

Home Assignments

Home assignment 1

Task A: Qiymətin hesablanması

Bir neçə dəftər və karandaşdan ibarət məhsulların qiymətini hesablayan proqramı yazın.

- Dəftər sayı = 12
- Dəftərin qiyməti = 0.5 man
- Karandaşın sayı = 5
- Karandaşın qiyməti = 0.2 man

```
=====
Defterin qiymeti: 0.5  azn
Karandasin qiymeti: 0.2  azn
12 defter ve 5 karandasin umumi qiymeti: 7.0  azn
>>>
```

Task B: Zaman

Dəqiqə ilə verilmiş zamanı saata çevirən proqramı yazın.

```
=====
Verilmis deqiqeni saatla ifade edin
Deqiqe-> 232
232 deqiqe = 3 saat 52 deqiqedir
>>>
```

Task C: Ədədin rəqəmləri cəmi və hasilı

Klaviaturadan dörd rəqəmli təsadüfi ədəd daxil edin və ədədin rəqəmlər cəmini və hasilini ekrana çıxarın.

```
Dordreqemli ededin reqemleri cemini ve hasilini tapmaq
Dordreqemli tam ededi daxil edin-> 4873
Ededin reqemleri: 4 8 7 3
Reqemleri cemi: 4 + 8 + 7 + 3 = 22
Reqemleri hasil: 4 * 8 * 7 * 3 = 672
>>>
```

Task D: Üçbucağın sahəsi

Üçbucağın sahəsi $= \sqrt{s(s-s_1)(s-s_2)(s-s_3)}$ s_1, s_2, s_3 üçbucağın tərəflərinin uzunluqlarıdır. Yuxarıdakı düsturdan istifadə edib, üçbucağın sahəsini tapın. s_1, s_2, s_3 həqiqi ədədlər olmalıdır, klaviatüradan daxil edilməlidir və düzgün üçbucaq tərəfləri olmalıdır.

Hint: $s = \frac{s_1+s_2+s_3}{2}$

=====

Ucbagin sahesinin hesablamaq ucun tereflerini daxil edin

Teref 1->4

Teref 2->6

Teref 3->9

Verilen ucbugagin sahesi: 9.56

>>>|

Task E: Əməliyyatlar

İstifadəçi tərəfindən a və b - $[a,b]$ aralıq tam ədədlər daxil edildikdən sonra sistem bu aralıqda təsadüfi olaraq 2 ədəd seçərək aşağıdakı əməliyyatları yerinə yetirir:

- a və b -nin cəmini
- a və b -nin fərqi
- a -nın b -yə bölünməsindən alınan tam hissəni
- a -nın b -yə bölünməsindən alınan qalıqı
- $\log_{10}(a)$ -nın nəticəsi
- a üstü qüvvət b -nin nəticəsi
- a və b hasili

Hint: random və math modullarından istifadəyə ehtiyac vardır.

Emeliyyatların yerinə yetirilməsi

İki tam ededi daxil edin [a, b] = 2,15

Secilmis tesadufi ededler: 6 5

6 + 5 = 11

6 - 5 = 1

6 * 5 = 30

6 % 5 = 1

6 // 5 = 1

6 ** 5 = 7776

log10(6) = 0.78

>>>

Home assignment 2

Task A: Sait və ya samit

Bu tapşırıqda istifadəçidən əlifba hərfini oxuyan bir proqram yaradacaqsınız. İstifadəçi *a, e, i, o* və ya *u* daxil edərsə, proqramınız daxil edilmiş hərfin sait olduğunu göstərən bir mesaj göstərməlidir. İstifadəçi *y-yə* daxil edərsə, proqramınızda bəzən *y*-nin sait, bəzən *y*-nin samit olduğunu göstərən bir mesaj olmalıdır. Əks təqdirdə proqramınız hərfin samit olduğunu göstərən bir mesaj göstərməlidir.

```
Enter a letter of the alphabet: h
Entered alphabet is a consonant!
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter a letter of the alphabet: y
Sometimes it is a vowel, and sometimes it is a consonant!
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter a letter of the alphabet: e
Entered alphabet is a vowel!
```

Task B: Mövsüm

İl dörd fəsilə bölünür: yaz, yay, payız və qış. Təqvimin tərtibinə görə fəsillərin dəyişdiyi dəqiq tarixlər ildən-ilə bir az dəyişsə də, bu tapşırıq üçün aşağıdakı tarixlərdən istifadə edəcəyik:

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

İstifadəçi ayı və günü daxil etdikdən sonra onun bürcünü təyin edən proqram yazın.

```
Enter name of the month [ex. June]: March
Enter the day [ex. 5]: 20
March 20 is in Spring
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter name of the month [ex. June]: June
Enter the day [ex. 5]: 25
June 25 is in Summer
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter name of the month [ex. June]: November
Enter the day [ex. 5]: 5
November 5 is in Fall
```

Task C: Görünən işıq dalğa uzunluqları

Görünən işıq dalğa uzunluğu 380 ilə 750 nanometr (nm) arasındadır. Spektir davamlı olsa da, aşağıda göstəriləyi kimi tez-tez 6 rəngə bölünür.

Color	Wavelength (nm)
Violet	380 to less than 450
Blue	450 to less than 495
Green	495 to less than 570
Yellow	570 to less than 590
Orange	590 to less than 620
Red	620 to 750

İstifadəçinin tərəfindən dalğa uzunluğunu daxil edildikdən sonra onun rəngini bildirən bir program yazın. İstifadəçi tərəfindən daxil edilmiş dalğa uzunluğu görünən spektrin xaricində olduqda uyğun bir səhv mesajı göstərin.

```
Enter the wavelength in nm->-560
Invalid input!
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#2 HW - If funksiyasi\HW2
solution to be updated.py
Enter the wavelength in nm->710
The relevant color is Red
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#2 HW - If funksiyasi\HW2
solution to be updated.py
Enter the wavelength in nm->540
The relevant color is Green
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#2 HW - If funksiyasi\HW2
solution to be updated.py
Enter the wavelength in nm->446
The relevant color is Violet
```

Task D: Çin Bürcü

Çin bürcü heyvanları 12 ildən bir təkrarlanan illərə təyin edilir. 12 illik bir dövr aşağıdakı cədvəldə göstərilmişdir.

