

تمرین ۱

(آشنایی با دریافت ورودی و قواعد نام گذاری)

صاحب یک فست فود، به دنبال فردی آشنا به پایتون می‌گردد تا فرآیند صدور فیش برای هر مشتری را اتوماتیک کند! ما شما را به او معرفی کرده‌ایم. برنامه‌ای بنویسید که تعداد پیتزا، ساندویچ، سیب‌زمینی و نوشیدنی سفارش داده شده توسط مشتری را دریافت کرده و با احتساب ۱۰ درصد ارزش افزوده، مبلغ قابل پرداخت را نمایش دهد. قیمت محصولات این فست‌فود به شکل زیر است:

| نام غذا | قیمت (هزار تومان) |
|-----------|-------------------|
| پیتزا | ۵۰ |
| ساندویچ | ۳۵ |
| سیب‌زمینی | ۳۰ |
| نوشیدنی | ۵ |

خروجی این برنامه مثلا برای مبلغ ۱۰۰ هزار تومان باید به شکل زیر باشد:

Please pay 100000 Toman

**قواعد نام‌گذاری متغیرها را حتما رعایت کنید.

تمرین ۲

(آشنایی با عملیات ریاضی)

یک مسابقه تنیس در حال برگزاری است. تنها دستگاهی که برای ثبت زمان مسابقه وجود دارد، فقط ثانیه را ثبت می‌کند. یعنی اگر مسابقه ۱۰ دقیقه طول بکشد، این ثانیه‌شمار عدد ۶۰۰ را ثبت می‌کند. یک مسابقه برگزار شده و این دستگاه عدد ۱۲۲۸۳ را ثبت کرده است. مسئولان مسابقه از شما خواسته‌اند که با پایتون این زمان را به ساعت، دقیقه و ثانیه برایشان دریاورد. برنامه‌ای بنویسید که مدت‌زمان مسابقه را به تفکیک ساعت، دقیقه و ثانیه نشان دهد.

****خروجی برنامه باید به شکل زیر باشد:**

xx hours, xx minutes, xx seconds

تمرین ۳

(آشنایی با الویت‌ها در عملیات ریاضی)

برنامه‌ای بنویسید که قد و وزن را دریافت کرده و شاخص BMI را محاسبه کند. شاخص BMI از فرمول زیر محاسبه می‌شود:

$$BMI = \frac{weight}{height^2}$$

****مقدار قد به متر و وزن به کیلوگرم باید وارد شود.**

****برای قد ۱۶۰ سانتیمتر و وزن ۵۹ کیلوگرم، خروجی باید ۲۳ باشد (مقدار اعشار را حذف کنید).**

تمرین ۴

(آشنایی با دستورات مربوط به لیست مانند sort, remove و ترتیب جمع کردن)

لیست زیر را در متغیری به نام my_list ذخیره کنید:

```
[[2], [0], [1, 3], [0, 7], [9, 11], [13, 15, 17]]
```

الف) لیست [13, 15, 17] را از my_list استخراج کرده، مقادیر آن را از بزرگ به کوچک مرتب کرده و سپس آن را به my_list برگردانید.

**خروجی باید به شکل زیر باشد:

```
[[2], [0], [1, 3], [0, 7], [9, 11], [17, 15, 13]]
```

ب) دو لیست آخر از my_list را جمع کرده و در ابتدای my_list اضافه کنید.

تمرین ۵

(آشنایی با اندیس‌دهی و اسلایسینگ)

در لیست زیر با استفاده از اندیس‌دهی 'd' را استخراج کنید.

```
list1 = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'k', 'l', 'm', 'n']
```

با استفاده از slicing لیست ['m', 'n'] را استخراج کنید.

تمرین ۶

(آشنایی با نحوه تعریف لیست)

بدون کدنویسی پایتون بگویید کدامیک از تعاریف زیر برای لیست درست است؟ چرا؟

- a) [1, 3, 4, 2]
- b) [[1, 2, 3], [4, 5, 7]]
- c) [1 + 2, 6 * 5, 3]

تمرین ۷

(آشنایی با sort, reverse)

برنامه‌ای بنویسید که سه المانی که از همه بزرگتر و از همه کوچکتر هستند را در یک لیست پیدا کرده و به شکل مناسبی نمایش دهد.

