

# PROGRAMIRANJE

ZBIRKA ZADATAKA

Ishod učenja 3

2023-2024.

## Zadatak 1

Napišite funkciju koja prima iznos temperature i vraća pozivatelju tekst prognoze prema pravilima:

- Ako je primljena vrijednost temperature između 15 i 30 stupnjeva, vratite tekst prognoze „Taman“.
- Ako je primljena vrijednost temperature iznad 30 stupnjeva, vratite tekst prognoze „Prevrucе“.
- Ako je primljena vrijednost temperature ispod 15 stupnjeva, vratite tekst prognoze „Prehladno“.

Pozovite funkciju iz glavnog dijela programa tri puta kako biste demonstrirali sva tri slučaja i ispišite vraćene tekstove.

*Moguće rješenje:*

---

```
def pretvori_u_tekst(temperatura):  
    if temperatura > 30.0:  
        return "Prevrucе"  
    elif temperatura < 15.0:  
        return "Prehladno"  
    else:  
        return "Taman"  
  
print(pretvori_u_tekst(7.5))  
print(pretvori_u_tekst(32.0))  
print(pretvori_u_tekst(22.5))
```

## Zadatak 2

Napišite funkciju koja prima prirodni broj N i vraća je li broj prost ili nije. Pozovite funkciju za sve brojeve između 1 i 500 te ispišite samo one proste.

*Moguće rješenje:*

---

```
def je_li_prost(broj):
    for i in range(2, broj):
        if broj % i == 0:
            return False
    return True

for i in range(1, 501):
    if je_li_prost(i):
        print(i, end=" ")
```

### Zadatak 3

Napišite funkciju koja prima formulu molekule i ispisuje količinu atoma kisika (element oznake O) u toj molekuli. Formula molekule se uvijek sastoji od točno četiri znaka i ima sljedeći format:

- Prvi znak je oznaka prvog elementa (primjerice, C, O, N, itd.)
- Drugi znak je količina atoma prvog elementa (broj između 1 i 9)
- Treći znak je oznaka drugog elementa (primjerice, C, O, N, itd.)
- Četvrti znak je količina atoma drugog elementa (broj između 1 i 9)

Primjerice, ako funkcija primi formulu „H2O3“, treba ispisati 3 ako primi „O2N9“, treba ispisati 2 ako primi „C6H6“, treba ispisati 0. Pozovite funkciju iz glavnog dijela programa.

*Moguće rješenje:*

---

```
def ispisi_kolicinu_kisika(formula):
    print("U formuli", formula, "ima atoma kisika: ", end="")

    if formula[0] == "O":
        print(formula[1])
    elif formula[2] == "O":
        print(formula[3])
    else:
        print("0")

ispisi_kolicinu_kisika("H2O3")
ispisi_kolicinu_kisika("O2N9")
ispisi_kolicinu_kisika("C6H6")
```

#### Zadatak 4

Napišite funkciju `prikazi_sastav` koja od korisnika učitava formulu molekule i ispisuje pune nazive elemenata od kojih se molekula sastoji. Količine pojedinih elemenata u formuli mogu biti izostavljene ili napisane u obliku cijelog broja i uvijek ih treba zanemariti. U formuli se smiju pojaviti samo elementi:

- Au – zlato
- Se – selenij
- Ag – srebro
- Cl – klor
- Al – aluminij
- Si – silicij

Primjeri za testiranje:

- Ako funkcija učitava formulu „Au2Se3“, treba ispisati: zlato selenij
- Ako funkcija učitava formulu „AgCl“, treba ispisati: srebro klor
- Ako funkcija učitava formulu „AuCl3“, treba ispisati: zlato klor
- Ako funkcija učitava formulu „Al6Cl13Si2“, treba ispisati: aluminij klor silicij

Napišite i koristite dodatne funkcije prema želji. Pozovite funkciju `prikazi_sastav` iz glavnog dijela programa.

*Moguće rješenje:*

```
def je_li_broj(c):
    return c in "0123456789"

def vrati_naziv_elementa(z1, z2):
    if z1 == 'A' and z2 == 'u':
        return "zlato"
    elif z1 == 'S' and z2 == 'e':
        return "selenij"
    elif z1 == 'A' and z2 == 'g':
        return "srebro"
    elif z1 == 'C' and z2 == 'l':
        return "klor"
    elif z1 == 'A' and z2 == 'l':
        return "aluminij"
    elif z1 == 'S' and z2 == 'i':
        return "silicij"
    else:
        return "greska"

def prikazi_sastav():
    formula = input("Upisite formulu: ")

    # Uvijek gledamo i-ti znak i njegovog sljedbenika
    skip_next = False

    for i in range(len(formula) - 1):
        if skip_next:
            skip_next = False
            continue
```

```
if not je_li_broj(formula[i]):  
    # Buduci da trenutni znak nije broj, nasao sam pocetak elementa  
    print(vrati_naziv_elementa(formula[i], formula[i + 1]), end=" ")  
    skip_next = True # "premotamo" jos jedan znak jer smo ga obradili
```

```
prikazi_sastav()
```

