

Р. Балабан

Збірник тренувальних задач

(Мова програмування Python 2.x)



```
def my_function(param1, param2, ...):  
    pass
```

Збірник представлений у вигляді класичних задач без пояснень принципу розв'язування, це зроблено з метою спонукання того хто хоче вивчити програмування не тільки копіювати код, а й зробити алгоритмічний розв'язок задачі.

Умови задач взято із збірника «М. Э. Абрамян. Programming Taskbook. Электронный задачник по программированию».

Збірник структуровано за тим самим принципом, що і в оригінальному збірнику. Задачі розміщено за принципом від простого до складного.

Матеріал збірника відповідає чинній програмі з предмету «Інформатика»

Переважна більшість розв'язків зроблена автором, але ідея розв'язку деяких задач взято із Інтернету.

Розв'язки задач створені в середовищі Python 2.x

Зміст

1. Короткі теоретичні відомості
2. Введення-Виведення.
3. Цілі числа
4. Умовний оператор (розгалуження)
5. Оператор повторення (цикли)
6. Списки (масиви)
7. Символи. Рядки

Короткі теоретичні відомості

Команда **input()** зчитує введене значення та переходить на наступний рядок.

int – величини будуть перетворюватись у цілочисельний тип

float – величини будуть перетворюватись у дробовий тип

```
a=int(input())  
b=float(input())
```

raw_input() – зчитування рядка величин

split() – розділення величин за ознакою пропуск

int – величини будуть перетворюватись у цілочисельний тип

```
x, y = map(int, raw_input().split())  
x, y, z = map(float, raw_input().split())
```

Команда **print** виводить значення та переходить на наступний рядок

```
print x
```

Команда **print** виводить через пропуск в одному рядку два числа

```
print x, y
```

Математичні оператори

Оператор	Команда	Приклад	Результат
+	додавання	15+3	18
-	віднімання	6-4	2
*	множення	5*4	20
/	ділення	24/3	8
//	ділення націло	7//3	2
%	залишок від ділення	9 %6	3
**	піднесення до степеню	2**2	4

Порядок операцій

1. Дужки ()
2. піднесення до степеню **
3. множення *, ділення /, остача від ділення %
4. додавання +, віднімання -

ВВЕДЕННЯ-ВИВЕДЕННЯ

№1

Дано сторону квадрату a . Знайти його периметр $P=4\cdot a$.

```
a = float (input())
P = 4 * a
print P
```

№2

Дано сторону квадрата a . Знайти його площу $S=a^2$.

```
a = float (input())
S1 = a*a
S2 = a**2
print S1
print S2
```

№3

Дано сторони прямокутника a і b . Знайти його площу $S=a\cdot b$ і периметр $P=2\cdot(a+b)$.

```
a = float (input())
b = float (input())
S = a * b
P = 2*(a+b)
print S
print P
```

№4

Дано діаметр кола d . Знайти його довжину $L=\pi\cdot d$. В якості значення π використовувати 3,14.

```
d = float (input())
L = 3.14 * d
print L
```

№5

Дано довжину ребра куба a . Знайти об'єм куба $V=a^3$ і площу його поверхні $S=6\cdot a^2$.

```
a = float (input())
V = a**3
S = 6* a**2
print V
print S
```

№6

Дано довжини ребр a , b , c прямокутного паралелепіпеда. Знайти його об'єм $V=a\cdot b\cdot c$ і площу поверхні $S=2\cdot(a\cdot b+b\cdot c+a\cdot c)$.

```
a = float (input())
b = float (input())
c = float (input())
V = a*b*c
S = 2*(a*b+b*c+a*c)
print V
print S
```

№7

Знайти довжину кола L і площу круга S заданого радіуса R . $L=2\cdot\pi\cdot R$, $S=\pi\cdot R^2$. В якості значення π використовувати 3,14.

```
R = float (input())
L = 2*3.14*R
S = 3.14*R*R
print L
print S
```

№8

Дано два числа a і b . Знайти їх середнє арифметичне: $(a + b)/2$.

```
a = float (input())
b = float (input())
c = (a+b)/2
print c
```