Year	Animal
2000	Dragon
2001	Snake
2002	Horse
2003	Sheep
2004	Monkey
2005	Rooster
2006	Dog
2007	Pig
2008	Rat
2009	Ox
2010	Tiger
2011	Hare

İstifadəçi tərəfindən daxil edilən ilə uyğun olan heyvanı göstərən program yazın. Nəzərə almaq lazımdır ki, il mənfi ola bilməz.

```
Enter the year [ex. 2021]: -36
Invalid year!
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter the year [ex. 2021]: 2010
2010 is the year of the Tiger - Peleng
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter the year [ex. 2021]: 2018
2018 is the year of the Dog - It
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter the year [ex. 2021]: 2021
2021 is the year of the Ox - Okuz
```

Task E: Bürcün təyin edilməsi

Qəzetlərdə tez-tez yayımlanan ulduz falı kiminsə anadan olma anını günəşin vəziyyətindən istifadə etməklə gələcəyi proqnozlaşdırır. Bu astrologiya sistemi, aşağıdakı cədvəldə göstəriləndi kimi ili 12 bürcə bölür:

Zodiac sign	Date range
Capricorn	December 22 to January 19
Aquarius	January 20 to February 18
Pisces	February 19 to March 20
Aries	March 21 to April 19
Taurus	April 20 to May 20
Gemini	May 21 to June 20
Cancer	June 21 to July 22
Leo	July 23 to August 22
Virgo	August 23 to September 22
Libra	September 23 to October 22
Scorpio	October 23 to November 21
Sagittarius	November 22 to December 21

İstifadəçi anadan olduğu ayı və günü daxil etdikdən sonra onun bürcünü təyin edən proqram yazın.

```
Determine your zodiac sign
Enter month [ex. March]: June
Enter the day [ex. 12]: 1
Your zodiac sign is Gemini - Ekizler!
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Determine your zodiac sign
Enter month [ex. March]: May
Enter the day [ex. 12]: 10
Your zodiac sign is Taurus - Bugal!
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Determine your zodiac sign
Enter month [ex. March]: October
Enter the day [ex. 12]: 35
Either a month or a day is invalid!
```

Home assignment 3

Task A: Temperatur

Verilən temperatur cədvəlini dərəcə Selsidən Faraheytə aşağıda verilən düstura əsasən çevirən bir proqram yazın.

$$T_F = T_C * \frac{9}{5} + 32$$

Cədvəlin cərgələri 0(sıfır) dərəcədən 10 dərəcə artımla 100 dərəcə Selsiyə qədər dəyişir. Cədvələ uyğun sütun başlıqlarını əlavə edin.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#3 HW - Dovrler\HW3 solution.py
```

Celsius	Fahrenheit
0	32
10	50
20	68
30	86
40	104
50	122
60	140
70	158
80	176
90	194
100	212

```
>>>
```

Task B: Giriş qiyməti

Zooparka giriş qiyməti gələn qonaqların yaşına görə dəyişir. 2 və 2-dən aşağı yaşda olanlar üçün pulsuz, 3 və 12 yaş arası uşaqlar üçün 14 dollar, 65 və 65 - dən yuxarı yaşda olanlar üçün 18 dollar təyin edilmişdir. Başqa qruplar üçün qiymət 23 dollar olaraq qalır.

Qonaqların yaşlarına görə zooparka ümumi giriş qiymətini hesablayan bir proqram yazın. Belə ki, istifadəçi qonaqların yaşlarını boşluq daxil edilənədək daxil edir və nəticə etibarilə ümumi qiymət hesablanıb nöqtədən sonra 2 rəqəm saxlamaqla ekranda göstərilir.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#3 HW - Dovrler\HW3 solution.py
Enter the age of guest (blank to quit): 2
Enter the age of guest (blank to quit): 5
Enter the age of guest (blank to quit): 24
Enter the age of guest (blank to quit): 7
Enter the age of guest (blank to quit): 66
Enter the age of guest (blank to quit): 45
Enter the age of guest (blank to quit):
The total for that group is $92.00
```

Task C: Orta qiymət

Aşağıda verilmiş cədvələ istinadən boşluq daxil edilənə qədər daxil edilmiş hərfi qiymətləri ədədləri çevirib orta qiyməti (balı) hesablayan bir proqram yazın. Boşluq bütün qiymətlərin daxil edilib bitdiyini göstərir. Qəbul edək ki, istifadəçi bütün hərfi qiymətləri düzgün daxil edir.

Letter	Grade points
A+	4.0
A	4.0
A–	3.7
B+	3.3
B	3.0
B–	2.7
C+	2.3
C	2.0
C–	1.7
D+	1.3
D	1.0
F	0

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#3 HW - Dovrler\HW3 solution.py
Enter the grade(blank to quit): A
Enter the grade(blank to quit): C+
Enter the grade(blank to quit): B
Enter the grade(blank to quit):

A grade point average: 3.1
>>> |
```


Task D: Parity bit – Uyğunluq bit

Bu bitlər məlumatın etibarsız xətt vasitəsilə ötürülməsi zamanı səhvlərin aradan qaldırılmasını təyin etmək üçün istifadə olunan kiçik bir mexanizmdir. Əsas mahiyyəti ondan ibarətdir ki, bu əlavə bit 8 bitlik məlumatdan sonra göndərilir və bir bitdə olan səhvi aşkar etmək asanlaşır.

Bu bitlər tək və cüt olmaqla iki cür hesablanı bilər. Cüt uyğunluq bitini təyin edilibsə, uyğunluq bitini elə seçilir ki, ümumi 1-lərin sayı (8 bitlik məlumat uyğunluq bitini daxil olmaqla) cüt olsun. Digər halda, tək uyğunluq bitini təyin edilibsə, uyğunluq bitini elə seçilir ki, ümumi 1-lərin sayı (8 bitlik məlumat uyğunluq bitini daxil olmaqla) tək olsun.

Cüv uyğunluq bitinin əvvəlcədən təyin olunduğunu nəzərə alaraq göndərilən uyğunluq bitini aşkarlayan bir proqram yazın. İstifadəçi 0 və 1 lərdən ibarət olan 8 bitlik bir məlumatı boşluq daxil edilənə qədər daxil edir və ekrana cüt və ya tək uyğunluq bitini göstərən bir məlumat çap edilir. Nəzərə alın ki, məlumat 8 bitdən artıq olarsa, ekrana səhv daxil edilmiş bir yazı çıxır.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#3 HW - Dovrler\HW3 solution.py
Enter 8 bits: 1101
That was not 8 bits ... Try again
Enter 8 bits: 1111001010101011000
That was not 8 bits ... Try again
Enter 8 bits: 10111000
The parity bit should be 0
Enter 8 bits: 10101011
The parity bit should be 1
Enter 8 bits:
>>> |
```