## Zadatak 5

Napišite funkciju koja od korisnika učitava cijeli broj između 1 i 5 (ako korisnik upiše neki drugi cijeli broj, ponavljajte učitavanje dok ne upiše broj između 1 i 5). Nakon ispravnog učitavanja, neka funkcija vrati učitani broj pozivatelju. U glavnom dijelu programa napravite listu i pomoću funkcije učitajte u nju brojeve između 1 i 5. Na kraju, ispišite sadržaj liste. Primjer rada programa:

```
Ucitavam 1. broj
Upisite broj: 4
Ucitavam 2. broj
Upisite broj: 5
Ucitavam 3. broj
Upisite broj: 3
Ucitavam 4. broj
Upisite broj: 9
Upisite broj: 12
Upisite broj: 1
Ucitavam 5. broj
Upisite broj: 6
Upisite broj: 5
Ispis brojeva: 4 5 3 1 5
```

*Moguće rješenje:*

---

```
def ucitaj():
    while True:
        broj = int(input("Upisite broj: "))
        if broj >= 1 and broj <= 5:
            break
    return broj

brojevi = []

for i in range(5):
    print("Ucitavam " + str(i + 1) + ". broj")
    ucitani_broj = ucitaj()
    brojevi.append(ucitani_broj)

print("Ispis brojeva: ", end="")
for i in range(len(brojevi)):
    print(brojevi[i], end=" ")
```

## Zadatak 6

Napišite funkciju koja prima  $a$  = iznos ubrzanja (u  $\text{m/s}^2$ ) i  $t$  = duljinu trajanja ubrzanja te vraća koliki put  $d$  je prošlo vozilo za to vrijeme. Koristite formulu za izračun puta:  $d = \frac{1}{2}at^2$ . Primjerice, ako vozilo 10 sekundi ubrzava  $2.1 \text{ m/s}^2$ , onda će za to vrijeme proći 105 metara. Pozovite funkciju iz glavnog dijela programa s istom akceleracijom i vremenima od 1, 10, 30 i 60 sekundi i ispišite rezultate.

*Moguće rješenje:*

---

```
import math

def put(a, t):
    return (1 / 2) * a * math.pow(t, 2)

print("Put prijedjen za 1 s =", put(2.1, 1.0), "m")
print("Put prijedjen za 10 s =", put(2.1, 10.0), "m")
print("Put prijedjen za 30 s =", put(2.1, 30.0), "m")
print("Put prijedjen za 60 s =", put(2.1, 60.0), "m")
```



## Zadatak 7

Napišite program koji od korisnika učitava dva decimalna broja i operaciju (zbrajanje, oduzimanje, množenje ili dijeljenje). Ispišite rezultat tražene operacije koristeći po jednu funkciju za svaku od operacija.

*Moguće rješenje:*

---

```
def zbroji(a, b):
    print(a, "+", b, "=", a + b)

def oduzmi(a, b):
    print(a, "-", b, "=", a - b)

def mnozi(a, b):
    print(a, "*", b, "=", a * b)

def dijeli(a, b):
    print(a, "/", b, "=", a / b)

a = float(input("Upisite a: "))
b = float(input("Upisite b: "))
op = input("Upisite operaciju: ")

if op == '+':
    zbroji(a, b)
elif op == '-':
    oduzmi(a, b)
elif op == '*':
    mnozi(a, b)
elif op == '/':
    dijeli(a, b)
else:
    print("Nepoznata operacija")
```

## Zadatak 8

UART je komunikacijski protokol koji se izvorno koristio za komunikaciju perifernih uređaja s računalom (primjerice, printer), a danas se uglavnom koristi na ugradbenim pločicama za komunikaciju mikroupravljača s ostalim komponentama (primjerice, modemom). UART paket izgleda ovako:

- 1. znak je uvijek '0'.
- Sljedećih 8 znakova su '0' ili '1' i predstavljaju poruku koju se želi prenijeti.
- Znak prije zadnjeg je paritetni znak koji se izračunava ovako: prebroje se sve jedinice u poruci i ako ih je paran broj, paritetni znak je '0', a ako ih je neparan broj, paritetni znak je onda '1'. Pošiljalac postavlja paritetni znak i primatelj ga ponovno provjerava. Ako postoji razlika, onda se dogodila pogreška u prijenosu.
- Zadnji znak je uvijek '1'.