№9

Дано два невід'ємних числа a і b . Знайти їх середнє геометричне, тобто квадратний корінь з їх добутку: $\sqrt{a\cdot b}$.

```
from math import *
a = float (input())
b = float (input())
c = sqrt (a*b)
print c
```

№10

Дано два ненульових числа. Знайти суму, різницю, добуток і частку їх квадратів.

```
a = float(input())
b = float(input())
c = a**2
d = b**2
print c+d
print c-d
print c*d
print c/d
```

№11

Дано катети прямокутного трикутника a і b . Знайти його гіпотенузу c і периметр

P : $c = \sqrt{a^2 + b^2}$, $P=a+b+c$

```
a = float(input())
b = float(input())
from math import *
c = sqrt (a**2 + b**2)
P = a + b + c
print c
print P
```

№12

Дано два кола із загальним центром і радіусами R_1 і R_2 ($R_1 > R_2$). Знайти площі цих кіл S_1 і S_2 , а також площу S_3 кільця, зовнішній радіус якого дорівнює R_1 , а внутрішній радіус дорівнює R_2 : $S_1=\pi\cdot(R_1)^2$, $S_2=\pi\cdot(R_2)^2$, $S_3=S_1 - S_2$.

```
R1 = float(input())
R2 = float(input())
S1 = 3.14*(R1**2)
S2 = 3.14*(R2**2)
S3 = S1 - S2
print S1
print S2
print S3
```

№13

Дана довжина L окружності. Знайти її радіус R і площу S круга, обмеженого цим колом, враховуючи, що $L=2\cdot\pi\cdot R$, $S=\pi\cdot R^2$. Як значення π вважати рівним 3,14.

```
L = float(input())
R = L/(2*3.14)
S = 3.14*(R**2)
print '%.2f' % R
print '%.2f' % S
```

№14

Дана площа S кола. Знайти його діаметр D і довжину L кола, що обмежує це коло, враховуючи, що $L=2\cdot\pi\cdot R$, $S=\pi\cdot R^2$. Як значення π вважати рівним 3.14.

```
S = float(input())
from math import *
R = sqrt (S/3.14)
D = 2*R
L = 2*3.14*R
print '%.2f' % D
print '%.2f' % L
```

№15

Дано три точки A , B , C на числової осі. Знайти довжини відрізків AC і BC і їх суму.

```
A = float(input())
B = float(input())
C = float(input())
from math import *
K = abs (C-A)
L = abs (C-B)
F = K+L
print K
print L
print F
```

№16

Дано координати двох протилежних вершин прямокутника: (x_1, y_1) , (x_2, y_2) . Сторони прямокутника паралельні осям координат. Знайти периметр і площу даного прямокутника.

```
x1 = float(input())
y1 = float(input())
x2 = float(input())
y2 = float(input())
from math import *
```

```

AB = abs (y2-y1)
BC = abs (x2-x1)
P = 2*(AB+BC)
S = AB*BC
print P
print S

```

№17

Знайти відстань між двома точками із заданими координатами (x_1, y_1) і (x_2, y_2) на площині. Відстань обчислюється за формулою: $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

```

x1, y1 = map (int, raw_input().split())
x2, y2 = map (int, raw_input().split())
import math
L = math.sqrt((x2-x1)**2+(y2-y1)**2)
print L

```

№18

Поміняти місцями вміст змінних A і B та вивести нові значення A і B .

```

a = input()
b = input()
a,b = b,a
print a , b

```

№19

Дано змінні A, B, C . Змінити їх значення, перемістивши вміст A в B , B - в C , C - в A , і вивести нові значення змінних A, B, C .

```

A = input()
B = input()
C = A
A = B
B = C
print A
print B

```

№20

Знайти значення функції $y = 3x^6 - 6x^2 - 7$ при заданому значенні x .

```

x = float(input())
y = 3*x**6-6*x**2-7
print y

```

№21

Знайти значення функції $y = 4 \cdot (x-3)^6 - 7 \cdot (x-3)^3 + 2$ при даному значенні x .

```

x = float(input())
y = 4*(x-3)**6-7*(x-3)**3+2
print y

