه قرار می‌دهد.

متد `find_two_numbers`: دو عدد اولی را پیدا می‌کند که مجموع آن‌ها برابر با عدد خوانده شده باشد.

متد	متغیر	هدف
main	n	عدد خوانده شده در هر مرحله
	done	از کاربر می‌پرسد که آیا ادامه دهد یا خیر
	p	آرایه عدد اول
	i, j	شماره اولین و دومین اندیس عدد اول
	tedad	تعداد عدد اول
prim	i	شمارنده
find_two_numbers	i, j	شمارنده‌هایی برای پیدا کردن دو عدد اول

```
from array import *
def prim (a, tedad, num):
    for i in range(0, tedad):
        if num % a[i] == 0:
            return 0
    a.append(0)
    a[tedad] = num
    return 1
def find_two_numbers (p, n1, n2, tedad, n):
    for i in range(0, tedad - 1):
        for j in range(i + 1, tedad):
            if (p[i] + p[j]) == n:
                n1 = p[i]
                n2 = p[j]
            return [n1, n2]
    return n1
def main():
    p = array ('i', [])
    i = 0
    j = 0
```

```

tedad = 0
n = 0
done = 'Y'
while True:
    tedad = 0
    n = int(input("Enter n:"))
    if n % 2 == 1:
        print("Enter a even number\n")
        continue
    for i in range(2, n+1):
        if prim(p, tedad, i) == 1:
            tedad= tedad + 1
    [i, j] = find_two_numbers(p, i, j, tedad - 1, n)
    print(n, " = ", i, " + ", j)
    done = input("You want to continue(y/n):")
    if done[0] == 'n' or done == 'N':
        return
main()

```

خروجی برنامه :

```

Enter n:۸
۸ = ۳ + ۵
You want to continue(y/n):y
Enter n:۲۰
۲۰ = ۳ + ۱۷
You want to continue(y/n):y
Enter n:۱۰۰
۱۰۰ = ۱۱ + ۸۹
You want to continue(y/n):n

```

۱. برنامه‌ای که رشته‌ای را گرفته، کاراکترهای با اندیس فرد آن را نمایش می‌دهد.

```

def odd_values_string(str):
    result = ""
    for i in range(len(str)):
        if i % 2 == 0:
            result = result + str[i]
    return result
str = input("Enter a string:")
print(odd_values_string(str))

```

خروجی برنامه :

```

Enter a string:Python is a very good language.
Pto savr odlnug.

```

۲. برنامه‌ای که یکرشته را گرفته تعداد تکرار هر کلمه را نمایش می‌دهد.

```

def word_count(str):
    counts = dict()
    words = str.split()
    for word in words:
        if word in counts:
            counts[word] += 1
        else:
            counts[word] = 1
    return counts

```



```
print( word_count(input("Enter several words:")))
```

خروجی برنامه:

```
Enter several words:Dog Book Teacher Student Language Book Student
{'Teacher': '\', 'Dog': '\', 'Language': '\', 'Book': '}', 'Student': '}'
```

۳. برنامه‌ای که رمزگذاری سزار را بر روی رشته‌ای انجام می‌دهد.

```
def caesar_encrypt(realText, step):
    outText = []
    cryptText = []
    uppercase = ['A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z']
    lowercase = ['a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v','w','x','y','z']
    for eachLetter in realText:
        if eachLetter in uppercase:
            index = uppercase.index(eachLetter)
            crypting = (index + step) % 26
            cryptText.append(crypting)
            newLetter = uppercase[crypting]
            outText.append(newLetter)
        elif eachLetter in lowercase:
            index = lowercase.index(eachLetter)
            crypting = (index + step) % 26
            cryptText.append(crypting)
            newLetter = lowercase[crypting]
            outText.append(newLetter)
    return outText
code=caesar_encrypt(input("Enter a string:"),int(input("Enter key:")))
code = ''.join(code)
print("Encryped text is ", code)
```

خروجی برنامه:

```
Enter a string:Python is a very good language.
Enter key:۴
Encryped text is Tcxlsrmwezivcksshperkyeki
```

۴. برنامه‌ای که رشته‌ای را دریافت کرده، مقلوب می‌کند.

```
def reverse_string(str1):
    return ''.join(reversed(str1))
print(reverse_string(input("Enter a string:")))
```

خروجی برنامه:

```
Enter a string:Python Exercises.
.sesicrexE nohtyP
```

۵. برنامه‌ای که رشته و مجموعه‌ای از کاراکترها را دریافت کرده، مجموعه کاراکترها را از رشته حذف می‌کند.

```
def strip_chars(str, chars):
    return "".join(c for c in str if c not in chars)
s = input("Original String:")
strip1 =input(" stripping chars:")
print(strip_chars(s, strip1))
```

خروجی برنامه:

```
Original String:Type "copyright", "credits" or "license()" for more information
```

stripping chars:oiacet  
 Typ "pyrgh", "rds" r "lms()" fr mr nfrman

۶. برنامه‌ای که یکرشته را خوانده، با استفاده از حلقه while مقلوب کرده، نمایش می‌دهد.

```
def string_reverse(str1):
    rstr1 = ''
    index = len(str1)
    while index > 0:
        rstr1 += str1[ index - 1 ]
        index = index - 1
    return rstr1
print("Reverse is ", string_reverse(input("Enter a string:")))
```

خروجی برنامه:

Enter a string:Python language  
 Reverse is egaugnal nohtyP

۷. برنامه‌ای که رشته‌ای را خوانده، یک کاراکتر از رشته را به‌طور تصادفی انتخاب می‌کند و نمایش می‌دهد.

```
import random
s = input("Enter a string:")
print("Random char is ", random.choice(s))
```

خروجی:

Enter a string:VerygoodLanguage

خروجی برنامه:

Random char is n

۸. برنامه‌ای که رشته‌ای را خوانده، تمام کاراکترهای کوچک رشته را به بزرگ و کاراکترهای بزرگ را به کوچک تبدیل می‌کند.   
 تابع (charChange)، رشته str را به‌عنوان پارامتر دریافت کرده، ابتدا رشته S (رشته تغییر یافته) را خالی می‌کند، سپس با حلقه for هر کاراکتر رشته str را پیمایش می‌کند، اگر کاراکتر رشته حرف کوچک باشد، آن را به حرف بزرگ تبدیل کرده، به انتهای رشته S اضافه می‌کند، در غیر این صورت، حرف کوچک str[i] را به انتهای رشته S اضافه می‌کند. در پایان، رشته S را برمی‌گرداند.

هدف	متغیر	ماژول
رشته ورودی	Str	ماژول اصلی
آرگومان که رشته ورودی را دریافت می‌کند	Str	charChange
رشته تغییر یافته	s	
اندیس هر کاراکتر رشته str	i	

```
def charChange(str):
    s = ""
    for i in range(0, len(str)):
        if str[i].islower() == True:
            s += str[i].upper()
        else:
            s += str[i].lower()
    return s
str = input("Enter a string:")
```

```
str = charChange(str)
print("Result is ", str)
```

خروجی برنامه:

Enter a string: Python Is A good lanGUAGE.  
Result is pYTHON iS a GOOd LANgUAGE.

۱. برنامه‌ای که یک لیست را دریافت کرده، با ستاره و مقادیر لیست هیستوگرام را رسم می‌کند.

```
def histogram( items ):
    for n in items:
        output = ''
        times = n
        while( times > 0 ):
            output += '*'
            times = times - 1
        print(output)
    histogram([3, 6, 4, 3, 6, 5])
```

خروجی برنامه:

```
***
*****
****
***
*****
*****
```

۲. برنامه‌ای که مجموع مقادیر یک لیست را نمایش می‌دهد.

```
s = sum([10,20,30, 56, -9, 87, -8])
print("Sum of the container: ", s)
```

خروجی برنامه:

Sum of the container: ۱۸۶

۳. برنامه‌ای که اطلاعات یک لیست را دریافت کرده و اعدادی که بره بخش‌پذیرند را فیلتر می‌کند.

```
num_list = [45, 55, 60, 37, 100, 105, 220, 660, 876]
# use anonymous function to filter
result = list(filter(lambda x: (x % 15 == 0), num_list))
print("Numbers divisible by 15 are",result)
```

خروجی برنامه:

Numbers divisible by ۱۵ are [۴۵, ۶۰, ۱۰۵, ۶۶۰]

۴. برنامه‌ای که یک لیست را دریافت کرده و اعداد مثبت آن را فیلتر می‌نماید.

```
nums = [34, 1, 0, -23, 7, -99, 90, -88]
print("Original numbers in the list: ",nums)
new_nums = list(filter(lambda x: x >0, nums))
print("Positive numbers in the list: ",new_nums)
```

خروجی برنامه:

Original numbers in the list: [۳۴, ۱, ۰, -۲۳, ۷, -۹۹, ۹۰, -۸۸]

Positive numbers in the list: [۳۴, ۱, ۷, ۹۰]

## ۴۲ حل مسائل پایتون ( حل ۶۵۰ برنامه - مرجع کامل)

۵. برنامه‌ای که یک لیست را دریافت کرده، بدون استفاده از حلقه تکرار حاصلضرب اعضای آن را نمایش می‌دهد.

```
from functools import reduce
s = input("Enter several numbers:")
numbers = s.split()
nums = []
for i in numbers:
    nums.append(int(i))
nums_product = reduce( (lambda x, y: x * y), nums)
print("Product of the numbers : ",nums_product)
```

خروجی برنامه:

Enter several numbers:۱۲ -۸ ۹ -۱۲ ۱۷ ۹۰ -۸۰

Product of the numbers : -۱۲۶۹۰۴۳۲۰۰

۶. برنامه‌ای که یک لیست را به آرایه از نوع بایت تبدیل می‌کند و نمایش می‌دهد.

```
s = input("Enter several numbers:")
numbers = s.split()
nums = []
for i in numbers:
    nums.append(int(i))
values = bytearray(nums)
print("Result is ", end = '')
for x in values: print(x, end = '\t')
```

خروجی برنامه:

Enter several numbers:۴۴ ۱۰ ۹۷ ۱۳ ۱۷

Result is ۴۴ ۱۰ ۹۷ ۱۳ ۱۷

۷. برنامه‌ای که لیستی از لغات را دریافت کرده، لغتی که بیشترین طول را دارد، نمایش می‌دهد.

```
def find_longest_word(words_list):
    word_len = []
    for n in words_list:
        word_len.append((len(n), n))
    word_len.sort()
    return word_len[-1][1]
words = input("Enter several words:").split()
print(find_longest_word(words))
```

خروجی برنامه:

Enter several words:Php Python C++ C# Cobol

Python

۸. برنامه‌ای که مجموعه‌ای از اعداد را خوانده، حاصلضرب آن‌ها را نمایش می‌دهد.

```
def multiply(numbers):
    total = 1
    for x in numbers:
        total *= x
    return total
```

```
s = input("Enter several numbers:")
numbers = s.split()
nums = []
for i in numbers: nums.append(int(i))
print("Multiply is ", multiply(nums))
```

خروجی برنامه:

Enter several numbers: ۱۰ ۲۰ ۳۰ ۴۰ ۵۰ ۶۰ ۷۰ ۸۰

Multiply is ۴۰۳۲۰۰۰۰۰۰۰۰۰

۹. برنامه‌ای که یک دیکشنری ایجاد می‌نماید که کلیدهای آن "UPPER-CASE" و "LOWER-CASE" و مقادیر آن‌ها تعداد حروف کوچک و بزرگ می‌باشند. برنامه‌ای که رشته‌ای را خوانده، تعداد حروف بزرگ و کوچک آن را شمارش می‌کند.

```
d={"UPPER_CASE":0, "LOWER_CASE":0}
s = input("Enter a string:")
for c in s:
    if c.isupper():
        d["UPPER_CASE"]+=1
    elif c.islower():
        d["LOWER_CASE"]+=1
    else:
        pass
print ("No. of Upper case characters : ", d["UPPER_CASE"])
print ("No. of Lower case Characters : ", d["LOWER_CASE"])
```

خروجی برنامه:

Enter a string: In the name of GOD

No. of Upper case characters : ۴

No. of Lower case Characters : ۱۰

. برنامه‌ای که کلاس Circle را طوری پیاده‌سازی می‌کند. سازنده شعاع دایره را به‌عنوان پارامتر دریافت می‌کند و در فیلد شعاع (r) قرار می‌دهد. این برنامه از کلاس با نمونه‌سازی استفاده می‌کند

```
import math
class Circle:
    PI = math.pi
    r = 0
    def __init__(self):
        self.r = 0
    def __init__(self, r):
        self.r = r
    def __del__(self):
        print("object is deleted")
    def Area(self):
        return self.PI* (self.r * self.r)
    def Perime(self):
        return 2 * self.PI * self.r
    def __str__(self):
        s = "R : "+ str(self.r)
        s += "\t\tArea : " + str(self.Area())
        s += "\t\tPerime : "+str(self.Perime())
        return s
r = int(input("Enter r:"))
c = Circle(0)
print(str(c))
```

```
c = Circle(r)
print(str(c))
del c
```

خروجی برنامه :

```
Enter r: ۱۰
R : ۰ Area : ۰,۰ Perime : ۰,۰
object is deleted
R : ۱۰ Area : ۳۱۴,۱۵۹۲۶۵۳۵۸۹۷۹۳ Perime : ۶۲,۸۳۱۸۵۳۰۷۱۷۹۵۸۶
object is deleted
```

۲. برنامه‌ای که نام کاربر و کلمه عبور را دریافت می‌کند. اگر نام ، وگرنه پیغام Correct و کلمه عبور ۱۲۳۴۵۶ باشد، پیغام fanavarienovin کاربر را نمایش می‌دهد. Not Correct

**توضیح:** این برنامه از کلاسی به نام User استفاده می‌کند که دو فیلد به نام‌های userName و Password دارد و متدی به نام isCorrect() دارد تا تعیین کند، نام کاربر و کلمه عبور صحیح وارد شده است یا خیر؟

```
class User:
    UserName = "fanavarienovin"
    Password = "123456"
    def isCorrect(self, userName, password):
        return self.UserName == userName and self.Password ==
password
userName = input ("Enter user name:")
password = input ("Enter password:")
user1 = User ()
if user1.isCorrect (userName, password) == True:
    print ("Correct")
else:
    print("Not Correct")
```

خروجی برنامه :

```
Enter user name:fanavarienovin
Enter password:۱۲۳۴۵۶
Correct
```

۳. برنامه‌ای که شعاع کره‌ای را دریافت کرده مساحت و حجم کره را محاسبه می‌نماید و نمایش می‌دهد (در این برنامه از کلاسی به نام Sphere استفاده می‌کند). حجم و مساحت کره با فرمول‌های زیر محاسبه می‌شود:

$$V = \frac{4}{3} \pi r^3$$

$$A = 4\pi r^2$$

حجم کره

این کلاس دارای اعضای زیر می‌باشد:

ثابت PI، مقدار ۳,۱۴۱۵ را نگه‌داری می‌کند.

فیلد r، شعاع کره را نگه‌داری می‌کند.

سازنده، مقدار r را به‌عنوان پارامتر دریافت کرده، در فیلد شعاع (r) قرار می‌دهد (چون نام متغیر با نام فیلد یکی می‌باشد از اشاره‌گر self استفاده شده است).

متد Volume()، حجم کره را حساب کرده، برمی‌گرداند.

متد Area()، مساحت کره را محاسبه کرده، برمی‌گرداند.

```
class Sphere:
    PI = 3.1415
    r = 0
    def __init__(self, r):
        if r > 0:
            self.r = r
        else:
            self.r = 0
```

```
def Volume (self):
    return (4.0 / 3) * self.r ** 3 * self.PI
def Area (self):
    return 4.0 * self.r ** 2 * self.PI
r = int(input("Enter r:"))
s = Sphere (r)
print("Volume is ", s.Volume (), " Area is ", s.Area ())
```

خروجی برنامه:  
Enter r: ۱۰

Volume is ۴۱۸۸,۶۶۶۶۶۶۶۶۶۶۶۷ Area is ۱۲۵۶,۶۰۰۰۰۰۰۰۰۰۰۱

۴. برنامه‌ای که دو زمان را مقایسه می‌کند. در این برنامه از کلاسی به نام **Time** استفاده شده است. این کلاس دارای سه فیلد **hour** (ساعت)، **minute** (دقیقه)، **second** (ثانیه)، سازنده و یک متد به نام **isEqual()** است که دو شیء از نوع **Time** را مقایسه می‌کند.

```
class Time:
    hour = 0
    minute = 0
    second = 0
    def __init__(self, hour=0, minute=0, second=0):
        self.hour = hour
        self.minute = minute
        self.second = second
    def isEqual (self, t):
        return self.hour == t.hour and self.minute == t.minute and
self.second == t.second
t1 = Time ()
t2 = Time (10, 55, 40)
if t1.isEqual (t2) == True:
    print ("t1 Equals t2")
else:
    print ("t1 Not Equals t2")
if t2.isEqual (t2) == True:
    print ("t2 Equals t2")
else:
    print ("t2 Not Equals t2")
```

این کلاس دارای اعضای زیر می‌باشد:

فیلدهای **hour**، **minute** و **second** که به ترتیب برای نگهداری مقادیر ساعت، دقیقه و ثانیه به کار می‌روند.