**مثلا برای لیست [۱۱, ۳, ۱۵, ۴, ۹, ۶, ۸, ۱, ۱۲, ۶, ۲۰, ۷, ۹] نتیجه باید به شکل زیر باشد:

largest three elements: [20, 15, 12]

smallest three elements: [1, 3, 4]

تمرین ۸

(آشنایی با دیکشنری)

هر رنگی در کامپیوتر، یک کد مخصوص به خود دارد. کامپیوتر رنگ‌ها را با این کدها می‌شناسد نه با نامشان! از شما خواسته شده برنامه‌ای بنویسید که اسم رنگ را دریافت و کد رنگی مربوط به آن رنگ نمایش داده شود. نام رنگ‌ها از جدول زیر انتخاب می‌شوند:

| Color | CSS Name | Code |
|---|----------|---------|
|  | Lime | #00FF00 |
|  | Blue | #0000FF |
|  | Yellow | #FFFF00 |
|  | Cyan | #00FFFF |
|  | Magenta | #FF00FF |
|  | Gray | #808080 |
|  | Maroon | #800000 |
|  | Green | #008000 |
|  | Purple | #800080 |
|  | Navy | #000080 |

تمرین ۹

(آشنایی با sort, concat)

دو لیست زیر را از کوچک به بزرگ مرتب کرده، سپس به هم بچسبانید و در متغیری به نام sorted_list ذخیره کنید.

```
[1, 3, 5, 7, 9, 11]
```

```
[0, 2, 4, 6, 8, 10]
```

تمرین ۱۰

(آشنایی با for)

برنامه‌ای بنویسید که یک عدد را دریافت کرده و تعداد تکرار آن عدد را در لیست زیر نمایش دهد:

```
ref = [0, 1, 3, 5, 0, 3, 4, 5, 0, 8, 0, 3, 6, 0, 3, 1, 1, 0]
```

تمرین ۱۱

(آشنایی با for, append)

برنامه‌ای بنویسید که sub-listهای متناظر را در دو لیست زیر به هم بچسباند و نتیجه را در یک لیست جدید نمایش دهد.

```
[['a', 'b'], ['b', 'c', 'd'], ['e', 'f']]
```

```
[['p', 'q'], ['p', 's', 't'], ['u', 'v', 'w']]
```

****نتیجه باید به شکل زیر باشد:**

```
[['a', 'b', 'p', 'q'], ['b', 'c', 'd', 'p', 's', 't'], ['e', 'f', 'u', 'v', 'w']]
```

تمرین ۱۲

(آشنایی با اندیس‌دهی)

در لیست زیر، جای دو المان p و s را با هم عوض کنید:

```
words = ['b', 'c', 'd', 'p', 's', 't']
```

تمرین ۱۳

(آشنایی با اندیس‌دهی، join , replace)

یک متغیر به نام my_string بسازید و عبارت زیر را در آن ذخیره کنید:

```
the quick brown fox jumps over the lazy dog
```

برنامه‌ای بنویسید که با سه کاراکتر اول و سه کاراکتر آخر از my_string، یک جمله جدید بسازد.

n-امین کاراکتر از my_string را حذف کند.

m-امین کلمه از my_string را حذف کند.

تعداد تکرار عبارت the در my_string را نمایش دهد.

تمرین ۱۴

(آشنایی با replace)

کدی بنویسید که در رشته "32.054,23" جای کاما و نقطه را عوض کند.

**خروجی کد بالا باید به شکل 32,054.23 باشد.

تمرین ۱۵

(آشنایی با for, if)

کدی بنویسید که تعداد حروف صدادار را در یک رشته محاسبه کند.