```

№22

Дано число A . Обчислити A^8 , використовуючи допоміжну змінну і три операції множення. Для цього послідовно знайдіть A^2, A^4, A^8 . Вивести всі знайдені степені числа A .

```

A = float(input())

```



```
#Варіант 1
B = A*A
print B
B = B * B
print B
B = B * B
print B
C = A**8 #Варіант 2
print C
```

№23

Дано значення кута α в градусах ($0 < \alpha < 360$). Обчислити значення цього ж кута в радіанах, враховуючи, що $180^0 = \pi$ радіанів. В якості значення π використовувати 3,14.

```
alfa = input()
from math import *
deg = radians (alfa)
print round (deg, 2)
```

№24

Дано значення кута α в радіанах ($0 < \alpha < 2 \pi$). Обчислити значення цього ж кута в градусах, враховуючи, що $180^0 = \pi$ радіанів. В якості значення π використовувати 3,14.

```
deg = input()
from math import *
alfa = degrees(deg)
print '%1.f' % alfa
```

№25

Швидкість першого автомобіля V_1 км/год, другого - V_2 км/год, відстань між ними S км. Визначити відстань між ними через T годин, якщо автомобілі віддаляються один від одного. Дана відстань рівна сумі початкової відстані і загального шляху, пройденого автомобілями; загальний шлях = час · сумарна швидкість.

```
V1 = float (input())
V2 = float (input())
S = float (input())
T = float (input())
L = T * (V1+V2)
print L+S
```

ЦІЛІ ЧИСЛА

№1

Дано масу M в кілограмах. Використовуючи операцію ділення націло, знайти кількість повних тон в ній (1 тонна = 1000 кг).

```
M=int (input())
k=M//1000
print k
```

№2

Дано розмір файлу в байтах. Використовуючи операцію ділення націло, знайти

кількість повних кілобайт, які займає даний файл (1 кілобайт = 1024 байт).

```
b=int (input())
kb=b//1024
print kb
```

№3

Дано двозначне число. Знайти суму і добуток його цифр.

```
N=int (input())
a=N//10
b=N%10
S=a+b
D=a*b
print ('Sum=',S)
print ('Dob=',D)
```

№4

Дано тризначне число. У ньому закреслили першу зліва цифру і приписали її справа. Вивести отримане число.

```
N=int (input())
a=N//100
b=(N//10)%10
c=N%10
M=b*100+c*10+a*1
print M
```

№5

Дано тризначне число. Вивести число, отримане при перестановці цифр сотень і десятків вихідного числа (наприклад, 123 перейде в 213).

```
N=int (input())
a=N//100
b=(N//10)%10
c=N%10
M=b*100+a*10+c*1
print M
```

№6

Дано ціле число, більше 999. Використовуючи одну операцію ділення націло і одну операцію взяття залишку від ділення, знайти цифру, відповідну розряду сотень в записі цього числа.

```
N=int (input())
M=(N//100)%10
print M
```

№7

З початку доби минуло N секунд (N - ціле). Знайти кількість повних хвилин, що минули з початку доби.

```
N = int (input())
K = N // 60
print K
```

№8

З початку доби минуло N секунд (N - ціле). Знайти кількість повних хвилин, що минули з початку останньої години.

```
N = int (input())
K = (N%3600)//60
print K
```

№9

Дні тижня пронумеровані наступним чином: 0 - неділя, 1 - понеділок, 2 - вівторок, . . . , 6 - субота. Дано ціле число K , що лежить в діапазоні 1-365. Визначити номер дня тижня для K -го дня року, якщо відомо, що цього року 1 січня було четвергом.

```
K = int (input())
D = (K+3)%7
print D
```

№10

Дано номер деякого року (ціле позитивне число). Визначити відповідний йому номер століття, враховуючи, що, наприклад, початком 20 століття був 1901 рік.

```
N = int (input())
S = (N//100)+1
print S
```

УМОВНИЙ ОПЕРАТОР (розгалуження)