Task E: Çoxbucaqlının perimetri

Çoxbucaqlının perimetrini hesablayan bir proqram yazın. İstifadəçi tərəfindən çoxbucaqlının perimetrinin birinci nöqtəsinin x və y koordinatları daxil edilir. Sonra boşluq daxil edilənədək növbəti x və y koordinatları daxil edilir. Hər dəfə yeni koordinat daxil edilərkən əvvəlki nöqtə ilə məsafə hesablanıb perimetrin üzərinə əlavə olunur. Boşluq daxil edildikdən sonra proqram sonuncu daxil edilən nöqtənin koordinatı ilə başlanğıc nöqtənin koordinatına əsasən məsafəni hesablayıb perimetrin üzərinə əlavə edir. Sonda perimetrin ümumi uzunluğu göstərilir.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#3 HW - Dovrler\HW3 solution.py
Enter the x part of the coordinate: 1
Enter the y part of the coordinate: 1
Enter the x part of the coordinate (blank to quit): 2
Enter the y part of the coordinate: 1
Enter the x part of the coordinate (blank to quit): 3
Enter the y part of the coordinate: 4
Enter the x part of the coordinate (blank to quit):
The perimeter of that polygon is 7.77
>>> |
```


Home assignment 4

Task A: Collatz ardıcılığı

Collatz ardıcılığında maksimum ədədi tapmaq üçün istifadəçi bir ədəd daxil edir. Bu ədəd cüt olduğu halda 2-yə bölünür, tək olduğu halda isə 3-ə vurularaq üzərinə 1 əlavə olunur. Bu ardıcılıq 1 alınadək davam edir.

Qeyd: Collatz ardıcılığı istənilən natural ədəd üçün sonda həmişə 1 qiymətini göstərəcəkdir.

```
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
The Collatz Sequence
Enter a number (N>0): 10
10 5 16 8 4 2 1 : max number-> 16
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
The Collatz Sequence
Enter a number (N>0): 32
32 16 8 4 2 1 : max number-> 32
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
The Collatz Sequence
Enter a number (N>0): 85
85 256 128 64 32 16 8 4 2 1 : max number-> 256
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
The Collatz Sequence
Enter a number (N>0): 15
15 46 23 70 35 106 53 160 80 40 20 10 5 16 8 4 2 1 : max number-> 160
>>> |
```

Task B: Çox bölünən ədədlər

Riyaziyyatda çox bölünən ədədlər (magic numbers) müəyyən say sistemində rəqəmləri abcd... olan və rəqəmləri aşağıdakı xüsusiyyətlərə malik olan bir ədədi göstərir:

- Birinci rəqəm a ədədi 1-ə bölünür.
- İlk iki rəqəminin əmələ gətirdiyi ab ədədi 2-yə bölünür.
- İlk üç rəqəminin əmələ gətirdiyi abc ədədi 3-ə bölünür.
- İlk dörd rəqəmi ilə meydana gələn abcd ədədi 4-ə bölünür. və s.

Bir tam ədədi götürüb çox bölünən ədəd olduqda **True**, əks halda **False** qaytaran bir funksiya yaradın.

```
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Polydivisible number
Enter a number (N>0): 13
1 % 1 = 0
13 % 2 = 1
False
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Polydivisible number
Enter a number (N>0): 122
1 % 1 = 0
12 % 2 = 0
122 % 3 = 2
False
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Polydivisible number
Enter a number (N>0): 162
1 % 1 = 0
16 % 2 = 0
162 % 3 = 0
True
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Polydivisible number
Enter a number (N>0): 1624
1 % 1 = 0
16 % 2 = 0
162 % 3 = 0
1624 % 4 = 0
True
```

Task C: Bax və nömrələri söylə

İstifadəçi tərəfindən ədəd daxil edilir və aşağıdakı şərtlər ödənilmək şərtlə nəticə çap olunur.

Ədədi iki rəqəm olmaqla qruplara bölün. Əgər ədəd təkrəqəmli olarsa, **invalid** sözü çap olunsun.

Hər bir iki rəqəmli qruplar üçün sonuncu rəqəmi birinci rəqəmin sayı qədər artırıb yan-yanı düzdükdən sonra ədədi tam olaraq çap edin.

```
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#4 HW -/HW4 solution.py
Look and Say Numbers
Enter a number (N>0): 145
invalid
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#4 HW -/HW4 solution.py
Look and Say Numbers
Enter a number (N>0): 45896
invalid
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#4 HW -/HW4 solution.py
Look and Say Numbers
Enter a number (N>0): 141527
4577
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#4 HW -/HW4 solution.py
Look and Say Numbers
Enter a number (N>0): 32291534
222995444
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#4 HW -/HW4 solution.py
Look and Say Numbers
Enter a number (N>0): 2324353629
334455566699
>>> |
```

Task D: Onluq və ikilik palindrom

İstifadəçi tərəfindən ədəd daxil olunur. Funksiya və proseduralardan istifadə etməklə ədədin nəticə olaraq ya ikilik, ya onluq, ikilik və onluq, nə ikilik nə də onluq palindrom olub-olmaması ekranda göstərilir.