تمرین ۱۶

(آشنایی با if, len)

کدی بنویسید که اگر طول جمله زیر بیشتر از ۶ باشد، عبارت This sentence is too long در خروجی نمایش داده شود.

sentence = "the quick brown fox jumps over the lazy dog"

تمرین ۱۷

(آشنایی با اندیس‌دهی معکوس)

کدی بنویسید که چک کند یک رشته palindrome است یا خیر. palindrome به عبارتی گفته می‌شود که از اول و آخر به یک شکل نوشته می‌شود مثل anna

** برای اینکه یک رشته را در پایتون معکوس کنید می‌توانید از اندیس‌دهی به شکل زیر استفاده کنید (step را منفی یک در نظر بگیرید).

a[begin:end:step]

تمرین ۱۸

(آشنایی با if)

نمره یک دانش‌آموز در متغیری به نام grade ذخیره شده است. کدی بنویسید که اگر نمره بالای ۹۰ بود، عبارت 'A' در خروجی نمایش داده شود. به grade مقادیر مختلف بدهید و کد خود را تست کنید.

تمرین ۱۹

(آشنایی با if , elif , else)

کد تمرین قبلی را طوری کامل کنید که اگر نمره بین ۵۰ تا ۹۰ باشد، عبارت 'B' و اگر نمره کمتر از ۵۰ باشد عبارت 'C' در خروجی نمایش داده شود.

تمرین ۲۰

(آشنایی با if, else, elif)

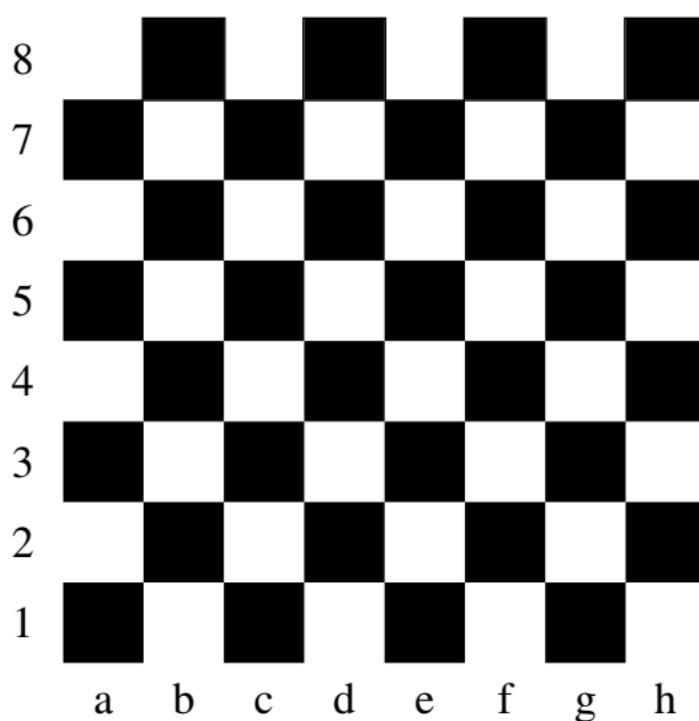
دو متغیر با نام‌های day و month داریم که روز و ماه در آن‌ها ذخیره شده است. کدی بنویسید که از این دو متغیر استفاده کند و نام فصل را در خروجی نمایش دهد. از جدول زیر برای مشخص کردن نام فصل استفاده کنید.

| Season | First day |
|--------|--------------|
| Spring | March 20 |
| Summer | June 21 |
| Fall | September 22 |
| Winter | December 21 |

تمرین ۲۱

(آشنایی با if تو در تو)

موقعیت مکانی در شطرنج با استفاده از یک حرف و یک عدد مشخص می‌شود. عدد، نشان دهنده سطر و حرف، نشان دهنده ستون است:



برنامه‌ای بنویسید که موقعیت مکانی را از کاربر دریافت کرده و مشخص کند که رنگ مربع وارد شده سفید است یا سیاه.