Структура умовного оператора

```
if <умова>:
    <серія1> (відступ складає 4 пробіли або 1 Tab)
else:
    <серія2>
```

№1

Дано ціле число. Якщо воно є позитивним, то додати до нього 1; в іншому разі не змінювати його. Вивести отримане число.

```
n = int(input())
if n>0:
    n=n+1
    print n
else:
    print n
```

№2

Дано ціле число. Якщо воно є позитивним, то додати до нього 1; в іншому випадку відняти від нього 2. Вивести отримане число.

```
n = int(input())
if n>0:
    n=n+1
    print n
else:
    n=n-2
    print n
```

№3

Дано ціле число. Якщо воно є позитивним, то додати до нього 1; якщо

негативним, то відняти від нього 2; якщо нульовим, то замінити його на 10.
Вивести отримане число.

```
n = int(input())
if n>0:
    n=n+1
    print n
if n<0:
    n=n-2
    print n
if n==0:
    n=10
    print n
```

№4

Дано два числа. Вивести більше з них.

```
a = float (input ())
b = float (input ())
if a>b:
    print a
else:
    print b
```

№5

Дано два числа. Вивести спочатку більше, а потім менше з них.

```
a = float (input ())
b = float (input ())
if a>b:
    print a, b
else:
    print b, a
```

№6

Дано дві змінні цілого типу: *A* і *B*. Якщо їх значення не рівні, то присвоїти кожній змінній суму цих значень, а якщо рівні, то присвоїти змінним нульові значення.
Вивести нові значення змінних *A* і *B*.

```
a = float (input ())
b = float (input ())
if a!=b:
    c=a+b
    a=c
    b=c
    print a
    print b
elif a==b:
    a=0
    b=0
    print a
    print b
```

№7

Дано три числа. Знайти найменше з них.

```
a = float (input ())
```

```

b = float (input ())
c = float (input ())
if a<b and a<c:
    print a
else:
    if b<c:
        print b
    else:
        print c

```

№8

Дано три числа. Знайти середнє з них (тобто число, розташоване між найменшим і найбільшим).

```

a = float (input ())
b = float (input ())
c = float (input ())
if (a > c) and (a > b):
    if c > b:
        print c
    else:
        print b
if (b > a) and (b > c):
    if a > c:
        print a
    else:
        print c
if (c > a) and (c > b):
    if a > b:
        print a
    else:
        print b

```

№9

Дано три числа. Знайти суму двох найбільших з них.

```

a = float (input ())
b = float (input ())
c = float (input ())
if (a < b) and (a < c):
    print b+c
else:
    if b < c:
        print a+c
    else:
        print a+b

```

№10

Для даного цілого x знайти значення наступної функції f , що приймає значення цілого типу:

$$f(x) = \begin{cases} 2 \cdot x, & \text{якщо } x < -2 \text{ або } x > 2 \\ -3 \cdot x, & \text{в інших випадках} \end{cases}$$

```

x = int (input ())
if (x < -2) or (x > 2):
    f = 2 * x

```

```
else:
    f = -3 * x
print f
```

№11

Для даного дійсного x знайти значення наступної функції f , що приймає дійсні значення:

$$f(x) = \begin{cases} -x, & x \leq 0 \\ x^2, & 0 < x < 2 \\ 4, & x \geq 2 \end{cases}$$

```
x = float (input())
if x <= 0:
    f = - x
if (x > 0) and (x < 2):
    f = x**x
if x >= 2:
    f = 4
print f
```

№12

Дано номер року (позитивне ціле число). Визначити кількість днів в цьому році, враховуючи, що звичайний рік нараховує 365 днів, а високосний - 366 днів. Високосним вважається рік, що ділиться на 4, за винятком тих років, які діляться на 100 і не діляться на 400 (наприклад, роки 300, 1300 і 1900 не є високосними, а 1200 і 2000 є високосними).

```
rik = int (input())
if rik%4 == 0:
    if rik%100 == 0:
        if rik% 400 <> 0:
            print '365'
        else:
            print '366'
    else:
        print '366'
else:
    print '365'
```

ОПЕРАТОР ПОВТОРЕННЯ (цикли)

Структура повторення (циклу)

Цикл while

```
while <умова>:
    <серія команд>
```

Цикл for

```
for <змінна> in <функція>:
    <серія команд>
```

!!! Відступи в циклах такі ж як і в розгалуженні.