سازنده، سه پارامتر برنامه‌های **hour**، **minute** و **second** را دریافت کرده، در فیلدهای **self.hour**، **self.minute** و **self.second** قرار می‌دهد. چون نام پارامترهای متد با نام فیلدها یکی بود، از کلمه **self** برای دسترسی به فیلدهای کلاس استفاده شده است.

متد **isEqual()** دو شیء از نوع **Time** را مقایسه می‌کند. شیء اول به صورت ضمنی به متد ارسال می‌شود.

خروجی برنامه:

t1 Not Equals t2  
t2 Equals t2

۵. برنامه‌ای که کلاسی طراحی می‌کند که اطلاعات زیر را دارد:  
۱. شماره استادی  
۲. نام استاد  
۳. نام خانوادگی استاد  
۴. ساعات تدریس  
۵. مبلغ پرداختی به ازای هر ساعت تدریس

در این کلاس مبلغ حق‌التدریس استاد را محاسبه می‌کند که به صورت زیر محاسبه می‌شود:  
مبلغ پرداختی به ازای هر ساعت تدریس \* ساعت تدریس = مبلغ حق‌التدریس  
برنامه‌ای که از این کلاس استفاده می‌کند.

```
class Teacher:
    id = ""
    firstName = ""
    lastName = ""
    hour = 0
    payPerOneHour = 0
    def getID(self):
        return self.id
    def setID(self, value):
        self.id = value
    def getFirstName(self):
        return self.firstName
    def setFirstName(self, value):
        self.firstName = value
    def getLastName(self):
        return self.lastName
    def setLastName(self, value):
        self.lastName = value
    def getHour(self):
        return self.hour
    def setHour(self, value):
        if (value > 0):
            self.hour = value
        else:
            self.hour = 0
    def getPayPerOneHour(self):
        return self.payPerOneHour
    def setPayPerOneHour(self, value):
        if value > 0:
            payPerOneHour = value
        else:
            payPerOneHour = 0
    def __init__(self, i="", f="", l="", h=0, p=0):
        self.setID(i)
        self.setFirstName(f)
        self.setLastName(l)
        self.setHour(h)
        self.setPayPerOneHour(p)
    def Payment(self):
        return self.hour * self.payPerOneHour
    def __str__(self):
        return self.id + " " + self.firstName + " " + \
            self.lastName + " " + str(self.hour) + " " + \
            str(self.payPerOneHour) + " " + str(self.Payment ())

def main():
    t1 = Teacher()
    print(str(t1))
    t2 = Teacher("12", "Ali", "Ahmadi", 140, 70000)
    print(str(t2))
main()
```

این کلاس اعضای زیر را دارد:

✚ **فیلد id:** شماره استادی را نگهداری می‌کند.



متد **getID**، برای بازیابی مقدار فیلد **id** و متد **setID** برای مقداری دهی به فیلد **id** به کار می‌رود.

متد **getFirstName**، برای بازیابی **firstName** و متد **setFirstName** برای مقداردهی به فیلد **firstName** به کار می‌رود.

متد **getLastName**، برای بازیابی **lastName** و متد **setLastName** برای مقداردهی به فیلد **lastName** به کار می‌رود.

متد **getHour**، جهت بازیابی مقدار **hour** و متد **setHour** جهت مقداردهی به فیلد **hour** به کار می‌رود. متد **setHour** از ورود داده‌های منفی برای فیلد **hour** جلوگیری می‌کند.

متد **getpayPerOneHour**، جهت بازیابی فیلد **payPerOneHour** و متد **setPerOneHour** برای مقداردهی به فیلد **payPerOneHour** به کار می‌رود و متد **setPerOneHour** از ورود مبلغ منفی در فیلد **payPerOneHour** جلوگیری خواهد کرد.

متد **Payment()**، مبلغ حق‌التدریس پرداختی به استاد را محاسبه کرده برمی‌گرداند.

خروجی برنامه :

Ali Ahmadi ۱۴۰۰ . .

۱. برنامه‌ای که چند جمله را دریافت کرده، در فایل **1.txt** در درایو **D** پوشه **Data** می‌نویسد.

```
import io
myFile = open('D:\\data\\1.txt', 'wt')
while True:
    sentence = input("Enter a sentence: ")
    if len(sentence) != 0:
        sentence += '\n'
        myFile.write(sentence)
    else:
        break
myFile.close ()
```

این برنامه ابتدا بسته **io** را به برنامه اضافه می‌کند، سپس متغیر **myfile** را برای نوشتن در فایل متنی **1.txt** در پوشه **Data** درایو **D** ایجاد می‌کند. در ادامه در یک حلقه با پیغام مناسب جمله‌ای را خوانده، در متغیر **sentence** قرار می‌دهد و چنانچه طول رشته **sentence** صفر نباشد، آن را در **myFile** می‌نویسد و این روند را ادامه می‌دهد تا کاربر رشته‌ای با طول صفر وارد کند. در پایان فایل **myFile** را می‌بندد.

خروجی برنامه :

Enter a sentence: Python is a high level language.  
 Enter a sentence: C# is a good language.  
 Enter a sentence: C++ is a very good language.  
 Enter a sentence:

۲. برنامه ای که چند جمله را میخواند به انتهای فایل متنی ۱.txt در پوشه Data درایو D اضافه میکند.

```
import io
myFile = open('D:\\data\\1.txt', 'at')
while True:
    line = input("Enter a sentence: ")
    if line != "":
        line += '\n'
        myFile.write(line)
    else:
        break
myFile.close ()
```

این برنامه ابتدا بسته io را به برنامه اضافه می کند، سپس فایل متنی ۱.txt موجود در پوشه درایو D را طوری باز می کند تا بتوان اطلاعاتی را به انتهای آن اضافه نمود. در حلقه تکرار با پیغام مناسب جمله ای را خوانده و در line قرار می دهد و اگر line خالی نباشد، آن را به انتهای فایل myFile اضافه می کند و اگر line خالی باشد، حلقه خاتمه می یابد و myFile را می بندد.  
خروجی برنامه :

Enter a sentence: Java is a good high level language.  
Enter a sentence: Pascal is a language.  
Enter a sentence:

۳. برنامه ای که نام یک فایل متنی را از کاربر دریافت کرده، اطلاعات آن را بر روی صفحه نمایش نشان می دهد.

```
import io
import os
fileName=input("Enter name of file: ")
if os.path.exists(fileName):
    myFile = open (fileName, 'rt')
    while True:
        line = myFile.readline()
        if line != "":
            print(line, end='')
        else:
            myFile.close ()
            break
```

این برنامه ابتدا بسته های io و os را به برنامه اضافه می کند و با یک پیغام مناسب نام فایل را خوانده، در متغیر رشته ای fileName قرار می دهد. سپس چک می کند که آیا فایل وجود دارد و در صورت موجود بودن فایل آن را باز می کند تا بتوان اطلاعات آن فایل را خواند. در ادامه در یک حلقه تکرار خط به خط از myFile می خواند و بر روی صفحه نمایش نشان می دهد. این کار را تا زمانی ادامه می دهد که line برابر null شود. یعنی خطی در فایل موجود نباشد.  
خروجی برنامه :

Enter name of file: D:\Data\1.txt  
Python is a high level language.  
C# is a good language.  
C++ is a very good language.  
Java is a good high level language.  
Pascal is a language.

۴. برنامه ای که نام یک دایرکتوری را از طریق خط فرمان دریافت کرده، فایلها و دایرکتوریها آن را بر اساس تاریخ ایجاد مرتب میکند.

```
from stat import S_ISREG, ST_CTIME, ST_MODE
```

```
import os, sys, time
dir_path = sys.argv[1] if len (sys.argv) == 2 else r'.'
data = (os.path.join (dir_path, fn) for fn in os.listdir (dir_path))
data = ((os.stat (path), path) for path in data)
data = ((stat[ST_CTIME], path)
        for stat, path in data if S_ISREG (stat[ST_MODE]))
for cdate, path in sorted (data):
    print (time.ctime (cdate), os.path.basename (path))
```

خروجی برنامه:

```
Fri Feb ۲۱۷:۱۲:۱۷ ۲۰۱۸ ۱.py
Fri Feb ۲۱۷:۳۸:۴۱ ۲۰۱۸ ۲.py
Fri Feb ۲۱۷:۵۶:۳۰ ۲۰۱۸ ۳.py
Fri Feb ۲۱۸:۲۳:۲۶ ۲۰۱۸ ۴.py
```

۵. برنامه‌ای که مسیر و پسوند چند فایل را که در یک لیست وجود دارند، جدا می‌کند و نمایش می‌دهد.

```
import os.path
for path in [ 'test.txt', 'filename', '/user/system/test.txt', '/', " ]:
    print("%s" : '% path, os.path.splitext(path))
```

خروجی برنامه:

```
"test.txt" : ('test', '.txt')
"filename" : ('filename', '')
"/user/system/test.txt" : ('/user/system/test', '.txt')
"/" : ('/', '')
"" : ("", '')
```

۶. برنامه‌ای که ویژگی‌های فایل از قبیل نام فایل، زمان دسترسی، زمان تغییر و غیره را نمایش می‌دهد.

```
import os.path
import time
print('File      :', __file__)
print('Access time :', time.ctime(os.path.getatime(__file__)))
print('Modified time:', time.ctime(os.path.getmtime(__file__)))
print('Change time :', time.ctime(os.path.getctime(__file__)))
print('Size       :', os.path.getsize(__file__))
```

خروجی برنامه:

```
File      : D:/BookCSharp/pythonPractice/^\^/^.py
Access time : Fri Feb ۲۱۸:۳۱:۵۳ ۲۰۱۸
Modified time: Fri Feb ۲۱۸:۳۱:۵۳ ۲۰۱۸
Change time : Fri Feb ۲۱۸:۳۱:۴۲ ۲۰۱۸
Size       : ۳۱۱
```

۷. برنامه‌ای که نام یک فایل متنی را خوانده و اطلاعات فایل را بر روی صفحه‌نمایش نشان می‌دهد. چنانچه تعداد خطوط فایل بیش از ۲۲ سطر باشد، پس از نمایش هر ۲۲ سطر یک مکث موقت می‌کند تا کاربر کلیدی را فشار دهد.

```
import io
def ShowData(urlFile):
    myFile = open (urlFile, "rt")
    count = 0
    while True:
        line = myFile.readline ()
        if line != "":
            if count != 0 and count % 22 == 0:
                print("Press any key to continue ... ")
                input()
            else:
                break
        print(line, end = "")
        count = count+ 1
    myFile.close ()
```

```
def main():
    fileName = input("Enter name of file: ")
    ShowData (fileName)
main()
```

**متد showData()** نام فایلی که می‌خواهد نمایش دهد را با پارامتر urlFile دریافت می‌کند و اطلاعات آن فایل را ۲۲ خط ۲۲ خط نمایش می‌دهد. برای این منظور، ابتدا فایل متنی که نام آن در urlFile قرار دارد را به صورت خواندنی باز کرده و اشاره‌گر آن در متغیر myFile قرار می‌دهد و متغیر count (تعداد خطوط نمایش داده شده) را تعریف می‌کند و مقدار صفر را در آن قرار می‌دهد. در ادامه با استفاده از یک حلقه while خط به خط از فایل myFile خوانده و بر روی صفحه نمایش نشان می‌دهد. اگر تعداد خطوط نمایش داده شده ۲۲ شود (یعنی ۲۲ % count برابر صفر گردد، یک پیغام .... Press any key to continue را نمایش می‌دهد و این روند را تا زمانی ادامه می‌دهد که تمام خطوط فایل خوانده شود. در پایان، فایل myFile را می‌بندد.

**متد main()** با پیغام مناسبی نام فایل را خوانده، در متغیر fileName قرار می‌دهد و با فراخوانی ShowData اطلاعات فایلی که نام آن در fileName است را به صورت صفحه به صفحه نمایش می‌دهد.  
خروجی برنامه :

```
Enter name of file: D:\BookCSharp\pythonPractice\۲\p۱.py
day= int(input("Enter number of a day: "))
if (day < ۱ or day > ۳۶۵):
    print ("Day should be between ۱ and ۳۶۵")
else:
    month = day // ۳۱
    if month > ۶:
        month = ۶
    day -= month * ۳۱
    if day != ۰:
        count = day // ۳۰
        month += count + ۱
        day -= count * ۳۰
    else:
        day = ۳۱
    if day == ۰:
        day = ۳۰
        month = month - ۱
    print("day ", day, " of ", month, " month ", end =")
    if month == ۱:
        print ("Farvardin) ", end =")
    elif month == ۲:
        print ("Ordibehesht) ", end =")
Perss any key to continue ...
```

**بقیه سوالات که در زیر  
عنوان آن‌ها آمده  
است حلشان در کتاب  
چاپی و الکترونیکی  
کاملاً حل شده است.**



✓ مدیر شرکتی به هر متخصص خود ماهانه ۷۵۰۰۰۰ تومان حقوق پرداخت می‌کند. او می‌خواهد بداند که اگر ۱۳/۵ درصد به حقوق هر متخصص اضافه کند، سالانه چقدر به هزینه شرکت اضافه می‌شود. برنامه‌ای که تعداد متخصصین شرکت را خوانده، این کار را انجام می‌دهد.

✓ برنامه‌ای که عددی دورقمی را خوانده، مجموع ارقام و مغلوب آن را چاپ می‌کند.

✓ برنامه‌ای که  $x$  را خوانده، بدون استفاده از عملگر ضرب، عبارت زیر را محاسبه می‌کند:

$$y = 31 * x - 17 * x + 5;$$

(توضیح: در این برنامه، برای انجام ضرب از عملگر << (شیفت به چپ) استفاده می‌کنیم).

✓ برنامه‌ای که تاریخ و زمان فعلی را نمایش می‌دهد.

✓ برنامه‌ای که عدد صحیح  $n$  را خوانده، حاصل عبارت  $n+nn+nnn$  را نمایش می‌دهد.

✓ برنامه‌ای که مبلغ فعلی، نرخ بهره و تعداد سال را خوانده و ارزش پول را برای چند سال آینده که از ورودی خوانده، محاسبه می‌کند.

✓ 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  $R_1$ ،  $R_2$  و  $R_3$  که به‌طور موازی به هم متصل‌اند، با فرمول زیر محاسبه می‌شود:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \dots$$

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.

✓ 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.  
 قاعده \* ارتفاع \* ۰,۵ = 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 به‌صورت درصد وارد می‌گردد که برنامه باید آن را به مقدار اعشاری تبدیل کند. به‌عنوان 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:

$$- = \text{شتاب}$$

✓ 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، تفریق و باقی‌مانده تقسیم صحیح آن‌ها را چاپ می‌کند.



✓ برنامه‌ای که تعیین می‌کند آیا پوسته اجرا ۳۲ یا ۶۴ بیتی است؟

✓ برنامه‌ای که نام سیستم‌عامل، اطلاعات پلت فرم release را نمایش می‌دهد.

✓ برنامه‌ای که نام سایت بسته‌های پایتون را نمایش می‌دهد.

✓ برنامه‌ای که یک برنامه اجرایی خارجی را در پایتون اجرا می‌کند.

✓ 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: همان طور 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 پنج رقمی 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

✓ برنامه‌ای که دو مقدار را خوانده، شماره شناسایی این اشیاء را نمایش می‌دهد.

✓ 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) دوزنقه 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):

$$\text{شعاع} * 2 = \text{قطر}$$

$$\pi * \text{شعاع} * 2 = \text{محیط دایره}$$

$$\text{مساحت دایره} = \pi * (\text{شعاع})^2$$

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

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

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

$$\begin{aligned} a &= m^2 - n^2 \\ b &= 2 * m * n \\ c &= m^2 + n^2 \end{aligned}$$

✓ 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 نام و مسیر فایل فعلی در حال اجرا را نمایش می‌دهد.

✓ 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  $1024$  بایت است).

✓ برنامه‌ای که تعداد پردازنده‌های رایانه 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 سن تان 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 به متغیرهای محیطی دست‌یابی پیدا می‌کند.

✓ برنامه‌ای که نام کاربر فعلی را نمایش می‌دهد.

✓ برنامه‌ای که سه عدد را خوانده، بدون استفاده از دستورات شرطی و حلقه تکرار مرتب می‌کند.

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

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

✓ برنامه‌ای که اطلاعات کپی‌رایت پایتون را نمایش می‌دهد.

