Kragujevac, školska 2026/27



PRAKTIKUM IZ PROGRAMIRANJA 1

VEŽBE 5

Ana Vidosavljević, Marija Trnavac

Zadatak 1

1. Koji je rezultat izvršavanja sledećeg programa:

```
p = 1
for i in range(6):
    p = p * i
print(p)
```

Zadatak 1

1. Koji je rezultat izvršavanja sledećeg programa:

```
p = 1
for i in range(6):
    p = p * i
print(p)
```

0

Zadatak 2

2. Šta će biti rezultat izvršavanja sledećih komandi:

a) print(“18 - 7 je:”, 18 - 7)
b) print(5**3 // 4 - 8)
c) print(sqrt(25) == 5 ** 2 // 1)
d) print(min(“pip”, “pipp”) + 1)

Zadatak 2

2. Šta će biti rezultat izvršavanja sledećih komandi:

a) print(“18 - 7 je:”, 18 - 7)
b) print(5**3 // 4 - 8)
c) print(sqrt(25) == 5 ** 2 // 1)
d) print(min(“pip”, “pipp”) + 1)

a) 18 – 7 je: 11
b) 23
c) False
d) Greška, moguća je samo konkatenacija stringova, ne stringova i celobrojnih vrednosti

3. Napisati program kojim se za prirodan broj **n** izračunava suma prvih **n** sabiraka:

$$S = \sum_{i=1}^{n} (3+5+7+\ldots+2i+1) \frac{\left|(2n)!-(2i+1)!\right|}{\left(2+2^2+\ldots+2^i\right)}$$

Zadatak 3

```
n = int(input("Unesi n: "))

s1 = 0
fakti = 1
faktn = 1
s2 = 0
stepen2 = 1
sFinal = 0

for i in range(2, 2*n+1):
    faktn *= i

for i in range(1,n+1):
    s1 += 2*i+1
    stepen2 *= 2
    s2 += stepen2
    fakti *= (2*i)*(2*i+1)
    if faktn-fakti < 0:
        pom = -1
    else:
        pom = 1
    sFinal += s1*(faktn-fakti)*pom/s2

print(sFinal)
```

Zadatak 3 – rešenje 2

```
from math import factorial

n = int(input("Unesi n: "))

s1 = 0
faktn = factorial(2*n)
s2 = 0
sFinal = 0

for i in range(1,n+1):
    s1 += 2*i+1
    fakti = factorial(2*i+1)
    s2 += 2**i

    sFinal += s1*abs(faktn-fakti)/s2

print(sFinal)
```

4. Napisati program i algoritam koji za unete realne vrednosti **x** i **y** računa vrednost funkcije **f(x, y)**, (funkciju abs ne smete koristiti ni kod uslova definisanosti, ni kod računanja vrednosti), na sledeći način.

$$f(x,y)=\begin{cases}(x+2)\cdot(y+2)^2, & |x+2y|>8\\ \dfrac{x+2y}{x+2}, & |x+2y|<2\\ \max\{x+2y,x^2,|x+2y+6|\}, & inače\end{cases}$$

Zadatak 4

```
x = float(input("Unesi x: "))
y = float(input("Unesi y: "))
uslov = x + 2 * y
if(uslov < 0):
    uslov = -uslov
if(uslov > 8):
    f = (x + 2) * (y + 2)**2
    print("f(%.2f, %.2f) = %.2f" %(x, y, f))
elif(uslov < 2):
    if x!=-2:
        f = (x + 2 * y)/(x + 2)
        print("f(%.2f, %.2f) = %.2f" %(x, y, f))
    else:
        print("Ne moze se deliti nulom!")
else:
    abs_max = x + 2 * y + 6
    if(abs_max < 0):
        abs_max = -abs_max
    f = max(max(x + 2 * y, x**2), abs_max)
    print("f(%.2f, %.2f) = %.2f" %(x, y, f))
```

Zadatak 5

5. U toku je šahovska partija. Na potezu je beli takmičar. Napisati program koji štampa informaciju o tome da li beli takmičar može svojom kraljicom da odnese crnog pešaka, pri čemu se zadaju trenutna pozicija bele kraljice i crnog pešaka koga takmičar želi da odnese. Pozicije se zadaju pomoću celobrojnih vrednosti (a – 1, b – 2, …, h – 8).

   Primer: pozicija kraljice: 1 6 = a6
   pozicija pešaka: 3 8 = c8
   odgovor: može

Zadatak 5

```
kraljica_x = int(input())
kraljica_y = int(input())
pesak_x = int(input())
pesak_y = int(input())

if kraljica_x == pesak_x or kraljica_y == pesak_y or
                        abs(kraljica_x - pesak_x) == abs(kraljica_y - pesak_y):
    print("može")
else:
    print("ne može")
```

Zadatak 6 – zadatak za vežbu

6. Napisati program kojim se za uneta dva cela broja **N** i **M** kreira broj tako što se naizmenično uzimaju cifre iz prvog, pa iz drugog broja. Ukoliko neki broj ima više cifara, na kraj se dodaju sve njegove preostale cifre.