№1

Дано цілі числа K і N ($N > 0$). Вивести N раз число K .

```
K = int (input())
```

```
N = int (input())
for i in range (N):
    print K,
```

№2

Дано дійсне число – ціна 1 кг цукерок. Вивести вартість 1, 2, . . . , 10 кг цукерок.

```
N = float (input())
for i in range (1,11):
    print i*N,
```

№3

Дано два цілих числа A і B ($A < B$). Знайти суму всіх цілих чисел від A до B включно.

```
A, B = map (int, raw_input().split())
S=0
for i in range (A,B+1):
    S=S+i
print S
```

№4

Дано дійсне число A і ціле число N ($N > 0$). Використовуючи один цикл, вивести всі цілі ступеня числа A від 1 до N .

```
A = float (input())
N = int (input())
P=1
for i in range (1, N+1):
    P=P*A
    print P
```

№5

Дано ціле число N ($N > 0$). Знайти добуток $N! = 1 \cdot 2 \cdot \dots \cdot N$ (N -факторіал).

```
N = int (input())
i=1
fact=1
while i<=N:
    fact=fact*i
    i=i+1
print fact
```

№6

Дано ціле число N ($N > 2$). Послідовність цілих чисел A_K визначається в спосіб:

$$A_1 = 1, \quad A_2 = 2, \quad A_3 = 3, \\ A_K = A_{K-1} + A_{K-2} - 2 \cdot A_{K-3}, \quad K = 4, 5, \dots$$

Вивести елементи A_1, A_2, \dots, A_N .

```
N = int (input())
A1=1
A2=2
A3=3
print A1
print A2
print A3
for i in range (4,N+1):
```

```
a=A3
A3=A3+A2-2*A1
A1=A2
A2=a
print A3,
```

№7

Дано цілі додатні числа N і K . Використовуючи тільки операції додавання і віднімання, знайти частку від ділення без остачі N на K , а також залишок від цього поділу.

```
N,K = map (int, raw_input().split())
dil=0
while N-K>=0:
    N=N-K
    dil=dil+1
print dil
print N
```

№8

Дано ціле число N ($N > 1$). Вивести найменше із цілих чисел K , для яких сума $1+2+\dots+K$ буде більше або дорівнює N , і саму цю суму.

```
N = int (input())
i=1
S=1
while S<N:
    i=i+1
    S=S+i
print i
print S
```

№9

Початковий внесок в банку дорівнює 1000 грн. Через кожен місяць розмір вкладу збільшується на P відсотків від наявної суми (P - дійсне число, $0 < P < 25$). За даним P визначити, через скільки місяців розмір вкладу перевищить 1100 грн., і вивести знайдену кількість місяців K (ціле число) і підсумковий розмір вкладу S (дійсне число).

```
P = float (input())
Summ=1000
K=1
while Summ<=1100:
    Summ=Summ*(P/100+1)
    K=K+1
print K
print Summ
```

№10

Дано ціле число N ($N > 0$). Використовуючи операції ділення націло і взяття залишку від ділення, вивести всі його цифри, починаючи з самої правої (розряду одиниць).

```
N = int (input())
while N>0:
```



```
print N%10,  
N=N//10
```

№11

Дано цілі додатні числа A і B . Знайти їх найбільший спільний дільник (НСД), використовуючи алгоритм Евкліда:

$$\text{НСД}(A, B) = \text{НСД}(B, A \bmod B), \text{ якщо } B = 0; \text{НСД}(A, 0) = A,$$

де «mod» позначає операцію взяття залишку від ділення.

```
A, B = map(int, raw_input().split())
```

```
while (A<>0) and (B<>0):
```

```
    if A>=B:
```

```
        A=A%B
```

```
    else:
```

```
        B=B%A
```

```
print A+B
```