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

✓ برنامه‌ای که تعداد آرگومان‌ها و آرگومان‌های خط فرمان را نمایش می‌دهد.

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

✓ 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 به‌عنوان 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 می‌کند 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  $11 = y = 3x + c$  و  $12 = y = ax + b$  را نمایش می‌دهد (برنامه  $a$ ،  $b$  و  $c$  را از ورودی می‌خواند). مختصات نقطه تقاطع به صورت زیر به دست می‌آید:

✓  $x = (b - c) / (3 - a)$

✓  $y = 3 * x + c$

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

- ✓ برنامه‌ای که میزان حافظه مصرفی چند شیء را نمایش می‌دهد.
- ✓ برنامه‌ای که نام میزبانی را که روال آن در حال اجرا است، نمایش می‌دهد.
- ✓ برنامه‌ای که نام فایل را از مسیر جدا کرده، نمایش می‌دهد.
- ✓ برنامه‌ای که یک کاراکتر را خوانده، کد اسکی آن را نمایش می‌دهد (در این برنامه متغیر ch کاراکتر خوانده‌شده را نگهداری می‌کند).
- ✓ برنامه‌ای که اطلاعات محیط کاربر را نمایش می‌دهد.
- ✓ 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 عدد دیگر n را می‌خواند و n امین بیت عدد را برابر ۱ (بک) قرار می‌دهد.

✓ برنامه‌ای که ابتدا عددی را خوانده، سپس n را می‌خواند و n امین بیت عدد خوانده شده را صفر می‌نماید. به عنوان مثال، شکل زیر امین بیت عدد ۵۵ را صفر می‌نماید.

55	←	num (in decimal)
4	←	n (in decimal)
0000 0001	←	1 (in binary)
0001 0000	←	$1 \ll n$
1110 1111	←	$\sim (1 \ll n)$
0011 0111	←	num (in binary)
0010 0111	←	$\text{num} \& (\sim (1 \ll n))$

✓ 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: ✓

$$\begin{array}{r} a \quad 00011010 \\ b \wedge 01000001 \\ \hline x \quad 01011011 \end{array} \quad \begin{array}{r} b \quad 01000001 \\ x \wedge 01011011 \\ \hline a \quad 00011010 \end{array} \quad \begin{array}{r} a \quad 00011010 \\ x \wedge 01011011 \\ \hline b \quad 01000001 \end{array}$$

✓ برنامه‌ای که سه عدد را به‌عنوان قاعده کوچک، قاعده بزرگ و ارتفاع یک دوزنقه خوانده، مساحت دوزنقه را نمایش می‌دهد. مساحت دوزنقه برابر با:

مساحت دوزنقه

✓ 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.

✓ برنامه‌ای که کم‌ترین، بیش‌ترین مقدار اعداد صحیح، اعداد صحیح با طول بلند و اعداد اعشاری را نمایش می‌دهد.

✓ برنامه‌ای که زمان فعلی سیستم را نمایش می‌دهد.

✓ برنامه‌ای که  $n$  ثانیه به زمان فعلی اضافه می‌کند.

✓ برنامه‌ای که زاویه‌های را به درجه دریافت کرده، به رادیان تبدیل می‌نماید و نمایش می‌دهد.

✓ برنامه‌ای که زاویه‌های را به رادیان دریافت کرده، به درجه تبدیل می‌نماید و نمایش می‌دهد.

✓ برنامه‌ای که سه مقدار را خوانده، نوع آن‌ها را نمایش می‌دهد.

✓ پروژه برنامه نویسی ۱: برنامه ای که یک عدد ۵ رقمی را خوانده، مقلوب ارقام عدد را بافاصله نمایش می دهد.

✓ 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 (۱-۹) 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

✓ برنامه‌ای که رشته‌ای را خوانده، تعداد حروف صدادر رشته را شمارش می‌کند (حروف صدا دار عبارتند از: `e`، `i`، `u`، `o`، `&`.) هدف برنامه آشنایی بیشتر با پیمایش رشته با for است.

✓ برنامه‌ای که خروجی زیر را ایجاد می‌کند.

۷	۶	۵	۴	۳	۲	۱
۶	۵	۴	۳	۲	۱	
۵	۴	۳	۲	۱		
۴	۳	۲	۱			
۳	۲	۱				
۲	۱					
۱						

✓ 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.

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:

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

```

○ ۱      ۱      ۱      ۱      ۱      ۱      ۱      ۱
○ ۲      ۲      ۲      ۲      ۲      ۲      ۲      ۲
○ ۳      ۳      ۳      ۳      ۳      ۳      ۳      ۳
○ ۴      ۴      ۴      ۴      ۴      ۴      ۴      ۴
○ ۵      ۵      ۵      ۵      ۵      ۵      ۵      ۵
○ ۶      ۶      ۶      ۶      ۶      ۶      ۶      ۶
○ ۷      ۷      ۷      ۷      ۷      ۷      ۷      ۷
○ ۸      ۸      ۸      ۸      ۸      ۸      ۸      ۸
    
```

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

```

۱      ۱      ۱      ۱      ۱      ۱      ۱      ۱
۲      ۲      ۲      ۲      ۲      ۲      ۲      ۲
۳      ۳      ۳      ۳      ۳      ۳      ۳      ۳
۴      ۴      ۴      ۴      ۴      ۴      ۴      ۴
۵      ۵      ۵      ۵      ۵      ۵      ۵      ۵
۶      ۶      ۶      ۶      ۶      ۶      ۶      ۶
۷      ۷      ۷      ۷      ۷      ۷      ۷      ۷
۸      ۸      ۸      ۸      ۸      ۸      ۸      ۸
    
```

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

```

۱      ۲
۲      ۲
    
```



۳ ۳ ۳  
 ۴ ۴ ۴ ۴  
 ۵ ۵ ۵ ۵ ۵  
 ۶ ۶ ۶ ۶ ۶ ۶  
 ۷ ۷ ۷ ۷ ۷ ۷ ۷  
 ۸ ۸ ۸ ۸ ۸ ۸ ۸ ۸

✓ 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 (استهلاك 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 ۲۰۰، ۵۰۰ و 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 ۱۰\*I ۱۰۰\*I ۱۰۰۰\*I  
 ۱ ۱۰ ۱۰۰ ۱۰۰۰

۲	۲۰	۲۰۰	۲۰۰۰
۳	۳۰	۳۰۰	۳۰۰۰
۴	۴۰	۴۰۰	۴۰۰۰
۵	۵۰	۵۰۰	۵۰۰۰
۶	۶۰	۶۰۰	۶۰۰۰

✓ 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. به‌عنوان مثال، ورودی به‌صورت زیر می‌باشد:

۸۰۰ ۲۷۰ ۲۰۰ ۱۷۰ ۱۵۰ ۱۰۰ ۱۰۰ ۶

✓ برنامه میانگین و مجموع اعداد ۸۰۰، ۲۷۰، ۲۰۰، ۱۷۰، ۱۵۰، ۱۰۰، ۱۰۰ و ۶ 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 به‌صورت زیر محاسبه می‌شود:

- در طول ترم دو کوئیز برگزار می‌گردد 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.

✓ برنامه‌ای که نمرات کوئیز میان ترم و پایان ترم یک دانشجو را گرفته، نمره کل او را محاسبه می‌کند و به‌صورت حروفی نمایش می‌دهد (برای نمرات بزرگ‌تر یا مساوی ۹۰، حروف A، نمرات بزرگ‌تر یا مساوی ۸۰ و کم‌تر از ۹۰ حرف B، نمرات بزرگ‌تر یا مساوی ۷۰ و کم‌تر از ۸۰ حرف C، نمرات بزرگ‌تر یا مساوی ۶۰ و کم‌تر از ۷۰، حرف D و نمرات زیر ۶۰ حرف E را نمایش دهد). (توضیح: در این برنامه، ابتدا نمرات کوئیز را به ۱۰۰ تبدیل می‌کنیم. یعنی، نمره کوئیز اول + نمره کوئیز دوم را در ۵ ضرب می‌کنیم. سپس برای محاسبه نمره کل، نمره به‌دست 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 نمایم تا نمره کل به ۱۰۰ به دست آید.)

- ✓ برنامه‌ای که تعیین می‌کند در یک فروشگاه آیا میزان خرید مشتری از سقف اعتبار حسابش بیش تر شده است یا خیر. برای هر مشتری اطلاعات زیر موجود می‌باشد:
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 - هزینه‌ها + تراز اولیه) را محاسبه و تعیین کند که آیا تراز جدید از سقف اعتباراتش 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.

✓ برنامه‌ای که  $n$  را خوانده، علاوه بر نمایش عبارت زیر، مجموع عبارت را نیز نمایش می‌دهد:

$$مجموع = 1 + 2 + 3 + \dots + n$$

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

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

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

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

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

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

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

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

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

• 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.

✓ برنامه‌ای که نشان می‌دهد یک اسکناس ۱۰۰۰۰ تومانی را به چند طریق می‌توان با اسکناس‌های ۵۰۰۰، ۲۰۰۰ و ۱۰۰۰ تومانی خرد کرد، به طوری که از هر اسکناس حداقل یکی در خرد کردن استفاده شود.

✓ 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:

```
Enter a even number: ۹
***
*
*
*
*
*
*
*
*
*
***
```

✓ 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
...
10 20 30 40 50 ... 100
    
```

✓

✓ ۵۲. 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 تعیین می‌کند چند عدد چهار رقمی با ارقام ۵، ۶، ۷ و ۸ (بدون تکرار ارقام) به صورت نزولی می‌توان تولید کرد.

✓ 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  $(۲+۵+۹)/۲$  است. میانگین ارقام زوج آن  $(۴+۶)/۲$  می‌باشد.

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

✓ برنامه‌ای که کلیه اعداد سه‌رقمی 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 یک‌بار تکرار شده باشد را نمایش می‌دهد.

✓ برنامه‌ای که یک جمله را خوانده، تعداد ارقام ۰ تا ۹ آن را می‌شمارد.

✓ 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:

100	200	300	400	500	600
150	250	350	450	550	650
200	300	400	500	600	700
...	...	...	...	...	...
500	600	700	800	900	1000

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

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

✓ برنامه‌ای که  $x$  و  $n$  و  $k$  را خوانده، حاصل عبارت  $F(x, n, k) = \frac{\sum_{j=n}^0 x^j}{\sum_{i=1}^k i!}$  را نمایش می‌دهد.

✓ برنامه‌ای که  $n$  (عددی بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. به‌عنوان مثال، اگر برای  $n$  عدد ۵ وارد شود، خروجی به‌صورت زیر نمایش دهد:

✓

*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*

✓ برنامه‌ای که  $n$  (عددی بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر برای  $n=5$  وارد شود، خروجی را به‌صورت زیر نمایش می‌دهد) یعنی، فقط چهار ضلع را نمایش می‌دهد:

✓

*✓	*	*	*	*
*				*
*				*
*				*
*	*	*	*	*

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی مانند زیر را نمایش می‌دهد. (اگر  $n=5$  وارد شود، خروجی را به‌صورت زیر نمایش می‌دهد). یعنی اضلاع مربع و قطر اصلی را نمایش می‌دهد:

✓

*	*	*	*	*
*	*			*
*		*		*
*			*	*
*	*	*	*	*

✓ 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 (اگر  $n=5$  وارد گردد، خروجی زیر را نمایش می‌دهد):

✓

*	*	*	*	*
*			*	*
*		*		*
*	*			*
*	*	*	*	*

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. ( اگر  $n = ۵$  وارد شود، خروجی زیر را نمایش می‌دهد): یعنی، اضلاع، قطر اصلی و قطر فرعی را نمایش می‌دهد.

```

✓
* * * * *
* *   * *
*     *   *
* *     * *
* * * * *
    
```

خ ✓

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. ( اگر  $n = ۷$  باشد، خروجی زیر را نمایش می‌دهد): یعنی، ستون اول، سطر آخر و قطر اصلی را نمایش می‌دهد: خروجی: ✓

```

Enter n: ۵
*
**
* *
* *
*****
    
```

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی مانند شکل زیر را نمایش می‌دهد. ( اگر  $n = ۷$  باشد، خروجی زیر را نمایش می‌دهد). یعنی، سطر اول، ستون آخر و قطر اصلی را نمایش می‌دهد:

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده و خروجی مانند زیر را نمایش می‌دهد. ( اگر  $n = ۵$  وارد شود، خروجی زیر را نمایش می‌دهد). یعنی، سطر اول، ستون اول 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 آخر و قطر فرعی را نمایش می‌دهد. ( اگر  $n = ۵$  وارد شود، خروجی زیر 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. ( اگر n=۵ وارد شود، خروجی زیر را نمایش می‌دهد):

```

      *
     * *
    * * *
   * * * *
  * * * * *
  
```

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. ( اگر n=۵ باشد، خروجی 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، عناصر روی و زیر قطر فرعی را نمایش می‌دهد. ( اگر n = ۵ وارد شود، خروجی زیر را نمایش می‌دهد):

```

      *
     * *
    * * *
   * * * *
  * * * * *
  
```

✓ 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. (اگر n = ۵ وارد گردد، خروجی زیر را نمایش می‌دهد):

```

* * * * *
* * *
* *
*
*
    
```

✓ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. (اگر n = ۵ وارد شود، خروجی زیر را نمایش می‌دهد):

```

* *
* * *
* * *
* * *
* * *
    
```

✓ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر n = ۵ وارد شود، خروجی زیر را نمایش می‌دهد):

```

* * * * *
* * *
* * *
* * *
* * *
    
```

✓ برنامه‌ای که عدد n (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر n = ۷ وارد شود، خروجی زیر را نمایش می‌دهد):

✓ Enter n: ۷

```

✓ ** **
✓ *** ***
✓ *****
✓ *** ***
✓ ** **
    
```

```

• *
• *
    
```

✓ خروجی:

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی مثل زیر را نمایش می‌دهد. (اگر  $n = 5$  وارد شود، خروجی زیر را نمایش می‌دهد):

```

* * * * * ✓
      * * *
        *
          * * *
* * * * *
    
```

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر  $n = 5$  وارد شود، خروجی زیر را نمایش می‌دهد):

```

* ✓
* *
* * *
* *
*
    
```

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر  $n = 5$  وارد شود، خروجی زیر را نمایش می‌دهد):

```

      * ✓
      * *
* * *
      * *
      *
    
```

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر  $n = 5$  وارد شود، خروجی زیر را نمایش می‌دهد):

```

✓
* * * * *
      * * *
          *
    
```

✓ برنامه‌ای که عدد  $n$  (بین ۱ تا ۸۰) را خوانده، خروجی زیر را نمایش می‌دهد. (اگر  $n = 5$  وارد شود، خروجی زیر را نمایش می‌دهد):

```

      * ✓
      * * *
* * * * *
    
```

✓ نامه‌ای 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:

	✓		1 2 4 8 16 64	64 16 8 4 2 1
--	---	--	------------------------------	------------------------------

✓ 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 داده شده‌اند. برنامه‌ای که داده‌هایی از قبیل قیمت خانه، هزینه سوخت سالانه و مالیات سالانه را دریافت می‌کند. این برنامه هزینه ۱۰ ساله خانه را محاسبه می‌نماید. برنامه تا زمانی که کاربر به‌جای قیمت خانه ۹۹۹- وارد کند، ادامه می‌یابد.

✓ برنامه‌ای که عدد  $n$  را خوانده، مقدار  $\sqrt{n + \sqrt{n + \sqrt{n + \dots}}}$  را تا دقت  $10^{-10}$  محاسبه می‌کند و نمایش می‌دهد.

✓ در یک بازی دونفره  $n$  چوب‌کبریت وجود دارد. هر یک از بازیکنان می‌توانند در نوبت خود، ۱، ۲ یا حداکثر ۳ چوب‌کبریت را بردارند. بازیکنی که آخرین چوب‌کبریت را برمی‌دارد، بازنده است. برنامه‌ای که این بازی را بین دو بازیکن اجرا می‌کند. این

برنامه، ابتدا تعداد چوب کبریت‌ها را خوانده و در هر مرحله تعداد چوب کبریت‌های باقی‌مانده را نمایش می‌دهد.

✓ برنامه‌ای که اگر کاربر هر یک از حروف A, B, C و D را وارد کند، پیام Big Character! را نمایش می‌دهد. 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 می‌دهد. وگرنه پیام Unknown Character! را نمایش می‌دهد.

✓ برنامه‌ای که خروجی زیر نمایش می‌دهد:

```

■ *****
■ *****
■ *****
■ *****
○ *****
○ *****
○ *****
○ *****
○ *****
● *****
● *****
● *****
● *****
● *****
● *****
■ *****
■ *****
■ *****
■ *****
■ *****

```

✓ برنامه‌ای که n را خوانده، خروجی مانند زیر نمایش را می‌دهد:

```

✓ Enter n: ۷
○ *
○ **
○ ***
○ ****
○ *****
○ *****
○ *****

```

✓ 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

✓ 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  $1$  تا  $x$  می‌باشد.

✓ 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.

✓ برنامه‌ای که معدل  $n$  دانشجو را خوانده، میانگین معدل‌های بالاتر از  $15$  را نمایش می‌دهد.

✓ برنامه‌ای که حاصل سری زیر را تا  $4$  رقم اعشار محاسبه کرده، نمایش می‌دهد:

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

✓ برنامه‌ای که ابتدا تعداد دانشجویان ( $n$ ) را خوانده، سپس  $n$  معدل را می‌خواند و تعداد دانشجویایی که معدل آن‌ها بین  $10$  تا  $15$  است را شمارش می‌کند و نمایش می‌دهد.

✓ 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.

$$\blacksquare \quad 1, 5, 3, 4, 3, 5, \dots$$

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

✓ 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 برابر آن عدد شود.

به‌عنوان 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:

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

✓ ۱۱۸. برنامه‌ای که عددی را خوانده، ارقام زوج آن را نمایش می‌دهد. به‌عنوان مثال، برای عدد ۱۳۵۲۷۴، عدد ۲۴ را نمایش خواهد داد.

✓ ۱۱۹. 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. به‌عنوان مثال،  $n=2$  حاصل به شکل زیر خواهد بود:

✓ اما، برای  $n=6$ ، حاصل به شکل زیر می‌باشد:

```

      *
     *
    * *
   * * *
  * * * *
 * * * * *
* * * * *

```

✓ ۱۲۰. برنامه‌ای که یک عدد خوانده، در خروجی یک عدد را نمایش دهد که هر رقم آن یک واحد اضافه‌شده است. به‌عنوان مثال، اگر کاربر عدد ۴۵۹۳۴ را وارد کند، در خروجی ۵۶۰۴۵ چاپ شود (رقم ۹ به صفر تبدیل گردد). در این برنامه متغیرهای  $n$  (عدد ورودی)،  $result$  (عدد نتیجه پس از افزایش یک واحدی هر رقم)،  $pow$  (توان‌های از ۱۰) و ۲ (هر رقم عدد قبل و بعد از افزایش یک واحدی) را داریم.



✓ ۱۲۱. برای محاسبه  $\pi$  می توان از رابطه زیر استفاده کرد:

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

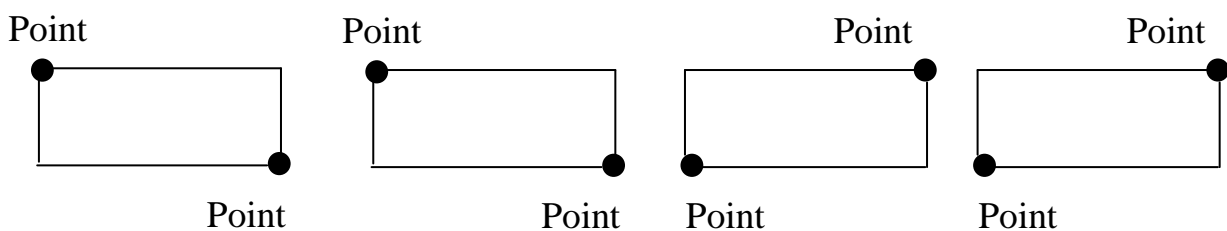
✓ برنامه ای که تعداد جملات سری فوق (n) را به عنوان یک عدد صحیح دریافت کرده و مقدار  $\pi$  را با استفاده از این رابطه محاسبه کرده و نمایش می دهد.

✓ برنامه ای که مختصات سه نقطه را از ورودی بگیرد و تعیین کند 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 می کند. 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.

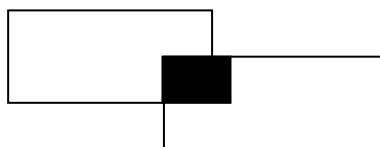
همچنین، معادله خط AB به صورت  $y - y_1 = m(x - x_1)$  نوشته می شود.

✓ برنامه ای که مختصات دو مستطیل را از ورودی گرفته و مساحت ناحیه ی مشترک بین دو مستطیل را حساب می نماید. به ازاء هر مستطیل دونقطه (جمعاً چهار عدد اعشاری) باید گرفته شود. توجه کنید که هر مستطیل با استفاده از دونقطه می تواند به یکی از چهار فرم زیر نمایش داده شود.



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

در این برنامه temp متغیر کمکی است.



✓ برای هر کسر به صورت  $1/k$  (k عدد صحیح بزرگ‌تر از صفر و کوچک‌تر از ۱۰۰۰)، دو عدد صحیح x و y وجود دارد که عبارت زیر برقرار باشد:

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

برنامه‌ای که مقدار k را از دریافت می‌کند و تمام x و yهایی 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

✓

$$\frac{1}{12} = \frac{1}{13} + \frac{1}{156}$$

✓

$$\frac{1}{12} = \frac{1}{14} + \frac{1}{84}$$

$$\frac{1}{12} = \frac{1}{15} + \frac{1}{60}$$

$$\frac{1}{12} = \frac{1}{16} + \frac{1}{48}$$

$$\frac{1}{12} = \frac{1}{18} + \frac{1}{36}$$

$$\frac{1}{12} = \frac{1}{20} + \frac{1}{30}$$

$$\frac{1}{12} = \frac{1}{24} + \frac{1}{24}$$

$$\frac{1}{12} = \frac{1}{21} + \frac{1}{28}$$

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

$$x + e^x = 5$$

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

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 به طوری که هر حرف بزرگ، مشابه کوچک آن هم نمایش داده شود. در این برنامه ch کاراکترهای 'A' تا 'Z' می‌باشد.

✓ برنامه‌ای که تعدادی عدد را که به صورت مرتب شده 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.

✓ برنامه‌ای که دو عدد را خوانده و حاصل عبارت زیر را در بازه بین این دو عدد محاسبه کرده، نمایش می‌دهد.

✓ ۱۳۰. 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:

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

✓ 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 اندازه  $n$  نقاط را به صورت اعشاری  $(x, y)$  می‌خواند و معادله  $y = mx + b$  را به دست آورد.  $m$  و  $b$  صورت‌های زیر محاسبه می‌شوند:

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

$$b = \bar{y} - m\bar{x}$$

$\bar{x}$  و  $\bar{y}$  به ترتیب میانگین  $x$  و  $y$  هستند. در این برنامه متغیرهای  $n$  (تعداد نقاط)،  $x$  و  $y$  (مختصات هر نقطه در هر مرحله)،  $\text{sum}X$  (مجموع مقادیر  $x$ )،  $\text{sum}Y$  (مجموع مقادیر  $y$ )،  $\text{sum}XY$  (مجموع  $x*y$ )،  $\text{mean}X$  (میانگین  $x$ )،  $\text{mean}Y$  (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 سطرها ۴ خروجی زیر را نمایش می‌دهد):

۱

۲ ۳

۴ ۵ ۶

✓ برنامه‌ای که تعداد سطرها را خوانده و خروجی را با الگوی زیر نمایش می‌دهد (اگر تعداد سطرها ۴ باشد):

```
    ۱
   ۲  ۳
  ۴  ۵  ۶
```

✓ 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:

```
    ۱
   ۲  ۲
  ۳  ۳  ۳
 ۴  ۴  ۴  ۴
```

✓ 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.

✓ برنامه‌ی که  $x$  و  $n$  را خوانده، حاصل عبارت زیر را محاسبه می‌کند:

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

✓ . برنامه‌ای که تعیین می‌کند کم‌ارزش‌ترین بیت ۱ است یا ۰ (کم‌ارزش‌ترین بیت، سمت

راست‌ترین بیت می‌باشد). به‌عنوان مثال، برای عدد ۱۵ (۰۰۰۰۱۱۱۱)، سمت راست‌ترین

بیت ۱ است، ولی برای عدد ۱۲ (۰۰۰۰۱۱۰) سمت راست‌ترین بیت ۰ است.

برای این منظور عدد خوانده شده را با ۰۰۰۰۰۰۰۱ (یک)، و بیتی می‌کند، چنانچه نتیجه صفر

شود، سمت چپ‌ترین بیت صفر، ولی اگر نتیجه یک شود سمت راست‌ترین بیت یک است:

✓ . برنامه‌ای که تعیین می‌کند باارزش‌ترین بیت یک عدد یک (۱) است یا صفر (۰)؟

باارزش‌ترین بیت عدد سمت چپ‌ترین بیت آن است.

(۱۲) ۰۰۰۰۱۱۰۰

(۱۵) ۰۰۰۰۱۱۱۱

(۱) \_\_\_\_\_

(۱) \_\_\_\_\_

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی مانند زیر را نمایش می‌دهد

(rows= ۵).

```
*****
*****
*****
*****
*****
*****
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی زیر را نمایش می‌دهد (rows=5):

```
*****
*      *
*      *
*      *
*      *
*****
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد. (rows=5):

```
*****
*****
*****
*****
*****
*****
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی زیر را نمایش می‌دهد (rows=5):

```
*****
*      *
*      *
*      *
*      *
*****
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد.  
:(rows=۵)

```

*****
****   ****
***     ***
**       **
*         *
*         *
**       **
***     ***
****   ****
*****
    
```

✓

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد.  
:(rows=۵)

```

*****
****
***
**
*
**
***
****
*****
    
```

✓

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی مانند زیر را نمایش می‌دهد.  
:(rows=۵)



```
*****
*****
***
**
*
**
***
****
***** ✓
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده و خروجی مانند زیر را نمایش می‌دهد  
:(rows=۵)

```
+
+
+
+
+++++++
+
+
+
+
```

✓ برنامه‌ای که تعداد سطرها را خوانده (rows) و خروجی مانند زیر را نمایش می‌دهد.  
:(rows=۵)

```
***
*   *
*   *
*   *
***
*   *
*   *
*   *
***
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند زیر را نمایش می‌دهد  
:(rows=۱۰)



```
01010
01010
01010
01010
01010
```

✓ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را از ورودی خوانده، خروجی نظیر زیر را نمایش می‌دهد (rows=۵ و cols=۵):

```
11111
10001
10001
10001
11111
```

✓ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی مانند زیر را نمایش می‌دهد:

```
10101
01010
10101
01010
10101
```

✓ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی نظیر زیر را نمایش می‌دهد:

```
11011
11011
00000
11011
11011
```

✓ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده و خروجی نظیر زیر را نمایش می‌دهد:

```
10001
01010
00100
01010
10001
```

✓ برنامه‌ای که تعداد سطرها (rows) و تعداد ستون‌ها (cols) را خوانده، خروجی مانند زیر را نمایش می‌دهد:

```
12345
23455
34555
45555
55555
```

✓ برنامه‌ای که با خواندن تعداد سطرها (rows) و تعداد ستون‌ها (cols) خروجی نظیر زیر را نمایش می‌دهد:

```
۱۲۳۴۵۶
۱۲۳۴۵۶۶
۱۲۳۴۵۶۶۶
۱۲۳۴۵۶۶۶۶
۱۲۳۴۵۶۶۶۶۶
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، خروجی مانند الگوی زیر را نمایش می‌دهد. (rows=۵).

```
555555555
544444445
543333345
543222345
543212345
543222345
543333345
544444445
555555555
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده و الگوی نظیر خروجی زیر را نمایش می‌دهد (rows=۵):

```

1
2 3
4 5 6 7
8 9 1 2 3 4 5 6
7 8 9 1 2 3 4 5 6 7 8 9 1 2 3 4
    
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده و الگوی مانند خروجی زیر را نمایش می‌دهد (rows=۵):

```

1           1
12          21
123         321
1234        4321
1234554321
    
```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، الگوی نظیر زیر را چاپ می‌نماید (rows=۵):

```

1
2   6
3   7   10
4   8   11   13
5   9   12   14   15
    
```

✓ برنامه‌ای که تعداد سطرها را خوانده و الگوی مانند خروجی زیر را نمایش می‌دهد:

```

1
2   4
7   11   16
22  29   37   46
56  67   79   92   106
    
```

✓ برنامه‌ای که تعداد سطرها را خوانده، الگوی نظیر زیر را نمایش می‌دهد (rows=۵):

```

1
3 2
4 5 6
10 9 8 7
11 12 13 14 15

```

✓ برنامه‌ای که تعداد سطرها (rows) را خوانده، الگوی نظیر خروجی زیر را نمایش می‌دهد. (rows=۵):

```

1
22
333
2222
11111

```

✓ برنامه‌ای که عددی را خوانده و بر اساس عدد خوانده شده، خروجی طبق الگوی زیر را نمایش می‌دهد:  
اگر عدد ورودی 22464 باشد، خروجی به صورت زیر باشد:

```

22464
2246
224
22
2

```

✓ برنامه‌ای که اگر عدد 24165 را از ورودی بخواند، خروجی با الگوی زیر را نمایش می‌دهد:

```

24165
4165
165
65
5

```

✓ برنامه‌ای که ابتدا n را خوانده و سپس الگوی مانند خروجی زیر را نمایش می‌دهد (n=۵):

```
1
12
123
1234
12345
1234
123
12
1
```

✓ برنامه‌ای که n را خوانده و الگوی خروجی نظیر زیر را نمایش می‌دهد (n=۵):

```
1
123
12345
1234567
123456789
1234567
12345
123
1
```

✓ ۱۷۴. برنامه‌ای که n را خوانده و الگوی مانند خروجی زیر را نمایش می‌دهد (n=۵):

```
*
*1*
*121*
*12321*
*1234321*
*123454321*
*1234321*
*12321*
*121*
*1*
*
```

✓ برنامه‌ای که ابتدا  $n$  را خوانده و الگویی نظیر خروجی زیر ( $n=5$ ) را نمایش می‌دهد:

```

1           1
 2         2
   3       3
    4     4
     5
    4     4
   3       3
  2         2
 1           1
    
```

✓ برنامه‌ای که عددی را خوانده، مکان باارزش‌ترین بیت آن را برمی‌گرداند و نمایش می‌دهد.

مکان باارزش‌ترین، سمت چپ‌ترین بیت آن یک (۱) است. به‌عنوان مثال عدد ۲۲ چنانچه در ۸ بیتی نمایش داده شود، به‌صورت زیر می‌شود: ۰۰۰۱۰۱۱۰

✓ برنامه‌ای که عددی را خوانده، مکان سمت راست‌ترین بیتی که ۱ باشد را نمایش می‌دهد. به‌عنوان 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 تعداد حروف الفبا را خوانده و الگوی نظیر خروجی زیر را نمایش می‌دهد ( $n=4$ ):

```

      A
     A B A
    A B C B A
   A B C D C B A
  
```

✓ برنامه‌ای که عددی را خوانده، تشخیص دهد عدد زشتس (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  $n$  عدد تشکیل تصاعد حسابی را می‌دهند یا خیر.

✓ دو سری از اعداد که تشکیل تصاعد حسابی را می‌دهند در زیر آمده‌اند:

$$\begin{array}{cccccccc} -1 & 3 & 7 & 11 & 15 & 19 & \dots\dots & \\ \bullet & 5 & 10 & 15 & 20 & 25 & 30 & \dots\dots \end{array}$$

✓ 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:

$$\begin{array}{cccc} 1 & & & \\ \frac{1}{2} & \dots\dots 1,25 & 2,5 & 5 & 10 \\ 2 & & & & \\ 3 & \dots\dots 54 & 18 & 6 & 2 \end{array}$$

✓ . 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 تک‌رقمی باشند، تعداد مثلث‌هایی که می‌توان تشکیل داد را نمایش می‌دهد. و 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 تک‌رقمی) باشند، برنامه‌ای که تعداد مثلث متساوی‌الساقین که با این اعداد می‌توان تشکیل داد را نمایش می‌دهد.

✓ اگر اضلاع مثلث اعداد تک‌رقمی باشند، تعداد مثلث‌های قائم‌الزاویه که می‌توان تشکیل داد را نمایش می‌دهد. 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  $a^2 = b^2 + c^2$  باشد)

✓ برنامه‌ای که مقدار سه زاویه را برحسب درجه دریافت کرده را تشخیص می‌دهد که آیا زاویه تشکیل مثلث را می‌دهند یا نه؟

✓ برنامه‌ای که عددی را خوانده، تعیین می‌کند که آیا توانی از ۲ است یا خیر؟ اعداد توان ۲ عبارت‌اند از:

✓ ... ۶۴ ۳۲ ۱۶ ۸ ۶ ۴ ۲

✓ برنامه‌ای که عددی را خوانده، تعیین می‌کند که آیا توانی از ۳ است یا خیر؟ اعداد توان ۳ عبارت‌اند از:

✓ ... ۲۷ ۸۱ ۲۴۳ ۹

✓ برنامه‌ای که دو عدد صحیح را خوانده، چک می‌کند که آیا یک عدد توانی از عدد دیگر است یا خیر؟

✓ 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.