** به عنوان مثال، خروجی برای a1، باید black و برای d5 باید white باشد.

تمرین ۲۲

(آشنایی با دیکشنری و math)

فرمول زیر مشخص می‌کند که اول ژانویه کدام روز هفته است:

$$\text{day_of_the_week} = (\text{year} + \text{floor}((\text{year} - 1) / 4) - \text{floor}((\text{year} - 1) / 100) + \text{floor}((\text{year} - 1) / 400)) \% 7$$

نتیجه فرمول بالا یک عدد صحیح بین ۰ و ۶ است که صفر نشان دهنده یکشنبه، یک نشان دهنده دوشنبه و ... است.

کدی بنویسید که سال را از کاربر دریافت کرده و سپس بگوید در آن سال، اول ژانویه چه روزی است (مثلا در خروجی چاپ کند Monday).

تمرین ۲۳

(آشنایی با دستورات شرطی)

برنامه‌ای بنویسید که سال را دریافت کرده و بگوید آن سال کبیسه هست یا خیر.

اگر شماره سال بر ۴۰۰ تقسیم‌پذیر باشد آنگاه آن سال کبیسه است. در غیر اینصورت اگر بر ۱۰۰ تقسیم‌پذیر باشد آن سال کبیسه نیست و اگر بر ۴ تقسیم‌پذیر باشد آن سال کبیسه است. اگر هیچکدام از این شرایط برقرار نباشند، آن سال کبیسه نیست.

تمرین ۲۴

(آشنایی با for)

با استفاده از حلقه تو در تو، الگوی زیر را ایجاد کنید:

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

تمرین ۲۵

(آشنایی با while)

کدی بنویسید که یک عدد از کاربر دریافت کرده و سری فیبوناچی را تا آن عدد نمایش دهد.

** مثلاً برای ورودی ۱۰، خروجی باید ۰, ۱, ۱, ۲, ۳, ۵, ۸ باشد.

** سری فیبوناچی از معادله زیر محاسبه می‌شود:

$$F_0 = 0, F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2}$$

تمرین ۲۶

(آشنایی با for و یافتن رابطه ریاضی)

کدی بنویسید که یک عدد (n) از کاربر دریافت کرده و عبارت زیر را محاسبه کند:

$$y = -1 + 2 - 3 + \dots \pm n$$

تمرین ۲۷

(آشنایی با for تو در تو)

برنامه‌ای بنویسید که خروجی زیر را تولید کند:

```

۱
۲ ۴
۳ ۶ ۹
۴ ۸ ۱۲ ۱۶
۵ ۱۰ ۱۵ ۲۰ ۲۵
... ..
۱۰ ۲۰ ۳۰ ۴۰ ۵۰ ... ۱۰۰

```

تمرین ۲۸

(آشنایی با for, if, append)

کدی بنویسید که دو عدد را خوانده و مقسوم علیه مشترکشان را نمایش دهد.

تمرین ۲۹

(یافتن رابطه ریاضی)

عدد π با استفاده از سری زیر تعریف می‌شود:

$$\pi \approx 3 + \frac{4}{2 \times 3 \times 4} - \frac{4}{4 \times 5 \times 6} + \frac{4}{6 \times 7 \times 8} - \frac{4}{8 \times 9 \times 10} + \frac{4}{10 \times 11 \times 12} - \dots$$

برنامه‌ای بنویسید که ۱۵ تخمین از عدد π را محاسبه کند (در سری بالا، ۱۵ بار عبارت کسری تکرار شود).

تمرین ۳۰

(آشنایی با حلقه‌های تو در تو)

با استفاده از حلقه، یک جدول ضرب بنویسید. سعی کنید خروجی را به شکل زیر در بیاورید:

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

تمرین ۳۱

(آشنایی با تابع)

تابعی بنویسید که یک عدد باینری را دریافت و آن را به دسیمال تبدیل کند.

تمرین ۳۲

(آشنایی با while)

کدی بنویسید که از کاربر چندین عدد دریافت کرده و میانگین آن‌ها را محاسبه کند. تعداد اعدادی که کاربر وارد می‌کند باید متغیر باشد. زمانی که اعداد کاربر تمام شد، باید عدد صفر را وارد کند تا عملیات دریافت ورودی، متوقف شود.