№12

Дано ціле число N ($N > 1$). Послідовність чисел Фібоначчі F_K визначається наступним чином:

$$F_1 = 1, \quad F_2 = 1, \quad F_K = F_{K-2} + F_{K-1}, \quad K = 3, 4, \dots$$

Перевірити, чи є число N числом Фібоначчі. Якщо є, то вивести *TRUE*, якщо ні - вивести *FALSE*.

```
N=int(input())
```

```
f1=1
```

```
f2=1
```

```
while N>f2:
```

```
    f=f2
```

```
    f2=f1+f2
```

```
    f1=f
```

```
if n==f2:
```

```
    print 'True'
```

```
else:
```

```
    print 'False'
```

СПИСКИ (масиви)

Одномірний масив

<ім'я списку> = [<елемент 1>, <елемент 2>, ..., <елемент N>]

```
nomer = [1, 2, 3, 4, 5, 6, 7]
```

```
veselka = ['R', 'O', 'Y', 'G', 'B', 'I', 'V']
```

```
print nomer
```

```
print veselka
```

Двомірний масив

Двомірний масив – це список рядків, кожен елемент якого є в свою чергу списком.

<ім'я масиву> = [[список 1], [список 2], ..., [список N]]

```
A = [ [1, 2, 3], [4, 5, 6] ]
```

№1

Дано ціле число N ($N > 0$). Сформувати і вивести цілочисельний масив розміру N , що містить N перших позитивних непарних чисел: 1, 3, 5, ...

```
N = int(input())
a = []
for i in range(1, (N+1), 2):
    a.append(i)
print(a)
```

№2

Дано ціле число N ($N > 0$). Сформувати і вивести цілочисельний масив розміру N , що містить ступеня двійки від першої до N -ї: 2, 4, 8, 16, . . .

```
N = int(input())
a=[2**i for i in range(1,N+1)]
print a
```

№3

Дано масив розміру N . Вивести його елементи у зворотному порядку.

```
N = int(input())
a= []
for i in range(0, N+1):
    a.append(i)
print a
for i in range(len(a)-1,-1,-1):
    print a[i],
```

№4

Дано цілочисельний масив розміру N . Вивести всі непарні числа, що містяться в даному масиві в порядку зростання їх індексів, а також їх кількість K .

```
from random import*
n=int(input())
a=[randint(0,100) for i in range (n)]
print(a)
b=[]
for i in range(len(a)):
    if a[i]%2!=0:
        b.append (a[i])
print b
print len (b)
```

№5

Дано масив A розміру N (N - парне число). Вивести його елементи з парними номерами в порядку зростання номерів: $A_2, A_4, A_6, \dots, A_N$. Умовний оператор не використовувати.

```
from random import*
N = int(input())
A = [randint(0,100) for i in range(N+1)]
print A
for i in range(1, (N+1), 2):
    print A[i],
```

№6

Дано масив A ненульових цілих чисел розміром 10. Вивести значення першого з тих його елементів A_K , які задовольняють нерівності $A_K < A_{10}$. Якщо таких

елементів немає, то вивести 0.

```
from random import *
N = 10
A = [randint(0,10) for i in range(N+1)]
print A
f = True
for i in range(0,N-1):
    if A[i] < A[N-1]:
        print(i,A[i])
        f = False
        break
if f:
    print(0)
```

№7

Дано масив A розміром N . Знайти мінімальний елемент із його елементів з парними номерами: A_2, A_4, A_6, \dots

```
from random import *
N = 10
A = [randint(0,10) for i in range(N+1)]
print A
print min(A[::2])
```

№8

Дано цілі додатні числа M і N . Сформувати цілочисельну матрицю розміру $M \times N$, у якій всі елементи I -го рядка мають значення $10 \cdot I$ ($I = 1, \dots, M$).

```
from random import *
M, N = map(int, raw_input().split())
a = [[(i*10) for j in range(M)] for i in range(N)]
for i in range(len(a)):
    print a[i]
```