به‌عنوان مثال (عدد ۱۴۵ = ۱! + ۴! + ۵!) 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. اعداد زیر Armstrong هستند:

$$6^1 = 6$$

$$371 = 3^3 + 7^3 + 1^3 = 371$$

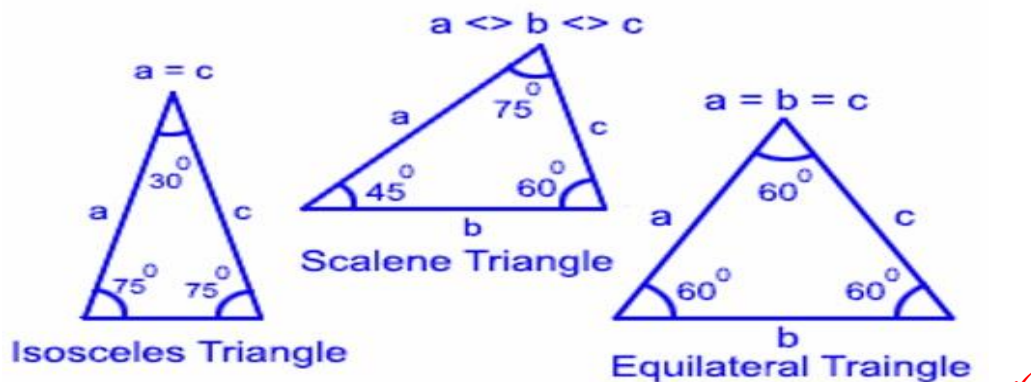
✓ برنامه‌ای که درجه حرارت را به سانتی‌گراد خوانده و بر اساس جدول زیر پیغام مناسب چاپ می‌کند.

پیغام	درجه حرارت
Freezing Weather	Or < ۰
Very Cold Weather	۰-۱۰
Cold Weather	۱۰-۲۰
Normal	۲۰-۳۰
Hot	۳۰-۴۰
Very Hot	> =۴۰

✓ 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:

$$1100101 = 1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 0 \times 2^3 + 0 \times 2^4 + 1 \times 2^5 + 1 \times 2^6 = 1 + 0 + 4 + 0 + 0 + 32 + 64 = 103$$

✓ 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، و گرنه 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 به‌صورت 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 که با ارقام فرد می‌توان نوشت را نمایش می‌دهد (در این برنامه هیچ یک از ارقام عدد تکراری نیستند).

✓ برنامه‌ای که تمام اعداد سه‌رقمی را تولید می‌کند که رقم صدگان 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 مرغ‌هایتان 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 عدد n را خوانده، n امین عدد زیبا را نمایش می‌دهد. اعداد زیبا به صورت زیر هستند:

۱ ۲ ۲ ۳ ۳ ۳ ۴ ۴ ۴ ۴ ۵ ۵ ۵ ۵ ۵ ۶ ۶ ۶ ۶ ۶ ۶ ۷ ۷ ۷ ۷ ۷  
۷ ۷ ...

✓ 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 این‌گونه محاسبه می‌کنیم:

$$\circ \text{ (قدیم) } \times \frac{x}{t} + \text{ (قدیم) } \times 0,5 = \text{ جدید } t$$

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 خوانده 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.

✓ 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.

✓ پروژه برنامه‌نویسی ۱: برنامه‌ای که روزی از سال را خوانده، مشخص می‌کند که چه فصلی، چه ماه و چه روزی از سال است. به‌عنوان 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.

✓ 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 به‌طور 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 پالس‌های 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 متر مکعب آب با توجه به نرخ مصرف از قبیل مصارف خانگی، تجاری یا صنعتی فرق می‌کند. برنامه‌ای که میزان مصرف آب به متر مکعب 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 متر مکعب مصرف ۵۰۰ ریال دریافت می‌شود.

اگر نوع مصرف حرف I یا i وارد شود (مصرف صنعتی تا چهار میلیون متر مکعب)، به ازای هر ۱۰۰۰ متر مکعب آب ۷۵۰ ریال و به ازای هر متر مکعب بیش از آن  $0.0025$  به مبلغ قبض اضافه می‌گردد.

اگر نوع مصرف حرف E یا e وارد شود (مصرف تجاری)، در صورتی که مصرف آب تا دو میلیون متر مکعب باشد، به ازای هر ۱۵۰۰ متر مکعب مصرف ۶۰۰ ریال و به ازای هر متر مکعب بیش از دو میلیون  $0.0004$  به مبلغ صورت حساب اضافه خواهد شد.

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





✓ 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 دو عدد را از ورودی می‌خواند و به متدی ارسال می‌کند. متد تفاضل حاصل ضرب و حاصل تقسیم آن‌ها را محاسبه کرده، به برنامه برمی‌گرداند و برنامه اصلی این حاصل را نمایش می‌دهد.

✓ 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.

✓ برنامه‌ای که n را خوانده و به روش بازگشتی حاصل سری زیر را محاسبه می‌کند:

$$\circ \sum_{n=1}^{\infty} \frac{1}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots$$

✓ قیمت سهام معمولاً به صورت کسری بیان می‌شود، مثلاً  $2\frac{7}{8}$  یا  $8\frac{1}{2}$ . 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 سالی را دریافت کرده، تعیین می‌کند که آیا کیسه است یا خیر؟

✓ 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. به‌عنوان مثال، اگر کاربر عدد ۲۲ را وارد کند، خروجی به‌صورت زیر نمایش داده شود.

$۲۲ = ۱۱ + ۱۱$        $۲۲ = ۱۷ + ۵$        $۲۲ = ۱۹ + ۳$

✓ یونانیان باستان اعداد را به‌صورت هندسی دسته‌بندی می‌کردند. به‌عنوان مثال، آن‌ها یک عدد را مثلثی می‌نامیدند که می‌توانستند با آن تعدادی ریگ (به‌اندازه 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

۶۴ ۴۹  
۸۱  
۱۲۱ ۱۰۰  
...

✓ 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، حرف C، وگرنه حرف F را نمایش می‌دهد).

✓ 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):

درصد مالیات	تا	از
0	483000	0
10%	600000	483001
15%	1000000	600001
20%	2000000	1000001
30%	99999999	2000001

✓ 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 عددی را خوانده، تعیین می‌کند که آیا توانی از ۳ است یا خیر؟ اعداد توان ۳ عبارت‌اند از:

۹ ۲۷ ۸۱ ۲۴۳ ... ○

✓ 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 × عرض × 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:

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

✓ 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:

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

$$\frac{x}{1} - \frac{x^2}{2}$$

✓ برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل  $n/2$  جمله عبارت زیر را 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.

✓ 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  $n$ ام را محاسبه می‌کند:  

$$\frac{d(n)}{dn}(e^{ax})$$
  
 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 جدول بندی 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:

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

✓ 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

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

✓ 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، 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:

$$F(x, y) =$$

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 به عنوان پارامتر دریافت می کند و تمام اعداد فرد بین این دو عدد را نمایش می دهد. برنامه ای که از این متد استفاده می کند.

✓ موسسه ای قصد دارد  $x$  درصد به حقوق کارمندان اضافه کند. متدی که ضریب افزایش حقوق و حقوق فعلی کارمند را دریافت کرده، حقوق جدید را برمی گرداند. برنامه ای که برای  $n$  کارمند از این متد استفاده می کند.

✓ متدی که دو عدد را به عنوان پارامتر دریافت کرده، تعیین می کند چند عدد اول بین این دو عدد وجود دارند. این برنامه، متدی برای تعیین عدد اول دارد. برنامه ای که از این متدها استفاده می کند. " عدد اول است که بر هیچ عدد کوچک تر از نصف خودش به جزء یک بخش پذیر نباشد."

✓ متدی که تأثیر نیروی جاذبه بر اجسام در حال سقوط را محاسبه می کند. برنامه ای که زمان را دریافت کرده (بر حسب ثانیه) و این تأثیر را محاسبه می نماید و چاپ می کند. این برنامه باید ارتفاع شیء سقوط کننده در هر ثانیه را نمایش می دهد. برای این منظور از فرمول های زیر استفاده می نماید:

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

✓ برنامه ای که عدد  $n$  را خوانده، سپس یک عدد دیگر  $k$  را بخواند و حاصل ضرب اعداد کوچک تر یا مساوی عدد  $n$  که مضرب آن عدد  $k$  باشد را نمایش می دهد.

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

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

✓ برنامه ای که  $B, C$  و  $\text{Alpha}$  را خوانده، با استفاده از متدی مقدار  $A$  را محاسبه می کند و نمایش می دهد.

✓ برنامه ای که دو عدد، سپس یک کاراکتر را خوانده، اگر کاراکتر  $R$  یا  $r$  باشد، مساحت مستطیل را با یک متد محاسبه می کند. چنانچه کاراکتر  $T$  یا  $t$  را وارد نماید، مساحت مثلث را با فرمول ارتفاع \* نصف قاعده محاسبه می نماید. در نهایت مساحت مستطیل یا مثلث را نمایش می دهد.

✓ رض کنید اساتید دارای حقوق کم تر از ۴۰۰۰۰۰۰ تومان ۰/۵ درصد اضافه حقوق می گیرند، ولی اساتید بین ۴۰۰۰۰۰۰ تا ۷۰۰۰۰۰۰ تومان ۱۲ درصد و اساتید با حقوق بیش تر ۱۰ درصد اضافه حقوق می گیرند. متدی که اضافه حقوق استاد را محاسبه می کند. برنامه ای که حقوق استادی را خوانده، اضافه حقوق و حقوق جدید استاد را نمایش می دهد.

✓ برنامه ای که  $n$  و  $m$  را خوانده، توسط متد بازگشتی  $C_n^m$  را برگرداند، نمایش می دهد:



$$C_n^m =$$

✓ 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.

$$A(m, n) =$$

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

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

نمایش می‌دهد:

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

$$f'(x_0)$$

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

$$f''(x_0) = \lim_{h \rightarrow 0} \frac{f(x_0+h) - 2f(x_0) + f(x_0-h)}{h^2}$$

✓ ۷۸. برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

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

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

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

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

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

۸۱. برنامه‌ای که اعداد  $x$  و  $n$  را خوانده، حاصل سری زیر را محاسبه کرده و نمایش می‌دهد: ✓

$$\begin{aligned} & \frac{x^0 0^n}{-n} + \frac{x^1 (1)^{n-1}}{-(n-1)} + \dots + \frac{x^{(n-1)} (n-1)^1}{-1} + \\ & \frac{x^n n^1}{1} + \frac{x^{(n+1)} (n+1)^2}{2} + \dots + \frac{x^{(2n-1)} (2n-1)^n}{n} \end{aligned}$$

۸۲. برنامه‌ای که  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد: ✓

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

۸۳. برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد: ✓

$$\frac{\tan x}{\sin(x^n)} + \frac{\tan^2(x)}{\sin(x^{n-1})} + \frac{\tan^3 x}{\sin(x^{n-2})} + \dots + \frac{\tan^n x}{\sin(x)}$$

۸۴. برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل عبارت زیر را نمایش می‌دهد: ✓

$$\left\langle \frac{\log(\log(\log \dots \log(\log(x))))}{n} \right\rangle$$

۸۵. برنامه‌ای که n را خوانده، حاصل سری زیر را نمایش می‌دهد: ✓

$$\frac{1!}{1^2} + \frac{2!}{1^2 + 2^3} + \frac{3!}{1^2 + 2^3 + 3^4} + \dots + \frac{n!}{1^2 + 2^3 + \dots + n^{n+1}}$$

۸۶. برنامه‌ای که x و n را خوانده، حاصل سری زیر را نمایش می‌دهد: ✓

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots \pm \frac{(-1)^n x^{2n}}{(2n)!}$$

۸۷. برنامه‌ای که x (بین -۱ تا ۱) و n را خوانده، حاصل سری زیر را نمایش می‌دهد: ✓

$$\arcsin x = x + \frac{2!x^3}{4!(1!)^2(3)} + \frac{4!x^5}{4^2(2!)^3(5)} + \dots + \frac{(2^n)!x^{2n+1}}{4^n(n!)^2(2n+1)}$$

۸۸. برنامه‌ای که n را خوانده، حاصل سری زیر را نمایش می‌دهد: ✓

$$\frac{\sqrt{1}}{1} - \frac{1!}{1^2} + \frac{\sqrt{2}}{2} - \frac{2!}{2^2} + \frac{\sqrt{3}}{3} - \frac{3!}{3^2} + \frac{\sqrt{n}}{n} - \frac{n!}{n^2}$$

۸۹. برنامه‌ای که n را خوانده، حاصل عبارت زیر را نمایش می‌دهد: ✓

$$\frac{\sqrt{3} - \sqrt{2}}{\sqrt{2^2 + 2}} + \frac{\sqrt{4} - \sqrt{3}}{\sqrt{3^2 + 3}} + \frac{\sqrt{n+1} - \sqrt{n}}{\sqrt{n^2 + n}}$$

۹۰. برنامه‌ای که n را خوانده، حاصل عبارت زیر را نمایش می‌دهد: ✓

$$\frac{1}{2^1(\log 2)^2} + \frac{1}{3^2(\log 3)^3} + \dots + \frac{1}{n^{n-1}(\log n)^n}$$

۹۱. برنامه‌ای که n را خوانده، حاصل عبارت زیر را نمایش می‌دهد: ✓

$$\log \frac{2}{2^2 - 1} + \log \frac{3^2}{3^3 - 1} + \dots + \log \left( \frac{n^{n-1}}{n^n - 1} \right)$$

۹۲. برنامه‌ای که n را خوانده، حاصل عبارت زیر را نمایش می‌دهد: ✓

$$\frac{\sqrt{5}}{(2^3 + 2)^2} + \frac{\sqrt{7}}{(3^3 + 3)^2} + \dots + \frac{\sqrt{2n+1}}{(n^3 + n)^2}$$

۹۳. برنامه‌ای که n را خوانده، حاصل عبارت زیر را محاسبه می‌کند و نمایش می‌دهد: ✓

$$\frac{\sin(\frac{1}{2})}{\cos(\frac{1}{1}) \cdot \cos(\frac{1}{2})} + \frac{\sin(\frac{1}{6})}{\cos(\frac{1}{2}) \cdot \cos(\frac{1}{3})} + \dots + \frac{\sin(\frac{1}{n(n+1)})}{\cos(\frac{1}{n}) \cdot \cos(\frac{1}{n+1})}$$

✓ ۹۴. برنامه‌ای که  $x$  و  $n$  را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$\frac{\sqrt{\sin(n) \cdot \log(2!)}}{\tan^n(1)} + \frac{\sqrt{\sin(n-1) \cdot \log(3!)}}{\tan^{n-1}(2)} + \dots + \frac{\sqrt{\sin(0) \cdot \log(n+2)!}}{\tan^n(n+1)}$$

✓ ۹۵. متدی که  $x$  و  $n$  را به‌عنوان پارامتر دریافت کرده، حاصل سری زیر را برمی‌گرداند. این برنامه از این متد استفاده می‌کند.

$$\frac{\tan(x^1) + \tan(x^2) + \dots + \tan(x^n)}{\sin^1(x) + \sin^2(x) + \dots + \sin^n(x)}$$

✓ ۹۶. متدی  $n$  را به‌عنوان دریافت کرده، سپس حاصل سری زیر را محاسبه و نمایش می‌دهد. برنامه از این متد استفاده می‌کند.

$$(n-1)! \cdot 1^{(n-1)} + (n-2)! \cdot 2^{(n-2)} + \dots + 0! \cdot n^{(n-n)}$$

✓ ۹۷. متدی که عدد صحیح  $n$  را به‌عنوان پارامتر دریافت کرده، حاصل سری زیر را محاسبه می‌کند و برنامه اصلی آن را نمایش می‌دهد:

$$\frac{1}{n-1}x + \frac{n-2}{2}x + \dots + \frac{n-1}{1}x$$

✓ ۹۸. برنامه‌ای که  $n$  را خوانده با استفاده از متدی حاصل سری زیر را محاسبه و نمایش می‌دهد:

$$\frac{1 * 2 * 3}{(n+1)(n+2)(n+3)} + \frac{2 * 3 * 4}{(n+2)(n+3)(n+4)} + \dots + \frac{n(n+1)(n+2)}{(n+n)(n+n+1)(n+n+2)}$$

✓ ۹۹. برنامه‌ای که  $x$  و  $n$  را خوانده و توسط متدهای حاصل سری زیر را محاسبه کرده، نمایش می‌دهد:

$$1 + \frac{nx^1}{1!} + \frac{n(n-1)x^2}{2!} + \dots + \frac{n!x^n}{n!}$$

✓ ۱۰۰. برنامه‌ای که n را خوانده و با استفاده از متدی حاصل سری زیر را محاسبه کرده و نمایش می‌دهد:

$$\phi = \left\{ \begin{array}{l} 1 \\ 1 + \frac{1}{1+1} \\ \dots \\ 1 + \frac{1}{1+1} \end{array} \right.$$

جمله n

✓ ۱۰۱. برنامه‌ای که عدد n را خوانده، کلیه اعداد مضرب ۷ زیر n که یکی از ارقام آن ۷ باشد را نمایش می‌دهد. در این برنامه برای نمایش اعداد مضرب ۷ یک متد و برای تعیین این که یک رقم عدد ۷ است از متد دیگری استفاده می‌شود.