```
Decimal and Binary Polindroms
```

```
Enter a number (N>0): 46064
```

```
Decimal -> 46064
```

```
Binary -> 1011001111110000
```

```
Polindron type is only Decimal.
```

```
>>>
```

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW - Funksiya, prosedura\HW4 solution.py
```

```
Decimal and Binary Polindroms
```

```
Enter a number (N>0): 313
```

```
Decimal -> 313
```

```
Binary -> 100111001
```

```
Polindron type is Decimal and Binary.
```

```
>>>
```

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW - Funksiya, prosedura\HW4 solution.py
```

```
Decimal and Binary Polindroms
```

```
Enter a number (N>0): 460641
```

```
Decimal -> 460641
```

```
Binary -> 1110000011101100001
```

```
Polindron type is neither Decimal nor Binary.
```

```
>>>
```

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW - Funksiya, prosedura\HW4 solution.py
```

```
Decimal and Binary Polindroms
```

```
Enter a number (N>0): 427787
```

```
Decimal -> 427787
```

```
Binary -> 1101000011100001011
```

```
Polindron type is only Binary.
```

Task E: Smith ədədləri

Smith ədədlərini təyin etmək üçün istifadəçi müsbət bir ədəd daxil edir. Bu ədədin rəqəmsal kökü həmin ədədin sadə bölənlərinin cəminin rəqəmsal kökünə bərabər olduğu halda ədədimiz Smith ədədi adlanır.

Qeyd: İstənilən sadə ədəd Smith ədədi sayılır.

Hint: Search for Digital Root over the Net.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Smith number
Enter a number (N>2): 7
```

```
Sum of digital roots of 7 -> 7
```

```
Prime factors of 7 -> 7
```

```
Sum of prime factors of 7 -> 7
```

```
Sum of digital roots of 7 -> 7
```

```
The number - 7 is a Smith number
```

```
>>>
```

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Smith number
Enter a number (N>2): 22
```

```
Sum of digital roots of 22 -> 4
```

```
Prime factors of 22 -> 2 11
```

```
Sum of prime factors of 22 -> 13
```

```
Sum of digital roots of 13 -> 4
```

```
The number - 22 is a Smith number
```

```
>>>
```

```
>>>
```

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Smith number
Enter a number (N>2): 16
```

```
Sum of digital roots of 16 -> 7
```

```
Prime factors of 16 -> 2 2 2 2
```

```
Sum of prime factors of 16 -> 8
```

```
Sum of digital roots of 8 -> 8
```

```
The number - 16 is not a Smith number
```

```
>>>
```

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#4 HW -\HW4 solution.py
Smith number
Enter a number (N>2): 728
```

```
Sum of digital roots of 728 -> 8
```

```
Prime factors of 728 -> 2 2 2 7 13
```

```
Sum of prime factors of 728 -> 26
```

```
Sum of digital roots of 26 -> 8
```

```
The number - 728 is a Smith number
```

Home assignment 5 (əvvəlki tapşırıqları funksiyalarla yazmaq)

Home assignment 6

Task A: Kənar qiymətlərin çıxarılması

Elmi araşdırmanın bir hissəsi kimi əldə olunan məlumatı analiz edərkən onun üzərində hər hansı bir əməliyyat aparmamışdan öncə ən yuxarı qiymətlərinin çıxarılması arzuolunandır. Qiymətlər listini və mənfi olmayan bir ədədi, n , qəbul edən bir funksiya yazın. Bu funksiya listedən çıxarılması nəzərdə tutulan n sayda ən böyük və ən kiçik qiymətlərindən ibarət yeni surət çıxarmalıdır. Bu yeni surəti çıxarılmış list funksiyasının qaytaracağı list olacaqdır.

Bu funksiyanı göstərən bir proqram yazın. Sizin funksiya istifadəçi tərəfindən qiymətlərin siyahısını oxumalı və n sayda ən böyük və n sayda ən kiçik qiymətləri oradan çıxarmalıdır. Pəncərədə çıxarılan qiymətlərin siyahısı ilə birgə orijinal olan siyahı göstərməlidir. Proqram siyahıya çıxarılan ümumi qiymətlərin sayından az qiymət daxil etdiyi halda ekranda müvafiq məlumat göstərsin.

Hint: Listlə bağlı uyğun funksiyalardan istifadə edə bilərsiniz: pop, sorted və sair.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.  
py  
Enter a value (blank to quit): 3  
Enter a value (blank to quit): 4  
Enter a value (blank to quit): 6  
Enter a value (blank to quit):  
Enter the number of outliers to be removed: 2  
You didn't enter enough values.  
>>>  
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.  
py  
Enter a value (blank to quit): 3  
Enter a value (blank to quit): 4  
Enter a value (blank to quit): 6  
Enter a value (blank to quit): -6  
Enter a value (blank to quit): -12  
Enter a value (blank to quit): -4  
Enter a value (blank to quit): 23  
Enter a value (blank to quit): 45  
Enter a value (blank to quit): 76  
Enter a value (blank to quit): 47  
Enter a value (blank to quit): 2  
Enter a value (blank to quit):  
Enter the number of outliers to be removed: 2  
With the outliers removed: [-4, 2, 3, 4, 6, 23, 45]  
The original data: [3, 4, 6, -6, -12, -4, 23, 45, 76, 47, 2]
```


Task B: Dioqanalların müqayisəsi

Dioqanalların müqayisəsi üçün istifadəçi $[1,10]$ aralığından təsadüfi seçilmiş ədədlərlə $N*N$ ölçüdə listin içində listi doldurur. Daha sonra sol və sağ dioqanalları ayırır yeni listlərə əlavə edir. Sol və sağ dioqanalların ədədlərinin cəminin müqayisəsinə (böyük, kiçik və bərabər) əsasən uyğun məlumatı ekranda göstərilir.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution
.py
Random data
5      7
3      1
Left diagonal-> [5, 1] -> 6
Right diagonal-> [7, 3] -> 10
Sum of left diagonal is less than sum of right diagonal
>>>

= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution
.py
Random data
7      2      8      3
6      3      8      8
8      6      9      6
1      9      7      10
Left diagonal-> [7, 3, 9, 10] -> 29
Right diagonal-> [3, 8, 6, 1] -> 18
Sum of left diagonal is greater than sum of right diagonal
>>>

= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution
.py
Random data
10     5     6
7      10    9
5      3     1
Left diagonal-> [10, 10, 1] -> 21
Right diagonal-> [6, 10, 5] -> 21
Sum of left diagonal is equal to sum of right diagonal
```