№9

Дана матриця розміру $M \times N$. Вивести її елементи, розташовані в стовпцях з непарними номерами (1, 3,...). Виведення елементів здійснювати по стовпцях, умовний оператор не використовувати.

```
from random import *
M, N = map(int, raw_input().split())
a = [[randint(0,100) for j in range(M)] for i in range(N)]
for i in range(len(a)):
    print a[i]
print '-----'
for i in range(M):
    print a[i][0::2]
```

№10

Дана матриця розміру $M \times N$. Для кожного рядка матриці знайти суму її елементів.

```
from random import *
M, N = map(int, raw_input().split())
a = [[randint(0,100) for j in range(M)] for i in range(N)]
for i in range(len(a)):
    print a[i]
```

```
print '-----'
for i in range(M):
    print sum (a[i])
```

№11

Дана матриця розміру $M \times N$. У кожному рядку матриці знайти мінімальний елемент.

```
from random import *
M, N = map (int, raw_input().split())
a = [[randint(0,100) for j in range(M)] for i in range(N)]
for i in range (len(a)):
    print a[i]
print '-----'
for i in range(M):
    print min (a[i]),
```

№12

Дана матриця розміру $M \times N$. У кожному стовпці матриці знайти максимальний елемент.

```
from random import *
M, N = map (int, raw_input().split())
a = [[randint(0,100) for j in range(M)] for i in range(N)]
for i in range (len(a)):
    print a[i]
print '-----'
for j in range(N):
    print max (a[j]),
```

СИМВОЛИ. РЯДКИ

Методи – це функції, що застосовуються до об'єкта. Виклик методу має наступну структуру:

ім'я об'єкту.ім'я методу(Параметри методу)

len() - дозволяє виміряти довжину рядка.

find() - знаходить у зазначеному рядку вказаний підрядок та повертає індекс першого входження шуканого підрядка у рядок

rfind() - знаходить індекс останнього входження шуканого підрядку у зазначений рядок

replace() - замінює один підрядок іншим

count() - підраховує кількість входжень одного рядка в інший

№1

Дан символ. Вивести його код (тобто номер в кодовій таблиці).

```
S = input ()
print ord (S)
```

№2

Дано рядок. Вивести рядок, що містить ті ж символи, але в зворотному порядку.

```
Варіант 1
S = input ()
print(S)
print(S[::-1])
```

```
Варіант 2
S=input()
a=list(S)
a.reverse()
c=''.join(a)
print c
```

№3

Дано непорожній рядок *S*. Вивести рядок, що містить символи рядка *S*, між якими вставлено по одному пробілу.

```
S = input ()
print ' '.join(S)
```

№4

Дано рядок. Підрахувати кількість цифр у ньому.

```
S = input ()
C = sum(i.isdigit() for i in S)
print C
```

№5

Дана рядок. Перетворити в ній все прописні латинські букви в рядкові.

```
S = input ()
en_Up = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
SN = ''
for C in S:
    if C in en_Up:
        C = C.lower()
    SN=SN+C
print SN
```

№6

Дано ціле позитивне число. Вивести символи, що зображують цифри цього числа (в порядку зліва направо).

```
N = int (input())
S = list(str(N))
print S
```

№7

Дано рядок, що зображає ціле позитивне число. Вивести суму цифр цього числа.

```
S = input ()
A = list (S)
N = list(map(int,A))
print A
print sum (N)
```

№8

Дан символ *C* і рядок *S*. Подвоїти кожне входження символу *C* в рядок *S*.

```
C = input ()
S = input ()
SN = ''
for i in S:
```

```
    if i == C:
        SN=SN+i
    SN=SN+i
print SN
```

№9

Дана рядок, що складається з слів, розділених пробілами. Знайти кількість слів у рядку.

```
S = input ()
for i in range (len(S)):
    d=S.count(' ')
print d+1
```

№10

Визначити, чи є введений рядок паліндромом, як от АВВА, kazak і т. ін.

```
s1 = input ()
a = list(s1)
a.reverse()
s2 = ''.join(a)
if s1 == s2:
    print 'PALINDROM'
else:
    print 'NOT PALINDROM'
```