✓ ۱۰۲. برنامه‌ای که عدد n را خوانده، کلیه اعداد مضرب ۷ زیر n که هیچ یک از ارقام آن ۷ نباشد را نمایش می‌دهد.

✓ ۱۰۳. 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 تعیین کند که اعداد از راست به چپ دو تا کم شدند از یک متد استفاده می‌کند. به‌عنوان مثال، برای اعداد ۲۴۶، ۳۵۷، ۱۳۵، "Yes" و برای اعداد ۱۲۴، ۳۵۶، 'No' را نمایش می‌دهد.

✓ برنامه‌ای که n را دریافت کرده، n امین عدد کاتالان را نمایش می‌دهد. N امین کاتالان با فرمول زیر محاسبه می‌شود:

$$C_n = \frac{1}{n+1} \binom{2n}{n} = \frac{2n!}{(n+1)!(n!)^2} = \prod_{k=2}^n \binom{n+k}{k}$$

برای  $n \geq 0$ ، ۱، ۲، ۵، ۱۴، ۴۲، ۱۳۲، ۴۲۹، ۱۴۳۰، ...

✓ پروژه برنامه‌نویسی ۱. برنامه‌ای که تمامی جفت‌های اعداد متحابه از یک تا عدد خوانده شده 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 آن‌ها عدد ۲۴۸ می‌شود و مقسوم‌علیه‌های ۲۴۸ که کوچک‌تر از ۲۴۸ هستند، برابر با ۱، ۲، ۴، ۷۱ که مجموع آن‌ها ۲۲۰ است. پس این دو عدد متحابه می‌باشند.

✓ پروژه برنامه‌نویسی ۲. 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:

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

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

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

✓ پروژه برنامه‌نویسی ۳. 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 فروخته شده 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 انتخاب شده 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 ظاهر شده گزینه ۵ را انتخاب نماید.

$$\frac{x^n}{x!}$$

$$x$$

$$\frac{((x)^n)^3(n)^{x+}}{n! + (x + n + 2)}$$

✓ 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 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. این برنامه ابتدا میانگین مقادیر را محاسبه می‌کند (با یک متد). سپس، با استفاده از فرمول زیر انحراف معیار را محاسبه کرده، چاپ می‌نماید (برای محاسبه انحراف معیار نیز از یک متد استفاده می‌کند):

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

a میانگین اعداد و N تعداد اعداد است.

✓ در یک دوره مسابقات ورزشی تعدادی تیم (حداکثر ۱۰۰ تیم) شرکت کرده‌اند، می‌خواهیم در انتهای مسابقه گزارشی از مجموع امتیازات هر تیم را چاپ کنیم. برنامه‌ای که برای هر مسابقه در یک خط ورودی شامل آیتیم‌های برنده و امتیاز کسب‌شده را می‌خواند و نتیجه را نمایش می‌دهد. اگر به جای تیم برنده ۱- وارد شد، برنامه خاتمه می‌یابد.

✓ برنامه‌ای که حداکثر ۱۰ عدد را از ورودی خوانده، می‌گوید هر عدد چند بار تکرار شده است. به عنوان مثال، اگر ورودی به صورت زیر باشد:

۱۴۳ ۱۴۳ ۷۰ ۱۴۳ ۷۰ ۱۴۳

خروجی زیر را چاپ می‌کند:

۱۴۳ OCCURS ۳ TIMES, AT POSITIONS ۲ ۴ ۶  
۷۰ OCCURS ۲ TIMES, AT POSITIONS ۱ ۳

✓ برنامه‌ای ماتریس  $A[3 \times 4]$  را از ورودی می‌خواند و ترانزاده آن را در  $B$  ذخیره می‌کند. در ترانزاده باید برای هر  $i, j$  رابطه مقابل برقرار باشد:  
 $B_{i,j} = A_{j,i}$

✓ برنامه‌ای که انداختن دو تاس را شبیه‌سازی می‌نماید. این برنامه برای انداختن تاس‌ها از متد `randint` استفاده می‌کند که عدد بین ۱ تا ۶ را تولید می‌نماید. سپس مجموع این دو مقدار محاسبه می‌گردد. از آنجایی که تاس‌ها مقادیر بین ۱ تا ۶ را نشان می‌دهند، پس مجموع این دو مقدار بین ۲ تا ۱۲ است که مجموع ۷، بیش‌ترین دفعات و مجموع ۱۲ کم‌ترین دفعات پیش می‌آیند. جدول زیر ترکیبات ممکن برای این دو تاس را نشان می‌دهد. این برنامه، ۳۶۰۰۰ بار تاس می‌ریزد و با استفاده از یک آرایه، تک‌بعدی تعداد 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 چندجمله‌ای زیر را در یک آرایه نمایش می‌دهد (سپس،  $x$  را می‌خواند) و عبارت  $p$  را محاسبه کرده، چاپ می‌نماید ( $n$  حداکثر ۱۰۰ باشد).  
 $p = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x^1 + a_0$



✓ برنامه‌ای که تعدادی عدد را خوانده، تشخیص می‌دهد مرتب‌شده هستند یا خیر؟

✓ برنامه‌ای که n عدد را خوانده، در آرایه قرار می‌دهد. سپس، توسط متدی عنصر nام را حذف می‌کند.

✓ برنامه‌ای که دو آرایه m عنصری و n عنصری را خوانده، n عنصر اول آرایه دوم را به انتهای آرایه اول اضافه می‌کند.  
متدی برای خواندن عناصر آرایه نوشته‌شده است.  
متدی برای اضافه کردن عناصر آرایه دوم در آرایه اول نوشته گردید.  
متدی برای چاپ اول و آرایه دوم نوشته‌شده است.

✓ برنامه‌ای که عناصر یک آرایه و مقدار x را خوانده، تعداد تکرار عنصر x در آرایه a را می‌شمارد. در این برنامه متدها زیر نوشته‌شده است:  
متدی برای خواندن عناصر آرایه a  
متدی برای شمارش تعداد تکرار x در آرایه a

✓ برنامه‌ای که دو آرایه n عنصری یک‌بعدی را خوانده، حاصل ضرب بیرونی n عنصر اول آرایه a با n عنصر اول آرایه b را برمی‌گرداند (حاصل ضرب این دو آرایه، در آرایه دیگری به نام c قرار می‌گیرد). به‌عنوان مثال، فرض کنید آرایه‌های 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.

✓ برنامه‌ای که عناصر یک آرایه دوبعدی را ۹۰ درجه چرخش می‌دهد. به‌عنوان مثال، اگر آرایه زیر را داشته باشیم:

۱۱	۲۲	۳۳
۴۴	۵۵	۶۶
۷۷	۸۸	۹۹

با چرخش ۹۰ درجه آرایه به‌صورت 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 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 $z = i$ باشد (یعنی، شماره سطر برابر شماره ستون باشد) و عناصری روی قطر فرعی قرار دارند که $z = i + j - 1$ 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 یک‌بار 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،  $i - 1 == z + 1$  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 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، عنصری 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 خوانده‌شده به سمت چپ چرخش می‌دهد و چاپ می‌کند.

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

✓ 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:  
مقدار اولیه همه عناصر آرایه را برابر یک قرار می‌دهد.  
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.  
در پایان عنصری که هنوز یک هستند، شماره اندیس‌هایشان اول اند.  
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.

✓ همان‌طور که می‌دانید فاکتوریل اعداد بزرگ در یک `double` جا نمی‌شود. برنامه‌ای که فاکتوریل اعداد بزرگ را به کمک آرایه محاسبه می‌کند و نمایش می‌دهد.

✓ برنامه‌ای که دو آرایه  $3 \times 4$  به نام‌های  $x$  و  $y$  را خوانده، حاصل عبارت زیر را در آرایه دیگر به نام  $z$  قرار می‌دهد:

$$Z = 2x - 3y$$

✓ برنامه‌ای که بین تعدادی عدد که در آرایه قرار دارند، بزرگ‌ترین، کوچک‌ترین مقدار و مکان‌های آن‌ها را پیدا کرده، نمایش می‌دهد.

✓ برنامه‌ای که ۲۰ عدد تصادفی (دو سری اعداد تصادفی ۱۰ تایی) ایجاد کرده، در دو آرایه قرار می‌دهد. سپس یک عدد را خوانده، تمام عناصر آرایه دوم را در آرایه اول از مکان وارد شده درج می‌نماید.

✓ برنامه‌ای که تعداد اعداد تصادفی تولید کرده، در آرایه  $4 \times 4$  قرار می‌دهد. سپس هر سطر آرایه را جداگانه مرتب می‌نماید. در پایان، آرایه‌ای که عناصر سطرهای آن مرتب‌شده‌اند را نمایش می‌دهد.

✓ برنامه‌ای که آرایه ۵ عنصری را خوانده، به مقدار هر یک از عناصر آرایه توسط علامت ستار (\*) میله افقی رسم می‌کند.

✓ برنامه‌ای که  $n$  عدد را خوانده، سپس دو عدد دیگر را می‌خواند و از این دو عدد خوانده‌شده، عدد دوم را جایگزین عدد اول در آن هفت عدد می‌نماید.

✓ برنامه‌ای که عدد صحیح  $n$  (کوچک‌تر از ۹) را از ورودی خوانده، خروجی زیر را با استفاده از آرایه دوبعدی (ماتریس) تولید و چاپ می‌کند:

1	2	3	...	$n$
2	3	4	...	$n+1$
...	...	...	...	...
$n+1$	$n+2$	$n+3$	...	$2n$

✓ 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  $(n*m)/2$  از عناصر آن صفر باشند.

✓ ۵۵. برنامه‌ای که یک ماتریس را می‌خواند و تعیین می‌کند که آیا متقارن است یا خیر؟ ماتریسی متقارن است که برابر ترانژاده خودش باشد. یعنی  $A=A^T$  به‌عنوان 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 تکرار هر رقم استفاده شده در آن را نمایش می‌دهد.

✓ برنامه‌ای که یک آرایه با  $n$  عنصر و سه عدد  $x$ ،  $y$  و  $z$  را خوانده 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 آرایه‌ای با  $n$  عنصر که مقادیر آن‌ها ۰ یا یک با ترتیب تصادفی هستند را خوانده، آن‌ها را به سمت چپ و آن‌ها را به سمت راست آرایه منتقل کرده، نمایش می‌دهد. به‌عنوان مثال، اگر آرایه ورودی به‌صورت زیر باشد:

$$\bullet [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$$

آرایه خروجی به‌صورت زیر خواهد بود:

$$\bullet [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]$$

✓ برنامه‌ای که آرایه‌ای را خوانده، سپس سه عدد  $x$ ،  $y$  و  $z$  را می‌خواند و ماکزیمم عبارت زیر را نمایش می‌دهد:

$$\max_{i \leq j \leq k} ((x * a[i]) + (y * a[j]) + (z * a[k]))$$

برای

به‌عنوان مثال، اگر آرایه  $a$  به‌صورت زیر باشد.

$$a[] = \{-1, -2, -3, -4, -5\}$$

و  $x=1$ ،  $y=2$  و  $z=-3$  باشد، آنگاه خروجی ۱۲ خواهد بود. زیرا:

$$(1 * -1) + (2 * -2) + (-3 * -5) = 12$$

✓ برنامه‌ای که یک آرایه را خوانده، دو عنصری را می‌یابد که حاصل جمع آن‌ها نزدیک‌ترین عدد به صفر است. برای انجام این برنامه باید عددی را پیدا کنیم که مینیم قدر مطلق مجموع دو عنصر آرایه می‌باشد.

✓ برنامه‌ای که آرایه‌ای را خوانده، ماکزیمم مقدار اختلاف بین هر دو عنصر آرایه را نمایش می‌دهد.

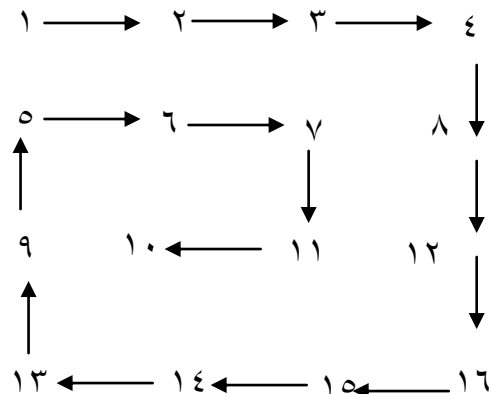
✓ برنامه‌ای که آرایه‌ای را خوانده، اعداد فرد و زوج آن را از هم جدا می‌کند.

✓ برنامه‌ای که آرایه‌ای را خوانده و ماکزیمم  $i$ -زا پیدا می‌کند به طوری که  $arr[j] > arr[i]$  باشد. سپس مقدار ماکزیمم را نمایش می‌دهد. به عنوان مثال، اگر آرایه به صورت زیر باشد:

$$\bullet \quad arr[] = \{34, 8, 10, 3, 2, 80, 30, 33, 1\}$$

آنگاه وقتی  $(j=7, i=1)$  باشد، مقدار ۶ را نمایش می‌دهد.

✓ برنامه‌ای که یک آرایه دوبعدی تعریف کرده و مقداردهی اولیه می‌کند. سپس اطلاعات آرایه را به صورت مارپیچی نمایش می‌دهد. به عنوان مثال، اگر آرایه دوبعدی به صورت زیر باشد:



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) یا (True) هستند)، اگر عنصر  $a[i][j]$  برابر یک (1) باشد، تمام عنصر سطر  $i$  ام و ستون  $j$  ام را برابر یک قرار می‌دهد. به‌عنوان مثال، اگر آرایه به‌صورت زیر باشد:

```
0 1 0
0 0 0
0 0 1
```

خروجی به‌صورت 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 وجود نداشته باشد، 1- را نمایش خواهد داد. به‌عنوان مثال، آرایه‌های زیر را در نظر بگیرید:

- $arr_1 = \{-10, -5, 0, 3, 7\}$
- $arr_2 = \{10, 2, 5, 8, 17\}$

برای  $arr_1$  خروجی 3 (چون  $arr_1[3]=3$  است) و برای  $arr_2$  خروجی 1- است.

✓ برنامه‌ای که آرایه‌ای را تعریف کرده و مقداردهی اولیه می‌کند به‌طوری که مقدار عناصر آن ابتدا افزایشی هستند (یعنی، عنصر قبلی کم‌تر از عنصر بعدی است) و سپس عناصر آن کاهشی می‌باشند. این برنامه بزرگ‌ترین مقدار این نوع آرایه را نمایش می‌دهد.

به‌عنوان مثال، اگر عناصر آرایه به‌صورت زیر باشند:

```
7      41    1    3    50    10
```

خروجی 50 خواهد بود.

✓ برنامه‌ای که یک آرایه را تعریف کرده، سپس مقداردهی اولیه می‌کند و در پایان، آرایه‌ای را ایجاد می‌کند که هر عنصر آن، تعداد عناصری کوچک‌تر از سمت راست خودش را شمارش می‌کند. به‌عنوان مثال، اگر آرایه به‌صورت زیر باشد:

•  $arr = \{12, 1, 2, 3, 0, 11, 4\}$

خروجی به‌صورت زیر خواهد بود:

○ ۶, ۱, ۱, ۱, ۰, ۱, ۰

✓ برنامه‌ای که آرایه‌ای را تعریف کرده، مقداردهی اولیه می‌کند. سپس یک عدد را خوانده و آرایه فرعی را نمایش می‌دهد که مجموع عناصر آن برابر عدد واردشده می‌باشد. اگر آرایه‌ای وجود نداشته باشد که مجموع عناصر آن برابر عدد واردشده باشد، پیغام "No SubArray Found" را نمایش می‌دهد. به‌عنوان مثال، اگر آرایه‌های ورودی به‌صورت زیر باشد:

✓ ۱۲ ۶ ۷ ۸ ۱۴ ۱۰

و اگر  $sum = 18$  باشد، آنگاه خروجی به‌صورت زیر است:

✓ Sum found between indexes ۰ and ۱

✓ برنامه‌ای که یک آرایه را تعریف کرده و مقداردهی اولیه می‌کند، سپس یک عدد را می‌خواند، توسط تابعی اگر مجموع سه عنصر آرایه برابر با عدد خوانده‌شده باشد، آن سه عدد را نمایش می‌دهد و True را برمی‌گرداند.