Task C: Orta qiymətdən yuxarı və aşağı olan ədədlər

İstifadəçi tərəfindən boşluq daxil edilənə qədər tam ədədlər daxil edilir. Sizin proqramınız daxil olunan ədədlərin orta qiymətini hesablayır və proqram 1-ci olaraq orta qiymətdən aşağı qiymətləri sonra orta qiymətə bərabər olan ədədləri (əgər varsa) sonda isə orta qiymətdən böyük olan ədədləri çap edir.

```
Enter a number - [blank to quit] ->2
Enter a number - [blank to quit] ->34
Enter a number - [blank to quit] ->5
Enter a number - [blank to quit] ->15
Enter a number - [blank to quit] ->21
Enter a number - [blank to quit] ->23
Enter a number - [blank to quit] ->9
Enter a number - [blank to quit] ->16
Enter a number - [blank to quit] ->
A list of numbers:
[2, 34, 5, 15, 21, 23, 9, 16]

Average value:15.625
Less than Average: 2 5 15 9
Average:
Higher than Average: 34 21 23 16
```

Task D: List çeşidlənməsi

İstifadəçi tərəfindən boşluq daxil edilənə qədər tam ədədlər daxil edilir. Sizin proqramınız daxil olunan ədədlərin artan və ya azalan sıra ilə çeşidləndiyini və ya çeşidlənmədiyini təyin edir. Nəzərə alaq ki, listedə 1-dən artıq ədəd var.

```
Enter a number - [blank to quit] ->1
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->5
Enter a number - [blank to quit] ->7
Enter a number - [blank to quit] ->
A list is ascendingly sorted
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.py
Enter a number - [blank to quit] ->7
Enter a number - [blank to quit] ->4
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->-2
Enter a number - [blank to quit] ->
A list is decendingly sorted
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.py
Enter a number - [blank to quit] ->1
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->6
Enter a number - [blank to quit] ->2
Enter a number - [blank to quit] ->
A list is unsorted
```

Task E: Ən uyğun düz xətt

İstifadəçi 3 nöqtənin koordinatlarını (0,5) aralığından təsadüfi götürülmüş ədədlərlə x listi və y listinə daxil edir. Verilən nöqtələrə əsasən ən uyğun düz xəttin, $y = mx + b$, çəkilməsi üçün aşağıdakı düsturlardan istifadə etməklə müvafiq parametrləri hesablayın.

$$m = \frac{\sum xy - \frac{(\sum x)(\sum y)}{n}}{\sum x^2 - \frac{(\sum x)^2}{n}}$$
$$b = \bar{y} - m\bar{x}$$

Burada \bar{x} və \bar{y} uyğun olaraq x və y koordinatlarının orta qiymətləri, n isə listin uzunluğudur.

Qeyd: $\sum x$ – cəm işarəsi olub, x koordinatlarının qiymətlərinin cəmi deməkdir.

Koordinatların cəmini hesablamaq bir neçə dəfə təkrarlandığı üçün funksiya yazıb ona müraciət etmək məsləhət görülür.

```
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.py
X coordinates: [4.46, 4.39, 3.04]
Y coordinates: [2.24, 0.23, 1.47]
f(x) = -0.11*x + 1.77
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.py
X coordinates: [4.33, 0.95, 2.25]
Y coordinates: [4.86, 0.51, 2.33]
f(x) = 1.28*x - 0.65
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.py
X coordinates: [4.29, 1.08, 1.86]
Y coordinates: [4.87, 4.75, 3.34]
f(x) = 0.18*x + 3.89
>>>
= RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\HW\#6 HW - Massiv\HW6 solution.py
X coordinates: [4.15, 0.93, 3.02]
Y coordinates: [1.29, 4.17, 2.22]
f(x) = -0.90*x + 4.99
```

Əlavə tapşırıqlar

Tapşırıq 2 Funksiyalar və prosedurlar:

1. (-1) daxil ediləndə eldə olunan natural ededlerin ceminin cut ve ya tek oldugunu gosteren funksiya yazin.
2. Daxil olunan (a, b) ededleri a-nin b-ye bolunmesinden alinan qismet cut olarsa, **True** eks halda **False** qaytaran funksiya yazin.
3. 2^{N+1} ededi $2N+1$ ededine tam bolunduyu halda N ededi Curzon ededi adlanir. Curzon ededini funksiya ile teyin eden program yazin.
Curzon ededler: 1, 2, 5, 6, 9, 14, 18, 21, 26, ...
4. Giriş olaraq (n, k) daxil edilir, eger $k^k=n$ serti dogru olduqda **True**, eks halda **False** qaytaran funksiya yazin.
5. Daxil olunan p- ededi $n(n+1)$ hasiline beraber olduqda pronic, eks halda heteromecic eded adlanir. Ededin pronic olub-olmamasini mueyyenlesdiren funksiya yazin.
Pronic ededler: 6, 12, 20, 30, 42, 56, ...
6. Tam ededin uzunlugunu – reqemlerinin sayini teyin eden funksiya yazin(eded menfi ola biler).
7. N ededinin Disarium eded olub-olmadigini teyin eden funksiya yazin.
($1^1 + 7^2 + 5^3 = 1 + 49 + 125 = 175$ – Disarium ededidir.)
8. Kempner funksiyasi: en kicik musbet ededin tapilmasi ucin istifade olunur, hansiki hemin kicik ededin faktoriali daxil olunan edede bolunur.
($6 \rightarrow 3!$, $10 \rightarrow 5!$, $7 \rightarrow 7!$). Sade eded daxil edildikde ele ozu cavab olaraq cap olunur.
9. Oz reqemleri cemine tam bolunen ededin qismeti sade eded oldugu halda Moran ededi adlanir. Moran ededini teyin eden funksiya yazin. ($132 \rightarrow \text{non-Moran}$, $133 \rightarrow \text{Moran}$)
10. n – tam ededini ve p-natural ededini qebul edib k ededini cap eden funksiya yazin.
a, b, c, d (soldan saga) - n tam ededinin reqemleri
 $(a^p + b^{(p+1)} + c^{(p+2)} + d^{(p+3)} + \dots) = n * k$
Numune (n,p->k): 89,1->1, 92,1->None, 695,2->2

Tapşırıq 3 List:

3.1 Hesablama

10 elementdən ibarət massivi [1,100] aralığında yerləşən təsadüfi tam ədədlər ilə doldurun və maksimal, minimal elementi sıra nömrələri ilə birgə və orta qiymətini tapın.