** برای ورودی‌های ۱، ۵، ۹ و ۰ خروجی برابر است با: ۵

تمرین ۳۳

(یافتن رابطه برای اندیس‌دهی)

ژولیوس سزار برای اینکه نامه‌هایی که برای ژنرال‌هایش می‌نوشت لو نرود، یک نوع رمزگذاری را اختراع کرد که امروزه به کد سزار معروف است! طبق این کدگذاری، هر حرف به اندازه سه واحد شیفت پیدا می‌کند. یعنی حرف A به D، حرف B به E و ... تبدیل می‌شود. برای سه حرف آخر الفبا، باید برگردید به اول. یعنی X تبدیل به A، Y تبدیل به B و Z تبدیل به C می‌شود. کدی بنویسید که برای یک متن کدگذاری سزار انجام دهد.

تمرین ۳۴

(استفاده از while)

کدی بنویسید که به روش نیوتن، ریشه دوم یک عدد را محاسبه کند. الگوریتم نیوتن به شکل زیر است:

Read x from the user

Initialize $guess$ to $x/2$

While $guess$ is not good enough **do**

Update $guess$ to be the average of $guess$ and $x/guess$

برنامه زمانی متوقف شود که اختلاف توان دوم $guess$ و x کمتر از ۱۰ به توان ۱۲- باشد.

** برای شکستن حلقه از break استفاده کنید.

تمرین ۳۵

(آشنایی با تابع و دستورات if, for)

Run-length encoding یک تکنیک ساده برای فشرده‌سازی داده است. این تکنیک زمانی مفید است که تعداد زیادی داده تکراری به صورت پشت سرهم قرار می‌گیرند. در این نوع encoding عنصر تکراری تنها یک بار نوشته شده و در ادامه تعداد تکرار آورده می‌شود. مثلاً اجرای Run-length encoding بر روی لیست زیر :

["A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "B", "B", "B", "B", "A", "A", "A", "A", "A", "A", "B"]

نتیجه زیر را در بر خواهد داشت:

["A", 12, "B", 4, "A", 6, "B", 1]

تابعی بنویسید که یک لیست دریافت کرده و Run-length encoding را برای آن انجام دهد.

تمرین ۳۶

(آشنایی با تابع)

یک معیار برای سنجش شباهت دو رشته، edit distance است. edit distance یا فاصله ویرایش، به این معنی است که یک رشته با چند تغییر به رشته دوم تبدیل خواهد شد. مثلاً دو کلمه kitten و sitting را در نظر بگیرید. با اعمال تغییرات زیر کلمه kitten به sitting تبدیل خواهد شد:

- k به s تبدیل شود.
- e به i تبدیل شود.
- حرف g به رشته اول اضافه شود.

به این ترتیب فاصله ویرایش kitten و sitting برابر با ۳ است. تابعی بنویسید که دو رشته دریافت کرده و edit distance را برای آن‌ها محاسبه کند. برای محاسبه edit distance از الگوریتم زیر استفاده کنید:

```

Let  $s$  and  $t$  be the strings
If the length of  $s$  is 0 then
    Return the length of  $t$ 
Else if the length of  $t$  is 0 then
    Return the length of  $s$ 
Else
    Set  $cost$  to 0
    If the last character in  $s$  does not equal the last character in  $t$  then
        Set  $cost$  to 1
    Set  $d1$  equal to the edit distance between all characters except the last one
        in  $s$ , and all characters in  $t$ , plus 1
    Set  $d2$  equal to the edit distance between all characters in  $s$ , and all
        characters except the last one in  $t$ , plus 1
    Set  $d3$  equal to the edit distance between all characters except the last one
        in  $s$ , and all characters except the last one in  $t$ , plus  $cost$ 
    Return the minimum of  $d1$ ,  $d2$  and  $d3$ 

```

تمرین ۳۷

(آشنایی با تابع و تعریف flag مناسب)

تابعی بنویسید که یک عدد از کاربر دریافت کرده و بگوید که آن عدد، اول هست یا خیر.