   Primer:

   Ulaz: 1357 2468 **Izlaz**: 12345678

   Ulaz: 1357 24 **Izlaz**: 123457

Zadatak 6

```
def daj_broj_cifara(x):
    broj_cifara_X = 0

    if(x != 0):
        broj_cifara_X = 1
        x = x // 10

        while(x > 0):
            broj_cifara_X = broj_cifara_X * 10
            x = x // 10

    return broj_cifara_X

N = int(input("Unesi N: "))
M = int(input("Unesi M: "))

broj_cifara_N = daj_broj_cifara(N)
broj_cifara_M = daj_broj_cifara(M)

novi_broj = 0
```

Zadatak 6

```
while(broj_cifara_N > 0 or broj_cifara_M > 0):

    if(broj_cifara_N > 0):
        cifra_N = N // broj_cifara_N #prva cifra sleva
        N = N % broj_cifara_N        #ostatak cifara
        broj_cifara_N = broj_cifara_N // 10  #smanjujemo broj cifara jer se i N smanjilo

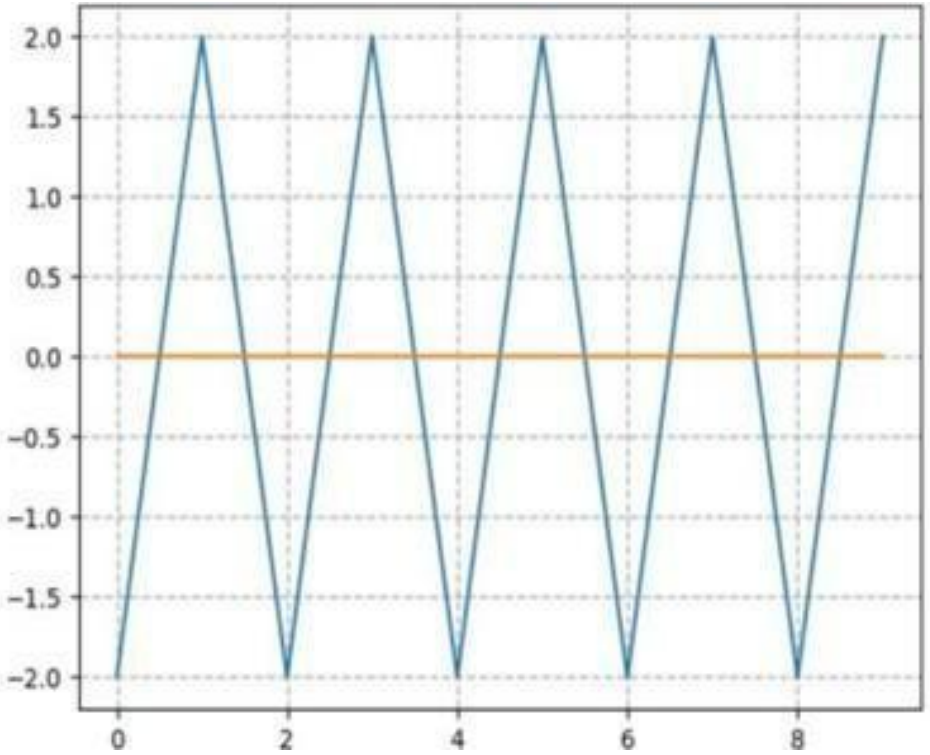
        novi_broj = novi_broj * 10 + cifra_N

    if(broj_cifara_M > 0):
        cifra_M = M // broj_cifara_M
        M = M % broj_cifara_M
        broj_cifara_M = broj_cifara_M // 10

        novi_broj = novi_broj * 10 + cifra_M

print(novi_broj)
```

7. Data je realna funkcija definisana nad intervalom [0, +∞] čiji grafik možete videti na slici. Napisati program kojim se za uneto **x** izračunava vrednost funkcije **y** (periodična funkcija sa periodom 2).



Zadatak 7

```
x = float(input("Unesite x: "))
x1 = x - (x // 2) * 2 #svodimo na interval [0,2] zbog
periodičnosti funkcije

if x1 < 1:
        #y = kx+n - jednačina prave
        #prava preseca Oy osu u tački (0,n)
        #k = (y2-y1)/(x2-x1), gde su (x1,y1) i (x2,y2) koordinate
dveju različitih tačaka prave
        y = 4 * x1 - 2
else:
        y = -4 * x1 + 6

print(y)
```

Zadatak 8

8. U dve kutije se nalaze samo crvene i bele kuglice. U prvoj kutiji se nalazi **C1** crvenih i **B1** belih, a u drugoj **C2** crvenih i **B2** belih.

a) Napisati program koji određuje najmanji broj premeštanja kuglica tako da posle premeštanja u svakoj kutiji budu kuglice iste boje. U svakom premeštanju se prebacuje jedna kuglica.
b) Napisati program koji određuje najmanji broj premeštanja kuglica tako da posle premeštanja dve kutije imaju identičan sastav. U svakom premeštanju se prebacuje jedna kuglica.

Zadatak 8

```
C1 = int(input("Broj crvenih kuglica u prvoj kutiji: "))
B1 = int(input("Broj belih kuglica u prvoj kutiji: "))

C2 = int(input("Broj crvenih kuglica u drugoj kutiji: "))
B2 = int(input("Broj belih kuglica u drugoj kutiji: "))

#a)

print("Broj premestaja tako da u svakoj kutiji budu kuglice iste boje: ", min(C1 + B2, C2 + B1))

#b)


if((C1 + C2) % 2 == 0 and (B1 + B2) % 2 == 0):
    broj_premestaja_crvenih = max(C1, C2) - ((C1 + C2) // 2)
    broj_premestaja_belih = max(B1, B2) - ((B1 + B2) // 2)

    print("Broj premestaja tako da kutije imaju identican sastav: ",
                                                broj_premestaja_crvenih + broj_premestaja_belih)
else:
    print("Nemoguc je identican sastav kutija")
```