Misal:

```
RESTART: C:\Users\User\Desktop\Python Foundation\Practices\#3 week\Massiv.py
Massiv: A= [44, 20, 94, 33, 36, 35, 76, 26, 98, 33]
Max number is A[8]=98
Min number is A[1]=20
Average value is 49.5
>>>
RESTART: C:\Users\User\Desktop\Python Foundation\Practices\#3 week\Massiv.py
Massiv: A= [2, 89, 62, 71, 24, 56, 73, 22, 60, 93]
Max number is A[9]=93
Min number is A[0]=2
Average value is 55.2
>>> |
```

3.2 Inversiya

Massivdə cüt sayda element var. Massivi təsadüfi elementlərlə doldurun. Massivin 1-ci və 2-ci yarısı üçün ayrı-ayrılıqda inversiya əməliyyatını yerinə yetirin

Misal:

```
RESTART: C:\Users\User\Desktop\Python Foundation\Practices\#3 week\Massiv.py
Massiv: E
[18, 36, 23, 8, 7, 45]
Half-Inverted
[23, 36, 18, 45, 7, 8]
>>>
RESTART: C:\Users\User\Desktop\Python Foundation\Practices\#3 week\Massiv.py
Massiv: E
[42, 31, 6, 40, 21, 32]
Half-Inverted
[6, 31, 42, 32, 21, 40]
>>>
RESTART: C:\Users\User\Desktop\Python Foundation\Practices\#3 week\Massiv.py
Massiv: E
[20, 45, 20, 11, 10, 13]
Half-Inverted
[20, 45, 20, 13, 10, 11]
>>>
```

3.3 Orta qiymətdən yuxarı və aşağı olan ədədlər

İstifadəçi tərəfindən boşluq daxil edilənə qədər tam ədədlər daxil edilir. Sizin proqramınız daxil olunan ədədlərin orta qiymətini hesablayır və proqram 1 ci olaraq

orta qiymətdən aşağı qiymətləri sonra orta qiymətə bərabər olan ədədləri (əgər varsa) sonda isə orta qiymətdən böyük olan ədədləri çap edir.

Misal:

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Enter a number - [blank to quit] ->2
Enter a number - [blank to quit] ->56
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->65
Enter a number - [blank to quit] ->21
Enter a number - [blank to quit] ->4
Enter a number - [blank to quit] ->32
Enter a number - [blank to quit] ->6
Enter a number - [blank to quit] ->7
Enter a number - [blank to quit] ->
List of numbers:
[2, 56, 3, 65, 21, 4, 32, 6, 7]

Average value:21.778
Less than Average:2 3 21 4 6 7
Average:
Higher than Average:56 65 32
>>> |
```

3.4 List çeşidlənmə

İstifadəçi tərəfindən boşluq daxil edilənə qədər tam ədədlər daxil edilir. Sizin proqramınız daxil olunan ədədlərin artan və ya azalan sıra ilə çeşidləndiyini və ya çeşidlənmədiyini təyin edir. Nəzərə alaq ki, listedə 1 dən artıq ədəd var.

Misal:

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Enter a number - [blank to quit] ->1
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->2
Enter a number - [blank to quit] ->4
Enter a number - [blank to quit] ->
A list is unsorted
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Enter a number - [blank to quit] ->4
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->1
Enter a number - [blank to quit] ->-3
Enter a number - [blank to quit] ->
A list is decendingly sorted
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Enter a number - [blank to quit] ->1
Enter a number - [blank to quit] ->3
Enter a number - [blank to quit] ->5
Enter a number - [blank to quit] ->7
Enter a number - [blank to quit] ->11
Enter a number - [blank to quit] ->
A list is ascendingly sorted
>>> |
```

3.5 Çeşidləmə

10 elementdən ibarət massivi [-100,100] intervalı arasında olan təsadüfi ədədlərlə doldurun və soldan sağa azalan sıra ilə düzdükdən sonra elementlərin yerlərini elə

dəyişdirin ki, massivlərin ilk yarısı soldan sağa artan sıra ilə, digər yarısı sağdan sola azalan sıra ilə yerləşsin.

Misal:

```
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Unsorted B= [-91, 52, -70, -45, 18, -99, 52, 91, 91, 20]
Sorted_1 B= [91, 91, 52, 52, 20, 18, -45, -70, -91, -99]
Sorted_2 B= [20, 52, 52, 91, 91, -99, -91, -70, -45, 18]
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Unsorted B= [13, -45, -40, 100, -36, -20, 68, 43, -82, 55]
Sorted_1 B= [100, 68, 55, 43, 13, -20, -36, -40, -45, -82]
Sorted_2 B= [13, 43, 55, 68, 100, -82, -45, -40, -36, -20]
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Massiv.py
Unsorted B= [-53, -15, 18, 38, -10, 24, -70, -3, -35, -29]
Sorted_1 B= [38, 24, 18, -3, -10, -15, -29, -35, -53, -70]
Sorted_2 B= [-10, -3, 18, 24, 38, -70, -53, -35, -29, -15]
>>> |
```

Tapşırıq 4 List in List:

4.1 Maksimal, minimal qiymet

Kvadrat matrisi [10,99] intervalında olan təsadüfi ədədlər ilə dolduran, həmin matrisdə maksimal və minimal elementlərini və onların indekslərini təyin edən proqram tərtib edin.