تمرین ۳۸

(آشنایی با تابع و یافتن روابط ریاضی)

مبلغ کرایه تاکسی برای هر مسافر عبارت است از ۴ دلار هزینه پایه و ۰٫۲۵ دلار به ازای هر ۱۴۰ متر. تابعی بنویسید که مسافت طی شده را به کیلومتر دریافت کرده و کرایه تاکسی را محاسبه کند.

تمرین ۳۹

(آشنایی با تابع و دستورات شرطی)

تابعی بنویسید که چک می‌کند یک پسورد قوی هست یا خیر! یک پسورد قوی است اگر:

- حداقل ۸ کاراکتر داشته باشد
- حداقل یک حرف بزرگ داشته باشد
- حداقل یک حرف کوچک داشته باشد
- حداقل یک عدد داشته باشد

تمرین ۴۰

(آشنایی با دیکشنری)

- کدی بنویسید که همه value-ها را در دیکشنری زیر با هم جمع کند:

```
my_dict = {'data1' : 100, 'data2' : -54, 'data3' : 247}
```

- کدی بنویسید که همه value-ها را در هم ضرب کند.

تمرین ۴۱

(آشنایی با sorted)

دو کلمه آناگرام هستند، اگر کاراکترهای مشترک با ترتیب متفاوت داشته باشند مانند دو کلمه evil و live کدی بنویسید که دو رشته از کاربر دریافت کرده و چک کند که آناگرام هستند یا خیر.

تمرین ۴۲

(آشنایی با دیکشنری)

در بازی Scrabble، برای هر حرف از حروف الفبا، یک امتیاز مشخص در نظر گرفته می‌شود. سپس امتیاز هر کلمه با جمع کردن امتیاز حروف سازنده آن محاسبه می‌شود. حروف رایج، امتیاز کمتری دارند و به حروفی که کمتر رایج هستند امتیاز بیشتری اختصاص داده می‌شود. امتیاز حروف مطابق جدول زیر است:

| Points | Letters |
|--------|---------------------------------|
| 1 | A, E, I, L, N, O, R, S, T and U |
| 2 | D and G |
| 3 | B, C, M and P |
| 4 | F, H, V, W and Y |
| 5 | K |
| 8 | J and X |
| 10 | Q and Z |

کدی بنویسید که امتیاز Scrabble برای یک کلمه را محاسبه می‌کند.

تمرین ۴۳

(آشنایی با کلید و ولیو در دیکشنری)

کدی بنویسید که دو دیکشنری از کاربر دریافت کرده و مقادیر دیکشنری که کلیدهای آنها یکسان است را با هم جمع کند. مقدار جمع شده و کلید متناظر باید در یک دیکشنری جدید ذخیره شوند.

تمرین ۴۴

(آشنایی با دیکشنری)

کدی بنویسید که سه مقدار بزرگتر در یک دیکشنری را به همراه کلیدهایش نمایش دهد.

تمرین ۴۵

(آشنایی با دیکشنری و for)

دو لیست زیر را در نظر بگیرید:

```
list1 = ['apple', 'banana', 'orange']
```

```
list2 = [20, 50, 40]
```

با استفاده از این دو لیست، یک دیکشنری بسازید که key-ها از list1 و value-ها از list2 باشد.

تمرین ۴۶

(آشنایی با کلاس و متد)

یک کلاس پایتونی بنویسید که دو متد `get_string` و `print_string` دارد. `get_string` یک رشته از کاربر دریافت کرده و `print_string` رشته را برعکس کرده و نمایش می‌دهد.

تمرین ۴۷

(آشنایی با کلاس و متد)

یک کلاس پایتونی به نام rectangle بسازید. این کلاس باید یک متد برای محاسبه محیط و یک متد برای محاسبه مساحت داشته باشد.