✓ برنامه‌ای که کوچک‌ترین عدد مثبت (غیر منفی) که از یک آرایه غیر مرتب حذف شده است را می‌یابد و نمایش می‌دهد. به‌عنوان مثال، اگر آرایه ورودی به‌صورت زیر باشد:

•  $\{2, 3, -7, 6, 8, 1, -10, 15\}$

خروجی عدد ۴ خواهد بود. ولی برای آرایه زیر خروجی ۲ خواهد شد:

○  $\{1, 1, 0, -1, -2\}$

✓ برنامه‌ای که آرایه‌ای را تعریف کرده، مقدار اولیه می‌دهد، سپس یک عدد را می‌خواند و زوج مقادیر از عناصر آرایه که اختلاف آن‌ها عدد واردشده باشد، را نمایش می‌دهد. به‌عنوان مثال، اگر عناصر آرایه به‌صورت زیر باشد:

○  $\{2, 20, 3, 2, 50, 80\}$

و عدد ورودی ۷۸ باشد، خروجی به‌صورت زیر است:



○ Pair Found: (۲, ۸۰)

✓ برنامه‌ای که آرایه‌ای را تعریف کرده، مقداردهی اولیه می‌کند، سپس هر عنصر را با بزرگ‌ترین عنصر بعد از خودش (سمت راست خودش) جایگزین می‌کند، به جای آخرین عدد ۱- قرار می‌گیرد و آرایه جدید را نمایش می‌دهد.

به‌عنوان مثال، اگر آرایه ورودی به صورت زیر باشد:

○ {۱۶, ۱۷, ۴, ۳, ۵, ۲}

آرایه خروجی به صورت زیر خواهد شد:

○ {-۱, ۲, ۵, ۵, ۵, ۱۷}

✓ برنامه‌ای که آرایه‌ای را تعریف کرده، مقداردهی اولیه می‌کند، سپس یک عدد را خوانده و مجموع ۴ عنصر آرایه که برابر با عدد خوانده شده باشد را نمایش می‌دهد. به‌عنوان 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 می‌دهند را شمارش می‌کند (سه مقدار از آرایه که مجموع هر دو مقدار بیش‌تر از مقدار سوم باشد، تشکیل مثلث را می‌دهند).

✓ پروژه برنامه‌نویسی ۱: یک شرکت 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.

✓ 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 می‌کند ماتریس بالا و پایین مثلثی است یا نه؟

تعیین می‌کند که آیا تمام عناصر هر سطر ماتریس برابرند یا نه؟

تعیین می‌کند که تمام عناصر هر ستون ماتریس برابرند یا نه؟

✓ پروژه برنامه‌نویسی ۵. فرض کنید بخواهیم وضعیت مدال‌های بین‌المللی یک کشور را در رشته ورزشی خاصی در سال‌های ۱۳۸۵ تا ۱۳۹۵ مطابق جدول زیر دریافت و پردازش کنیم:

سال	تعداد مدال طلا	تعداد مدال نقره	تعداد مدال برنز
۱۳۸۵			
۱۳۷۱			
۱۳۷۲			
۱۳۹۴			
۱۳۹۵			

- ✓ برنامه‌ای که ضمن دریافت داده‌های لازم (خانه‌های خالی طول) موارد زیر را نمایش می‌دهد.
- ✓ ۱- تعداد کل مدال‌های کسب‌شده در همه‌ی سال‌ها
- ✓ ۲- تعداد کل مدال‌های طلا کسب‌شده در همه‌ی سال‌ها
- ✓ ۳- شماره سال‌هایی که هیچ مدالی کسب نشده
- ✓ ۴- سالی که بیش‌ترین مدال کسب‌شده
- ✓ ۵- سالی که بیش‌ترین مدال طلا کسب‌شده است.
- ✓ ۶- امتیاز تمام سال‌ها 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.
- ✓ ۷- تعداد سال‌هایی که امتیاز بیش از ۱۰ کسب‌شده است.

✓ پروژه برنامه‌نویسی ۶. برنامه‌ای که دو آرایه  $4 \times 4$  تعریف کرده، مقادیر تصادفی بین ۰ تا ۲۰ را به اعضای آن‌ها تخصیص می‌دهد. سپس اعمال زیر را انجام می‌دهد (هدف از این برنامه آشنایی با عملگرهای +، -، \* برای ماتریس است).

حاصل جمع، ضرب، تفریق دو ماتریس را نمایش می‌دهد.

ترانزاده ماتریس‌های اول و دوم را نشان می‌دهد.

$$C = 3A - 2B^T$$

حاصل عبارت مقابل را حساب کرده، نمایش می‌دهد:

✓ برنامه‌ای که رشته و عدد n را خوانده، کلمات به طول n را حذف کرده و بقیه رشته را نمایش می‌دهد.

✓ برنامه‌ای که رشته‌ای را خوانده، آخرین کلمه آن را نمایش می‌دهد.

✓ برنامه‌ای که رشته‌ای را خوانده، تمام حروف اول کلمات آن را به حروف بزرگ تبدیل کرده و بقیه حروف کلمات را به حروف کوچک تبدیل می‌کند.

✓ برنامه‌ای که رشته‌ای را خوانده، کوچک‌ترین کلمه رشته را نمایش می‌دهد. برای پیدا کردن کوچک‌ترین کلمه رشته یک تابع بازگشتی به نام minimum() نوشته شده است که لیست کلمات را به‌عنوان پارامتر دریافت می‌کند و کوچک‌ترین کلمه آن را برمی‌گرداند.

✓ برنامه‌ای که یک کلمه را دریافت کرده، حرف نقره‌ای آن را نمایش می‌دهد. حرف نقره‌ای در یک کلمه، حرفی است که فاصله‌ی آن از دو طرف رشته مساوی باشد. مثلاً، در کلمه Ali، حرف ا نقره‌ای است. چون فاصله آن از دو طرف یکسان است. ولی کلمه Book حرف نقره‌ای ندارد. چون حرفی را نمی‌توان پیدا کرد که فاصله‌ی آن از دو طرف برابر باشد.

✓ برنامه‌ای که نامی را خوانده، معادل عددی نام را نمایش می‌دهد. معادل عددی یک نام برابر با مجموع معادل عددی هر کاراکتر موجود در نام است. کاراکتر 'a' معادل عددی ۱، کاراکتر 'b' معادل عددی ۲، کاراکتر 'c' معادل عددی ۳ و همین‌طور کاراکتر 'z' معادل عددی ۲۶ را دارد (معادل عددی کاراکترهای کوچک و بزرگ برابرند. یعنی معادل عددی 'A' و 'a' هر دو ۱ است). به‌عنوان مثال، اگر کاربر zelle را وارد کند، معادل عددی آن برابر است با:

$$\checkmark \quad ۲۶+۵+۱۲+۱۲+۵=۶۰$$

✓ برنامه‌ای که دو رشته S1 و S2 را از ورودی خوانده، رشته S1 را در رشته S2 جست‌وجو می‌کند. در این برنامه، خواندن رشته‌ها توسط متد اصلی و جست‌وجوی رشته توسط متدی انجام می‌شود.

- ✓ برنامه‌ای که رشته‌ای را در رشته دیگر کپی می‌کند (تابعی که کپی رشته را پیاده‌سازی می‌کند).
- ✓ برنامه‌ای که رشته‌ای را به انتهای رشته دیگر الحاق می‌کند (پیاده‌سازی تابع اتصال دو رشته).
- ✓ برنامه‌ای که رشته عددی را که حاوی نقطه اعشار است از ورودی خوانده، آن را به عدد اعشاری تبدیل کند. به‌عنوان 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 می‌شوند.
- ✓ 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. به‌عنوان 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.
- ✓ برنامه‌ای که رشته‌ای را از ورودی می‌خواند. تمام کلمات چهارحرفی آن را با کلمه "Love" جایگزین می‌کند. مثلاً رشته "hate you, you doer" به رشته "love you, you love" تبدیل می‌گردد.
- ✓ برنامه‌ای که یک رشته را خوانده، تعداد تکرار هر حرف را نمایش می‌دهد. به‌عنوان مثال، ACCURENCE را در نظر بگیرید که حرف C سه بار تکرار شده است.
- ✓ این برنامه ابتدا، رشته را می‌خواند. سپس، با استفاده از یک حلقه for کاراکترهای رشته خوانده‌شده را پیمایش می‌نماید. چنانچه رشته بین حروف A تا Z یا a تا z باشد، یک واحد به عنصر مربوط به آن کاراکتر اضافه می‌کند.  $ord(S[i]) - ۳۲ - ۶۵$  یا  $ch$   $[ord(S[i]) - ۶۵]$ . در پایان تعداد تکرار هر کاراکتر اگر، بزرگ‌تر از ۰ باشد، را نمایش می‌دهد.
- ✓ برنامه‌ای که یک رشته را خوانده تشخیص می‌دهد آیا متجانس است یا خیر؟ رشته‌ای متجانس است که از دو طرف یکی خوانده شود.
- ✓ 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 یک رشته حداکثر ۱۰۰ کلمه‌ای را خوانده، تمام کلمات آن را وارونه می‌کند و چاپ می‌نماید (بین کلمات کاراکتر blank (فاصله) قرار می‌گیرد). به‌عنوان مثال، Ali and Reza به‌صورت ila dna azeR تبدیل می‌شود.
- ✓ برنامه‌ای که یک رشته را خوانده، تعداد کلمات یک حرفی، دو حرفی، سه حرفی، چهار حرفی و غیره را می‌شمارد.
- ✓ برنامه‌ای که یک رشته را خوانده، کلیه جاهای خالی از قبیل کاراکتر blank، '\n' و '\t' رشته را حذف می‌کند و نمایش می‌دهد.
- ✓ برنامه‌ای که یک رشته را خوانده، سپس دو عدد را می‌خواند و از کاراکترهای عدد اول وارد شده تا عدد دوم وارد شده رشته ورودی را حذف می‌کند (یعنی، بخشی از رشته را حذف می‌نماید).
- ✓ برنامه‌ای که رشته‌ای را خوانده، با استفاده از یک متد فضای خالی سمت چپ (ابتدا رشته) را حذف می‌کند و نمایش می‌دهد.
- ✓ برنامه‌ای که رشته‌ای را از ورودی خوانده و بین حروف آن فاصله (Blank) ایجاد می‌کند. به‌عنوان مثال، اگر کاربر رشته Test را وارد کند، برنامه رشته Test را نمایش می‌دهد.
- ✓ 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 کاراکتر خوانده شده را قرار داده و رشته را نمایش می‌دهد.

✓ برنامه‌ای که رشته‌های  $S_1$  و  $S_2$  را خوانده و کاراکترهای آن‌ها را یک‌درمیان در رشته  $S_3$  قرار می‌دهد و رشته  $S_3$  را نمایش می‌دهد.

✓ برنامه‌ای که رشته‌ای را خوانده، کاراکتری از رشته که بزرگ‌ترین 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.

✓ ۴۰. برنامه‌ای که رشته‌ای را خوانده، سپس دو عدد را می‌خواند و با توجه به اعداد خوانده‌شده، حروف مکان‌های آن‌ها را جابجا می‌نماید.

✓ برنامه‌ای که تعدادی 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 به وسط رشته اول اضافه می‌کند.

✓ برنامه‌ای که رشته‌ای را خوانده و بعد از هر حرف 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 امضای زیر است:

- `String insert (string s1, string s2, int n);`

✓ 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 عدد مبنای ۲ را به صورت رشته خوانده، متمم یک آن را نمایش می‌دهد.  
 ✓ در این برنامه متغیرهای binary (رشته دودویی)، onescom (متمم یک رشته دودویی واردشده)، i (شمارنده برای پیمایش کاراکترهای رشته binary) و error (آیا رشته ورودی باینری است (مقدار ۰) یا خیر (مقدار ۰)؟)

✓ 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 ۲	معادل مبنای ۸	معادل باینری	معادل مبنای ۸
۰۰۱	۱	۰۰۰	۰
۰۱۱	۳	۰۱۰	۲
۱۰۱	۵	۱۰۰	۴
۱۱۱	۷	۱۱۰	۶

اگر عدد باینری ۰۱۱۰۱۱۰۰ باشد، معادل آن ۶(۱۰۰)، ۵(۱۰۱) و ۱(۰۱) است (یعنی معادل ۱۵۶)

✓ ۵۰. برنامه‌ای که عددی در مبنای ۲ را 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
۱۶	۲	۱۶	۲	۱۶	۲	۱۶	۲
۰	۰۰	۰	۰۰	۰	۰۰	۰	۰۰
۴	۰۱	۴	۰۱	۴	۰۱	۴	۰۱
۸	۱۰	۸	۱۰	۸	۱۰	۸	۱۰
C	۱۱	E	۱۱	D	۱۱	F	۱۱



به عنوان مثال، اگر عدد باینری ۱۰۱۱۱۱۱۰۱۰۱۰۱ باشد، معادل مبنای ۱۶ آن (۱۰۰۰۱) ۱۷۰۵ (۱۰۱=۵، ۱۱۰۱=E، ۰۱۱۱=۷) است.

✓ پروژه برنامه نویسی ۱. برنامه ای که رشته ای را خوانده، درصد کاراکتر، حرف کوچک، حروف بزرگ، ارقام و حروف صدا دار را محاسبه کرده، نمایش می دهد.

✓ پروژه برنامه نویسی ۲: برنامه ای که اعمال زیر را انجام می دهد (این برنامه توابعی برای انجام هر یک از اعمال دارد که به صورت جداگانه پیاده سازی شده است و از توابع آماده پایتون استفاده نمی کند).

تابعی به نام Lower دارد که کلیه حروف بزرگ رشته را به حروف کوچک تبدیل می کند.  
تابعی به نام Upper دارد که کلیه حروف کوچک رشته را به حروف بزرگ تبدیل می کند.  
تابعی به نام strLen دارد که طول رشته را برمی گرداند.  
تابعی به نام strCpy دارد که رشته ای را در رشته دیگر کپی می کند.  
تابعی به نام isDigits دارد که تعیین می کند آیا همه کاراکترهای رشته رقم هستند یا نه؟  
تابعی به نام isAlphas دارد که تعیین می کند آیا کلیه کاراکترهای رشته حروف الفبا هستند یا خیر؟

تابعی به نام right دارد که چند کاراکتر انتهای رشته را برمی گرداند.  
تابعی به نام left دارد که چند کاراکتر ابتدای رشته را برمی گرداند.  
تابعی به نام strCmp دارد که دو رشته را باهم مقایسه می کند.  
تابعی به نام substr دارد که بخشی از رشته را از نقطه خاص به طول n کاراکتر برمی گرداند.

تابعی به نام convertToNumber دارد که رشته عددی را به عدد صحیح تبدیل می کند و برمی گرداند.

تابعی به نام join دارد که چند رشته را با کاراکتر جای خالی باهم اتصال می دهد.

تابعی به نام insert دارد که رشته ای را در درون رشته دیگر درج می کند.

مجموعه‌ها، لیست‌ها و دیکشنری‌ها

- ✓ برنامه‌ای که یک دیکشنری ایجاد می‌نماید که کلیدهای آن "UPPER-CASE" و "LOWER-CASE" و مقادیر آن‌ها تعداد حروف کوچک و بزرگ می‌باشند. برنامه‌ای که رشته‌ای را خوانده، تعداد حروف بزرگ و کوچک آن را شمارش می‌کند.
- ✓ برنامه‌ای که یک لیست را گرفته، عناصر تکراری آن را یکتا کرده و نمایش می‌دهد.
- ✓ برنامه‌ای که یک لیست را گرفته، مجموع اعضای آن را نمایش می‌دهد (با استفاده از حلقه تکرار).
- ✓ برنامه‌ای که یک لیست را گرفته، بزرگ‌ترین مقدار آن را نمایش می‌دهد.
- ✓ برنامه‌ای که بررسی می‌کند آیا لیست خالی (تهی) است یا خیر؟
- ✓ برنامه‌ای که دو لیست را گرفته، تعیین می‌کند که آیا آن‌ها حداقل یک عضو مشترک دارند یا خیر؟
- ✓ برنامه‌ای که یک عنصر از لیست را به‌صورت تصادفی انتخاب می‌کند.
- ✓ برنامه‌ای که لیستی را دریافت کرده، دومین کوچک‌ترین مقدار آن را نمایش می‌دهد.
- ✓ برنامه‌ای که یک لیست را دریافت کرده، تمام زیرمجموعه‌های آن لیست را نمایش می‌دهد.
- ✓ برنامه‌ای که چک می‌کند آیا لیستی زیرمجموعه لیست دیگر است یا خیر؟
- ✓ برنامه‌ای که مقداری را قبل از هر عنصر لیست اضافه می‌کند.
- ✓ برنامه‌ای که تعیین می‌کند آیا تمام عناصر لیست با یک مقدار برابرند یا خیر؟  
✓ ['Red', 'Yellow']
- ✓ برنامه‌ای که آخرین عنصر لیستی را با لیست دیگر جایگزین می‌کند.
- ✓ برنامه‌ای که تمام مقادیر لیست را که از مقدار خاصی بیش‌تر باشند، نمایش می‌دهد.
- ✓ برنامه‌ای که بدون متد append() و حلقه تکرار یک لیستی را به انتهای لیست دیگر اضافه می‌کند.
- ✓ برنامه‌ای که دو دیکشنری را با هم ادغام می‌کند.
- ✓ برنامه‌ای که تمام مقادیر دیکشنری را با هم جمع می‌کند.