Misal:

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
39 62 72 24
25 39 63 10
83 59 23 55
80 78 67 48
Max value A[2][0]=83
Min value A[1][3]=10
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
58 93 57 28
16 28 73 56
48 93 35 35
60 27 92 54
Max value A[0][1]=93
Min value A[1][0]=16
>>> |
```

4.2 Piksel

Şəkildə piksellərin parlaqlığı, qiymətləri [0..255] intervalında olan ədədlərnən kodlaşdırılıb.

- 1) *Şəkildə bütün piksellərin ədədi ortasını hesablayın.*
- 2) *Parlaqlıq qiyməti ədədi ortadan kiçik olan bütün pikselləri qara (0 kodu), digər pikselləri ağ edin (255 kodu)*

Misal:

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Matris
```

```
53  35  85  70
28   0  61  26
75  90  91  81
55  58  63  69
Average brightness: 58.75
```

Result

```
0    0  255  255
0    0  255   0
255  255  255  255
0    0  255  255
```

```
>>>
```

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Matris
```

```
11  21  63  91
69  92   1   7
52  44  25  27
10  11  13  62
Average brightness: 37.44
```

Result

```
0    0  255  255
255  255   0   0
255  255   0   0
0    0   0  255
```

```
>>> |
```

4.3 Yuxari dioqanal

Kvadrat matrisi [10,99] intervalında olan təsadüfi ədədlərlə doldurun, və baş diaqonaldan yuxarıda yerləşən bütün elementlərini 0 ilə əvəzləyən proqram tərtib edin. Proqram istənilən cut ölçülü matris üçün işləməlidir.

Misal:

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Enter number of rows and columns accordingly:
-->5 5
Matris

  69   68   85   70   41
  81   92   94   68   33
  68   86   15   77   72
  11   76   22   17   90
  12   86   95   46   42

Result

   0    0    0    0    0
  81    0    0    0    0
  68   86    0    0    0
  11   76   22    0    0
  12   86   95   46    0
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Enter number of rows and columns accordingly:
-->2 2
Matris

  57   13
  18   55

Result

   0    0
  18    0
>>> |
```

4.4 Yerdeyismə

Şəklin pikselləri rəngləri bildirən ədədlərlə matris formatında təsvir olunub. Şəklin yuxardan aşağıya doğru əksini təyin edin:

Misal:

```
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Matris
  9  10  3
  3   1  3
  6   9  1

Result
  6   9  1
  3   1  3
  9  10  3
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Matris
  6  10  6  3
  7   1  7  7
  4   3  7  9
  9   2  3  6

Result
  9   2  3  6
  4   3  7  9
  7   1  7  7
  6  10  6  3
>>>
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py
Matris
  7   9  1  8  5  7
 10   8  5  7  9  2
  8   1  5  2  1  2
  9   6  1  2  7  5

Result
  9   6  1  2  7  5
  8   1  5  2  1  2
 10   8  5  7  9  2
  7   9  1  8  5  7
>>>
```

4.5 Döndərme

Şəklin pikselləri rəngləri bildirən ədədlərlə təsvir olunub. Şekli 90 dərəcəli bucaqla sağ tərəfə döndərin:

Misal:

```
...  
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py  
Matris
```

```
6 6 7  
6 8 9  
4 1 7
```

Result

```
4 6 6  
1 8 6  
7 9 7
```

```
>>>  
RESTART: C:/Users/User/Desktop/Python Foundation/Practices/#3 week/Matris.py  
Matris
```

```
9 1 10 1  
5 5 1 8  
9 5 8 7 |  
8 3 7 8
```

Result

```
8 9 5 9  
3 5 5 1  
7 8 1 10  
8 7 8 1
```

```
>>>
```

Quiz #1

Mövzu: String, Function, IF, loop

Proqramları yazarkən şərhlərdən (""" """, və ya #) mütləq şəkildə istifadə edin.

Tapşırıq 1 [10%]: Daxil edilmiş cümlədə axtarılan simvolların sayını müəyyən edin. (Heç bir hazır sətir metodu və funksiyaadan istifadə etmək olmaz!)

```
=== RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\Quiz\Quiz 2.4.py ==
Setri daxil edin: abra-abra, abra-kadabra
Axtarilan tekrarlanan soz:ab
ab simvollarin umumi sayi 4
```

Tapşırıq 2 [10%]: n tam ədədi və x həqiqi ədədi verilib. Hesablamalı:

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

```
===== RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\Quiz\Q1.py =====
Ardicilligin -- x/1!+x^2/2!+...+x^n/n! -- hesablanmasi ucun n-heddini daxil edin-> 4
x-ededini daxil edin->3
Netice->15.375
>>>
===== RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\Quiz\Q1.py =====
Ardicilligin -- x/1!+x^2/2!+...+x^n/n! -- hesablanmasi ucun n-heddini daxil edin-> 5
x-ededini daxil edin->0.8
Netice->1.225
>>>
===== RESTART: D:\TL\F\#2 Semester\Python Foundation\Practices\Quiz\Q1.py =====
Ardicilligin -- x/1!+x^2/2!+...+x^n/n! -- hesablanmasi ucun n-heddini daxil edin-> 3
x-ededini daxil edin->0.65
Netice->0.907
>>> |
```

Tapşırıq 3 [20%]: Çin Bürcü

Çin bürcü heyvanları 12 ildən bir təkrarlanan illərə təyin edir. 12 illik bir dövr aşağıdakı cədvəldə göstərilmişdir.