Primjer poruke: 00100000101 – u primjeru vidimo da poruka sadrži 2 jedinice (paran broj jedinica), stoga je predzadnji, paritetni znak postavljen na '0'.

Napišite funkciju koja prima string s kompletnom primljenom porukom i vraća pozivatelju je li poruka ispravna provjerom prvog, zadnjeg i paritetnog znaka. Pozovite funkciju iz glavnog dijela programa i ispišite je li ispravna ili neispravna.

*Moguće rješenje:*

```
def poruka_ispravna(poruka):
    if poruka[0] != '0':
        return False # Prvi znak nije '0'

    kolicina = 0
    for i in range(1, 9):
        if poruka[i] == '1':
            kolicina += 1

    if kolicina % 2 == 0 and poruka[9] == '1':
        # Postoji parni broj jedinica u poruci, ali je paritetni znak 0
        return False
    elif kolicina % 2 == 1 and poruka[9] == '0':
        # Postoji neparan broj jedinica u poruci, ali je paritetni znak 1
        return False

    if poruka[10] != '1':
        return False # Zadnji znak nije '1'

    return True # Nema gresaka

if poruka_ispravna("00100000101"):
    print("Poruka je ispravna")
else:
    print("Poruka je neispravna")
```

## Zadatak 9

Napišite funkciju koja prima string i ispisuje ga tako da prvo ispiše „>->-> “ , zatim znakove stringa razdvojene razmakom i na kraju „ <-<-<“. Pozovite funkciju s nekim stringom.

*Moguće rješenje:*

---

```
def ispisi_naglaseno(tekst):  
    print(">->-> ", end="")  
    for i in range(len(tekst)):  
        print(tekst[i], end=" ")  
    print("<-<-<")  
  
ispisi_naglaseno("nova knjiga!")
```

## Zadatak 10

Napišite funkciju koja prima riječ i vraća je li palindrom. Palindrom je antički naziv za riječ koja čitana od početka prema kraju ili obrnuto glasi jednako (primjerice: kisik, ratar, potop). Pretpostavite da će riječ uvijek sadržavati mala slova bez ikakvih dodatnih znakova.

*Moguće rješenje:*

---

```
def palindrom(tekst):
    indeks_s_kraja = len(tekst) - 1

    indeks_polovice = int(len(tekst) / 2)
    for i in range(indeks_polovice):
        if tekst[i] != tekst[indeks_s_kraja]:
            return False
        indeks_s_kraja -= 1

    return True

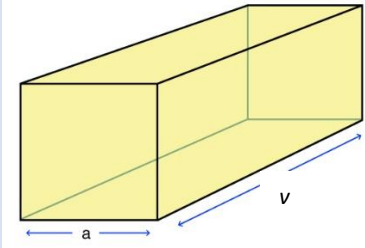
print(palindrom("nova knjiga!"))
print(palindrom("kisik"))
print(palindrom("brankova ulica"))
print(palindrom("ratar"))
print(palindrom("potop"))
print(palindrom("racecar"))
print(palindrom("saippuakivikauppias"))
print(palindrom("hannah"))
```

## Zadatak 11

Napišite jednu funkciju koji prima sve potrebne parametre od pozivatelja te izračunava i vraća volumen odabrane pravilne prizme. Neka funkcija zna izračunati volumene sljedećih pravilnih prizmi:

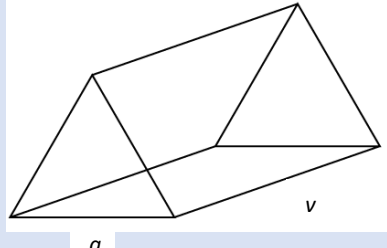
Kvadratna prizma ima volumen:

$$V = a^2 v$$



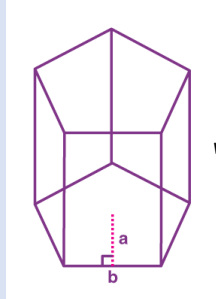
Trokutasta prizma ima volumen:

$$V = \frac{\sqrt{3}}{4} a^2 v$$



Pentagonalna prizma ima

$$\text{volumen: } V = \frac{5}{2} a b v$$



Moguće rješenje:

```
import math
```

```
def volumen(vrsta_prizme, v, a, b):  
    if vrsta_prizme == 'K':  
        return math.pow(a, 2) * v  
    elif vrsta_prizme == 'T':  
        return (math.sqrt(3) / 4) * math.pow(a, 2) * v  
    elif vrsta_prizme == 'P':  
        return (5.0 / 2.0) * a * b * v
```

```
print(volumen('K', 9.5, 4.5, 0.0))  
print(volumen('T', 9.5, 4.5, 0.0))  
print(volumen('P', 9.5, 4.5, 5.0))
```