تمرین ۴۸

(آشنایی با اندیس‌ها و دستورات for, if)

ماتریس زیر را در نظر بگیرید:

| | | |
|---|----|----|
| 0 | 10 | 3 |
| 9 | 0 | 7 |
| 6 | 5 | 11 |

این ماتریس در پایتون به شکل زیر تعریف شده است:

```
my_mat = [[0, 10, 3], [9, 0, 7], [6, 5, 11]]
```

- کدی بنویسید که بدون استفاده از توابع آماده مانند max و sort، ماکزیمم را در هر ستون از این ماتریس محاسبه کند.
- کدی بنویسید که مقادیر هر ستون را از کوچک به بزرگ مرتب کند.

تمرین ۴۹

(آشنایی با for, if, break, join, lower)

Pig Latin زبانی است که از تغییر کلمات انگلیسی ساخته شده است. ریشه این زبان ناشناخته است، اما حداقل در دو سند از قرن نوزدهم به این زبان اشاره شده است، که نشان می‌دهد قدمتش بیش از ۱۰۰ سال است.

برای ترجمه انگلیسی به Pig Latin از قوانین زیر استفاده می‌شود:

- اگر کلمه با یک حرف صامت (شامل y) آغاز شود، تمام حروف ابتدای کلمه تا اولین حرف صدادار (به استثنای y) حذف شده و به انتهای کلمه اضافه می‌شوند. سپس ay نیز به آخر کلمه اضافه می‌شود. به عنوان مثال، computer به computercay و think به inkthay تبدیل می‌شود.

- اگر کلمه با یک حرف صدادار شروع شود (شامل y نیست)، عبارت way به انتهای کلمه اضافه می‌شود. به عنوان مثال، algorithm به algorithmway و office به officeway تبدیل می‌شود.

برنامه‌ای بنویسید که یک جمله از کاربر گرفته و آن را به Pig Latin ترجمه کرده و نتیجه را نمایش دهد.

**رشته وارد شده توسط کاربر فقط شامل حروف کوچک و فاصله است.

تمرین ۵۰

(آشنایی با دستورات شرطی)

به پروسه تبدیل یک رشته به مجموعه‌ای از زیررشته‌ها Tokenize کردن گفته می‌شود. به هرکدام از زیررشته‌ها، یک توکن گفته می‌شود. در بسیاری از مواقع کار کردن با لیستی از توکن‌ها راحت‌تر از کار کردن با رشته اصلی است.

در عبارات ریاضی، توکن‌ها عبارتند از اعداد، علائم ریاضی و پرانتز. عملگرهای ریاضی که در این تمرین استفاده می‌شوند، `*`، `/`، `^`، `-` و `+` هستند. کدی بنویسید که یک عبارت ریاضی از کاربر دریافت کرده و آن را Tokenize کند.

تمرین ۵۱

(آشنایی با دستورات شرطی و حلقه)

دو لیست زیر را در نظر بگیرید:

```
dictionary = ['erheat', 'cold', 'clod', 'reheat', 'docl']
```

```
query = ['codl', 'heater', 'abcd']
```

کدی بنویسید که محاسبه کند برای هر عنصر از query، چند آناگرام در dictionary وجود دارد.

*** خروجی کد باید به صورت [3, 2, 0] باشد.

تمرین ۵۲

گوشی‌های قدیمی، لمسی نبودند و فقط دکمه‌های شماره‌گیر داشتند. هر شماره در چنین گوشی‌هایی علاوه بر آن عدد، نماینده چند حرف نیز بودند. با چند بار زدن روی هر شماره می‌توانستیم به همه حروف دسترسی داشته باشیم. به این ترتیب می‌توانستیم یک پیامک متنی بنویسیم! با استفاده از جدول زیر، کدی بنویسید که یک رشته را دریافت کند و بگوید اگر بخواهیم آن رشته را در یک گوشی قدیمی بنویسیم چندبار باید هر کلید را فشار دهیم.

** مثلاً برای ورودی Hello, World! خروجی باید به شکل زیر باشد:

4433555555666110966677755531111

| Key Symbols | |
|-------------|-----------|
| 1 | . , ? ! : |
| 2 | A B C |
| 3 | D E F |
| 4 | G H I |
| 5 | J K L |
| 6 | M N O |
| 7 | P Q R S |
| 8 | T U V |
| 9 | W X Y Z |
| 0 | space |