Year	Animal
2000	Dragon
2001	Snake
2002	Horse
2003	Sheep
2004	Monkey
2005	Rooster
2006	Dog
2007	Pig
2008	Rat
2009	Ox
2010	Tiger
2011	Hare

İstifadəçi tərəfindən daxil edilən ilə uyğun olan heyvanı göstərən proqram yazın. Nəzərə almaq lazımdır ki, il mənfi ola bilməz.

```
Enter the year [ex. 2021]: -36
Invalid year!
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter the year [ex. 2021]: 2010
2010 is the year of the Tiger - Peleng
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter the year [ex. 2021]: 2018
2018 is the year of the Dog - It
>>>
= RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/HW/#2 HW - If funksiyasi/HW2
solution.py
Enter the year [ex. 2021]: 2021
2021 is the year of the Ox - Okuz
```

Tapşırıq 4 [10%]: İstifadəçi tərəfdən daxil edilən simvollar sətrində təkrarlanan simvolların sayını tapın. (Heç bir hazır metod və funksiyadan, List və Dictionary-dən istifadə etmək olmaz!)

```
=== RESTART: D:/TL/F/#2 Semester/Python Foundation/Practices/Quiz/Quiz 2.3.py ==
Daxil edilen setir: 018450304488
Tekrarlana simvollar: 0 8 4
```

Quiz 1

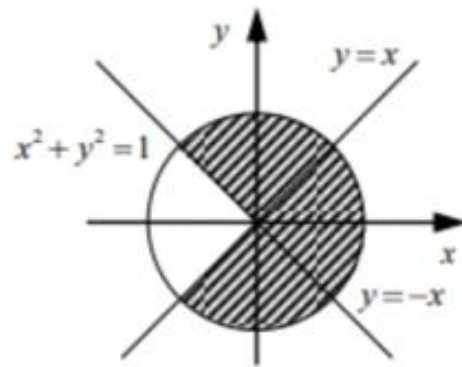
Variant A (100%)

Task 1 (25%)

İstənilən daxil edilmiş ədədin birinci və sonuncu rəqəmlərinin kvadrları cəmi 100-dən kiçik olanlarını yoxlayan məntiqi **funksiya** yazın. Funksiya True yaxud False qaytarmalıdır. Daha sonra Funksiyanı çağırıb, müxtəlif ədədlər üçün yoxlayın. (String ilə yazmaq olmaz)

Task 2 (25%)

Ştrixlənmiş oblast verilir. Klaviaturadan hər hansı bir nöqtənin koordinatları daxil edilir. Daxil edilmiş nöqtənin həmin oblasta düşüb düşmədiyini təyin edən proqramı tərtib edin:



Task 3 (25%)

Seriyanın cəmini tapın.

$$\frac{1^3}{x} + \frac{3^3}{x} + \frac{5^3}{x} + \frac{7^3}{x} \dots N \quad x \in R$$

Task 4 (25%)

İctimati giriş biletlərinin qiymətləri müxtəlif yaş qrupları üçün fərqlidir:

Age group	Entrance Fee \$
Baby	Free of charge
4-18 years	4
65 and above	6
Others	8

Mənfi ədəd girilinə qədər sistem işləməlidir. Daxil edilmiş yaşlara əsasən biletlərin qiymətlərinin cəmin tapın.

Input:	Output:
Enter the age to calculate the total fee. Neqativ number for exit: age: 4 age: 25 age: 30 age: 65 age: -1	total fee: 26 \$
Terminal example:	
<pre>Enter the age to calculate the total fee. Neqativ number for exit: age: 4 age: 25 age: 30 age: 65 age: -1 total fee: 26 \$</pre>	<pre>Enter the age to calculate the total fee. Neqativ number for exit: age: 3 age: 17 age: 18 age: 19 age: 64 age: 66 age: -1 total fee: 30 \$</pre>

Quiz 1

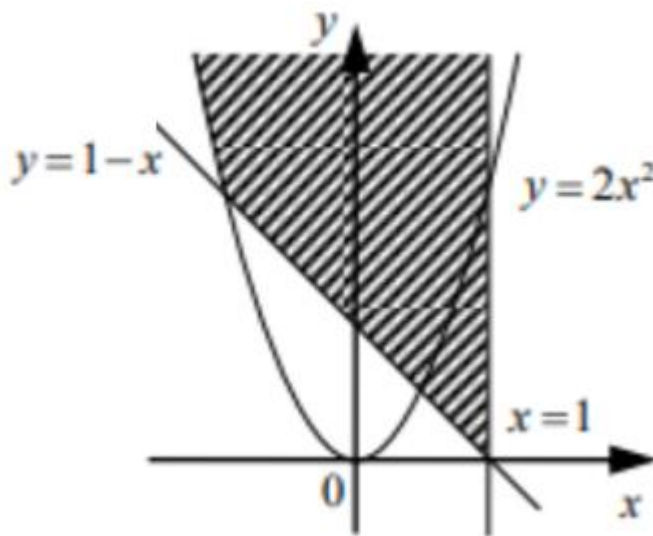
Variant B (100%)

Task 1 (25%)

İstənilən daxil edilmiş ədədin birinci və sonuncu rəqəmlərinin cəminin kökaltısının 3-də böyük olanlarını yoxlayan məntiqi **funksiya** yazın. Funksiya True yaxud False qaytarmalıdır. Daha sonra Funksiyanı çağırıb, müxtəlif ədədlər üçün yoxlayın. (String ilə yazmaq olmaz)

Task 2 (25%)

Ştrixlənmiş oblast verilir. Klaviaturadan hər hansı bir nöqtənin koordinatları daxil edilir. Daxil edilmiş nöqtənin həmin oblasta düşüb düşmədiyini təyin edən proqramı tərtib edin:



Task 3 (25%)

Seriyanın cəmini tapın.

$$x + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \dots + \frac{x^{2k+1}}{(2k+1)!} \quad x \in R$$

Task 4 (25%)

Parking qiymətləri müxtəlif saatlar üçün fərqlidir:

Hours	Entrance Fee \$
[3; 5) hours	3
[5;10) hours	5
10 and above	7
Others	Free of charge

Mənfi ədəd girilinə qədər sistem işləməlidir. Daxil edilmiş saatlara əsasən parking qiymətlərinin cəmin tapın.

Input:	Output:
Enter the age to calculate the total fee. Neqativ number for exit: hours:2 hours: 5 hours: 7 hours: 15 hours: -5	total fee: 17 \$

Quiz 2 A-quiz 2 B will be loaded