- ✓ برنامه‌ای که تمام مقادیر دیکشنری را باهم ضرب می‌کند.
- ✓ برنامه‌ای یک دیکشنری را مرتب می‌کند.
- ✓ برنامه‌ای که کوچک‌ترین و بزرگ‌ترین مقدار یک دیکشنری را نمایش می‌دهد.
- ✓ برنامه‌ای که دو دیکشنری را دریافت کرده، مقادیر دو دیکشنری که کلیدهای آن‌ها یکی باشد را با هم جمع می‌کند و دو دیکشنری را ترکیب می‌کند.
- ✓ برنامه‌ای که یک دیکشنری را دریافت کرده، سه مقدار بزرگ‌تر را به همراه کلید آن‌ها نمایش می‌دهد.
- ✓ برنامه‌ای یک دیکشنری را دریافت کرده، مقادیر آن‌ها را در لیست ترکیب می‌کند.
- ✓ برنامه‌ای که مقادیر کلید دو دیکشنری را با هم مقایسه می‌کند.
- ✓ برنامه‌ای که کمترین و بیشترین مقدار یک مجموعه را پیدا کرده، نمایش می‌دهد.
- ✓ برنامه‌ای که به روش بازگشتی مجموع عناصر لیست را حساب کرده، نمایش می‌دهد.
- ✓ برنامه‌ای که یک عدد حذف‌شده از لیست مرتب را پیدا کرده، نمایش می‌دهد.
- ✓ برنامه‌ای که یک لیست را دریافت می‌کند به طوری که همه اعضای آن به‌جز یک عضو ۲ بار تکرار شده‌اند، برنامه عضوی را که یک‌بار تکرار شده است، پیدا کرده و نمایش می‌دهد.
- ✓ برنامه‌ای که یک لیست را دریافت می‌کند که تمام اعضای آن به‌جز یک عضو سه بار تکرار شده‌اند، عضوی که سه بار تکرار نشده را نمایش می‌دهد.
- ✓ برنامه‌ای که ماتریسی را دریافت کرده، تست می‌کند که آیا تشکیل مربع جادویی را می‌دهد یا خیر؟ مربع جادویی، مربعی است که جمع سطرها و ستون‌ها و قطرهای آن یک عدد باشد (مانند شکل زیر):

8	1	6	→	15
3	5	7	→	15
4	9	2	→	15
↙	↓	↓	↓	↘
15	15	15	15	15

- ✓ برنامه‌ای که یک لیست را گرفته، یک آیتم از لیست را به صورت تصادفی انتخاب می‌کند و نمایش می‌دهد.



✓ برنامه‌ای که کلاسی به نام SavingAccount ایجاد می‌کند. این کلاس از متغیری به نام annualInterestRate برای ذخیره سازی نرخ سود همه حساب‌ها استفاده می‌کند. هر شی از این کلاس شامل متغیر خصوصی به نام SavingBalance می‌باشد که مقدار پس-انداز جاری سپرده را در خود نگه می‌دارد. متدی به نام CalculateModifyInterest() دارد که سود ماهانه را با فرمول زیر محاسبه می‌نماید:

$$\frac{(\text{SavingBalance} * \text{annualInterestRate})}{12}$$

اضافه می‌نماید. کلاس متد استاتیکی به نام SavingBalance سپس سود را به یک مقدار جدید annualInterestRate که به مقدار فیلد ModifyInterestRate تخصیص می‌دهد. برنامه‌ای که از این کلاس استفاده می‌کند. در این برنامه دو شیء به نام- ایجاد می‌کند و به هر یک تراز ۲۰۰۰۰۰ دلار و ۳۰۰۰۰۰ دلار نسبت saver۱ و saver۲ را ۴٪ قرار می‌دهد و سپس سود ماهانه را حساب می‌کند. مقدار محاسبه کرده، نمایش می‌دهد. سپس مقدار saver۱ و saver۲ و تراز را برای ۵٪ قرار می‌دهد و سود ماه بعد را حساب می‌نماید و تراز جدید annualInterestRate را savingBalance را برای هر یک از حساب‌ها نشان می‌دهد (تراز همان فیلد است).

✓ کلاسی به نام ChkNum که دو متد به نام‌های IsPrime() و LeastComFactor() دارد. متد IsPrime() پارامتری به نام x را دریافت کرده، تعیین می‌کند اول است یا خیر؟ متد LeastComFactor() دو پارامتر به نام‌های a و b را دریافت می‌کند و کوچکترین فاکتور مشترک بین آن‌ها را برمی‌گرداند (کوچکترین فاکتور مشترک، کوچکترین عدد اولی است که به هر دو پارامتر a و b بخش پذیر باشد).

✓ اگر طول دو ضلع (B و C) از مثلث و زاویه بین این دو ضلع Alpha باشد. می‌توان طول ضلع سوم را با فرمول زیر حساب کرد:

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

( و متدهای مساحت و A را دریافت می‌کند. طول ضلع سوم (Alpha و C، B کلاسی که محیط را حساب کرده، نمایش می‌دهد. مساحت و محیط مثلث به صورت زیر حساب می‌شوند.

$$\text{محیط} = (a + b + c) \quad p = \frac{\text{محیط}}{2} \quad \text{مساحت} = \sqrt{p(p-a)(p-b)(p-c)}$$

✓ برنامه‌ای که مجموع مساحت‌ها و محیط‌های چند مربع را نمایش می‌دهد (در این برنامه تعداد مربع‌ها و مقادیر اضلاع آن‌ها به صورت تصادفی تولید می‌گردند).

✓ برنامه‌ای که کلاسی به نام کارمند (Employee) پیاده‌سازی می‌کند که اعضای داده‌ای از قبیل کد، نام، نام خانوادگی و حقوق دارد. این کلاس اعضای متدی نظیر `__init__` (سازنده کلاس)، `__del__` (مخرب کلاس)، `calTax()` (برای محاسبه مالیات)، `calInsurance()` (برای محاسبه بیمه)، `Pay()` (برای محاسبه دریافتی یا حقوق خالص) و `__str__` (برای پیاده‌سازی متد `str`) دارد. بیمه برابر با ۱۰ درصد حقوق می‌باشد. اما، اگر حقوق کم‌تر یا مساوی ۲۳۰۰۰۰۰ تومان باشد، مالیات برابر صفر وگرنه اگر حقوق بین ۲۳۰۰۰۰۰ تا ۵۰۰۰۰۰۰ باشد، ۱۰ درصد مازاد بر ۲۳۰۰۰۰۰ تومان مبلغ مالیات می‌باشد. وگرنه اگر حقوق بالای ۵۰۰۰۰۰۰ باشد، ۱۰ درصد ۲۳۰۰۰۰۰ تا ۵۰۰۰۰۰۰ تومان به علاوه ۲۰ درصد مازد بر ۵۰۰۰۰۰۰ مالیات می‌باشد. حقوق خالص دریافتی برابر با (بیمه + مالیات) - حقوق است.

✓ برنامه‌ای که اندازه پارچه را به یارد دریافت کرده، به متر تبدیل می‌کند (هر متر ۱٫۱۹۶ یارد است).

✓ برنامه‌ای که قاعده و ارتفاع مثلثی را دریافت کرده، محیط و مساحت مثلث را حساب می‌کند. محیط و مساحت مثلث به صورت زیر محاسبه می‌گردد:

$$\text{محیط مثلث} = \text{ارتفاع} + \text{قاعده} + \sqrt{(\text{ارتفاع})^2 + (\text{قاعده})^2}$$

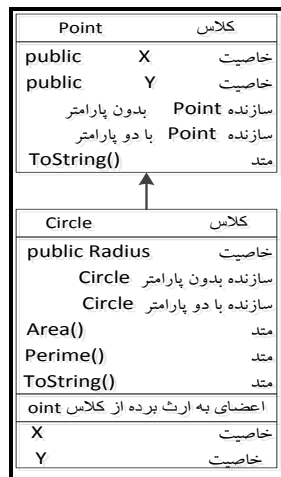
$$\text{مساحت مثلث} = \frac{\text{ارتفاع} \times \text{قاعده}}{2}$$

✓ برنامه‌ای که زمان را دریافت کرده و آن را به صورت Am (قبل از ظهر) یا Pm (بعد از ظهر) نمایش می‌دهد.

✓ برنامه‌ای که شعاع بزرگ (طولی) و شعاع کوچک (عرضی) بیضی را دریافت کرده، مساحت بیضی را با فرمول زیر محاسبه می‌کند:

$$\text{مساحت بیضی} = \pi * \text{شعاع کوچک} * \text{شعاع بزرگ}$$

✓ برنامه‌ای که کلاس‌های زیر را تعریف کرده و از آن‌ها استفاده می‌کند:



✓ برنامه‌ای که کلاس‌های مربع (Square) و مستطیل را پیاده‌سازی می‌کند. کلاس مستطیل از کلاس مربع مشتق می‌شود.

✓ کلاس استاتیکی که با دو متد درجه سانتی‌گراد را به فارنهایت و درجه فارنهایت را به سانتی‌گراد تبدیل می‌کند. برنامه‌ای که از این کلاس استفاده می‌کند. برای تبدیل درجه‌ها از فرمول‌های زیر استفاده می‌شود.

$$\text{درجه به فارنهایت} = ۳۲ + ۹/۵ * \text{درجه سانتی‌گراد}$$

$$\text{درجه به سانتی‌گراد} = ۵/۹ * (\text{درجه فارنهایت} - ۳۲)$$

در این برنامه منوی نمایش داده می‌شود تا نوع درجه را دریافت کند.

✓ برنامه‌ای که کلاس کسر (Fraction) را پیاده‌سازی کرده، عملگرهای +، -، \*، /، =، <، >، <=، >= و <= را برای آن پیاده‌سازی می‌کند (کسر دارای صورت (x) و مخرج (y) است).

فصل



فایل‌ها

- ✓ برنامه‌ای که نام فایل متنی را خوانده، سپس یک کاراکتر (حرف) را می‌خواند و تعداد تکرار حرف خوانده شده را در فایل شمارش کرده و نمایش می‌دهد.
- ✓ برنامه‌ای که نام فایلی را خوانده، تعیین می‌کند که آیا فایل یک فایل BMP است یا خیر؟ فایلی با نوع BMP است که بایت اول آن ۰x۴۲ و بایت دوم آن ۰x۴۰ باشد.
- ✓ برنامه‌ای که نام یک فایل متنی را از کاربر دریافت کرده، اطلاعات آن را بر روی صفحه نمایش نشان می‌دهد (بدون استفاده از حلقه تکرار).
- ✓ برنامه‌ای که نام یک فایل متنی را از کاربر دریافت کرده، اطلاعات آن را در یک لیست کپی کرده و اطلاعات لیست را نمایش می‌دهد.
- ✓ برنامه‌ای که نام یک فایل متنی را از کاربر دریافت کرده، کلماتی که بیش‌ترین طول را دارند، نمایش می‌دهد.
- ✓ برنامه‌ای که نام یک فایل متنی را از کاربر دریافت کرده، تعداد خطوط آن را نمایش می‌دهد.
- ✓ برنامه‌ای که نام یک فایل متنی را از کاربر دریافت کرده، تعداد تکرار هر کلمه را شمارش کرده و نمایش می‌دهد.
- ✓ برنامه‌ای که نام یک فایل متنی را از کاربر دریافت کرده، اطلاعات یک لیست را در آن می‌نویسد و سپس اطلاعات آن فایل را نمایش می‌دهد.
- ✓ برنامه‌ای که نام دو فایل را از کاربر دریافت کرده، اطلاعات یک فایل را در فایل دیگر کپی می‌کند (بدون استفاده از حلقه).
- ✓ برنامه‌ای که نام دو فایل را از کاربر دریافت کرده، اطلاعات دو فایل را با هم ترکیب کرده، بر روی صفحه نمایش نشان می‌دهد.
- ✓ برنامه‌ای که نام فایلی را از کاربر دریافت کرده، اطلاعات یک خط را به طور تصادفی انتخاب کرده و نمایش می‌دهد.



- ✓ برنامه‌ای که نام فایلی را از کاربر دریافت کرده، کاراکتر خطوط جدید ('\n') آن را حذف کرده و بقیه اطلاعات را نمایش می‌دهد.
- ✓ برنامه‌ای که نام و مسیر فایل فعلی در حال اجرا را نمایش می‌دهد.
- ✓ برنامه‌ای که به متغیرهای محیطی دست‌یابی پیدا می‌کند.
- ✓ برنامه‌ای که نام فایلی را خوانده، n سطر اول آن را نمایش می‌دهد.
- ✓ برنامه‌ای که نام فایلی را خوانده، n سطر آخر آن را نمایش می‌دهد.
- ✓ برنامه‌ای که نام فایلی را خوانده، اطلاعات آن را در آرایه‌ای قرار می‌دهد.
- ✓ برنامه‌ای نام فایلی را خوانده، اندازه آن را نمایش می‌دهد.
- ✓ برنامه‌ای که نام یک فایلی را خوانده، سپس دو عدد را می‌خواند و خطوطی از فایل که در این سطرها و بین آن‌ها قرار دارند را به صورت آرایه نمایش می‌دهد.
- ✓ برنامه‌ای که اطلاعات یک فایل را خوانده، و در خودش از آخرین خط به اولین خط می‌نویسد (اطلاعات فایل را برعکس می‌کند).
- ✓ برنامه‌ای که اطلاعات دو فایل را با هم تعویض می‌کند. یعنی، اطلاعات فایل اول را در فایل دوم و اطلاعات فایل دوم را در فایل اول کپی می‌کند.
- ✓ برنامه‌ای که نام فایلی را خوانده، تعیین می‌کند که آیا این فایل از نوع GIF است یا خیر؟ فایلی با نوع GIF است که بایت‌های اول تا چهارم آن "GIF۸" باشند.
- ✓ برنامه‌ای که نام فایلی را خوانده و اطلاعات آن فایل را به حروف بزرگ تبدیل کرده در خودش می‌نویسد.
- ✓ برنامه‌ای که نام یک فایل CSV را خوانده، کلمات آن را جدا کرده و هر کلمه را در یک سطر چاپ می‌کند. فایل‌های CVS، فایل‌هایی متنی هستند که کلمات آن با کاما از یکدیگر جدا می‌شوند.

✓ برنامه‌ای که نام یک فایل و عدد  $n$  را خوانده، سپس  $n$  سطر انتهای فایل را از انتها به ابتدا نمایش می‌دهد.

✓ برنامه‌ای که نام یک فایل و  $n$  را خوانده،  $n$  سطر ابتدای فایل را از انتها به ابتدا نمایش می‌دهد.

✓ برنامه‌ای که نام یک فایل و عدد  $n$  را خوانده و از  $n$  امین سطر فایل تا انتهای فایل را نمایش می‌دهد.

✓ برنامه‌ای که نام فایلی را خوانده، اطلاعات شماره سطرهای فرد آن را نمایش می‌دهد

✓ برنامه‌ای که نام چند فایل را از ورودی می‌خواند و اطلاعات آن فایل‌ها را در فایل kol.txt در پوشه Data درایو D ادغام می‌کند (برای خاتمه کاربر باید به جای نام فایل Enter را فشار دهد (یعنی نام فایل را وارد نکند)).

✓ برنامه‌ای که نام سه فایل را از کاربر می‌خواند و خطوط فایل‌های اول و دوم را یک دو میان در فایل سوم می‌نویسد (یعنی، یک خط از فایل اول و یک خط از فایل دوم و همین روند را ادامه می‌دهد).

✓ - برنامه‌ای که نام دو فایل را خوانده، سپس دو عدد را می‌خواند و اطلاعات فایل اول را در فایل دوم می‌نویسد، به طوری که خطوط بین بازه بسته بین این دو عدد را در فایل دوم نمی‌نویسد.

✓ برنامه‌ای که نام فایلی را خوانده، سپس یک کلمه را می‌خواند و تعداد تکرار این کلمه در فایل را شمارش می‌کند.