## Zadatak 12

Napišite program koja od korisnika učitava parni broj N i crta kvadrat stranice N (ako korisnik unese neparni broj, nemojte iscrtati ništa). Za crtanje treba koristiti '\*' za jednu četvrtinu kvadrata, 'o' za drugu, '#' za treću i 'x' za četvrtu. Koristite funkcije gdje je zgodno kako biste pojednostavili izgled programa. Primjerice, za N=6 program, ispis može izgledati ovako:

```
***ooo
***ooo
***ooo
xxx###
xxx###
xxx###
```

*Moguće rješenje:*

```
def iscrtaj_znak(stupac, n, redak):
    if stupac <= int(n / 2) and redak <= int(n / 2): # lijevo gore
        print('*', end='')
    elif stupac <= int(n / 2) and redak > int(n / 2): # lijevo dolje
        print('x', end='')
    elif stupac > int(n / 2) and redak <= int(n / 2): # desno gore
        print('o', end='')
    else: # desno dolje
        print('#', end='')

def iscrtaj_redak(redak, n):
    for stupac in range(1, n + 1):
        iscrtaj_znak(stupac, n, redak)
    print()

def iscrtaj_kvadrat(n):
    for redak in range(1, n + 1):
        iscrtaj_redak(redak, n)

n = int(input("Upisite stranicu: "))

if n % 2 == 0:
    iscrtaj_kvadrat(n)
```

## Zadatak 13

Napišite funkciju koja prima cijeli broja  $a$  i vraća sljedeću vrijednost:

- Ako je  $a$  manji od 20, vratite 20.
- Ako je  $a$  veći od 30, vratite 30.
- U ostalim slučajevima vratite  $a$ .

U glavnom dijelu programa pripremite listu od 5 cijelih brojeva. Za svaki broj iz liste pozovite funkciju i ispišite vraćenu vrijednost.

*Moguće rješenje:*

---

```
def ograniči(a):  
    if a < 20:  
        return 20  
    elif a > 30:  
        return 30  
    else:  
        return a  
  
brojevi = [ -8, 523, 3, 24, 29 ]  
  
for i in range(len(brojevi)):  
    print(ograniči(brojevi[i]))
```

## Zadatak 14

Napišite funkciju koja simulira rad aparata za kavu. Neka funkcija primi ubačeni iznos novca i odabir kave (kratka kava, duga kava, cappuccino ili espresso, a pretpostavite da sve kave koštaju jednako i to 1 EUR) te ispiše koju je kavu pripremila i u kojim kovanicama je vratila ostatak novca. Neka funkcija uvijek vraća ostatak novca samo u kovanicama (2 EUR ili 1 EUR) i to u što manjoj količini. Primjerice, ako treba vratiti 5 EUR, onda funkcija uvijek vraća 2 EUR, 2 EUR, 1 EUR. Pazite na situaciju kad korisnik ubaci premalo novca.

*Moguće rješenje:*

```
def naziv_kave(odabir):
    if odabir == 1: return "kratka kava"
    elif odabir == 2: return "duga kava"
    elif odabir == 3: return "cappuccino"
    elif odabir == 4: return "espresso"

def ispisi(odabir):
    print("Pripremam:", naziv_kave(odabir))

def vrati_novac(ubaceni_novac):
    print("\nPovrat novca:")
    while ubaceni_novac > 0:
        if ubaceni_novac >= 2:
            print("Vracam 2 EUR")
            ubaceni_novac -= 2
        else:
            print("Vracam 1 EUR")
            ubaceni_novac -= 1

def skuhaj_kavu(odabir, ubaceni_novac):
    cijena_kave = 1
    if ubaceni_novac < cijena_kave:
        print("Nedovoljno sredstava")
        return

    ispisi(odabir)
    vrati_novac(ubaceni_novac - cijena_kave)

print("Odaberite kavu: ")
print("1 = kratka kava")
print("2 = duga kava")
print("3 = cappuccino")
print("4 = espresso")
odabir = int(input("> "))

novac = int(input("Ubacite novac: "))

skuhaj_kavu(odabir, novac)
```



## Zadatak 15

Napišite funkciju koja prima dva cijela broja i ispisuje sve brojeve od manjeg do većeg (pazite, ne znamo koji od dva primljena broja je manji, a koji je veći). Pozovite funkciju tri puta: u prvom pozivu neka je prvi parametar manji od drugog, u drugom pozivu neka je prvi parametar veći od drugog, a u trećem pozivu neka su im vrijednosti jednake.

*Moguće rješenje:*

---

```
def ispisi(a, b):  
    # Ako je a < b, zamijenimo vrijednosti  
    if a > b:  
        temp = a  
        a = b  
        b = temp  
  
    for i in range(a, b + 1):  
        print(i)
```

```
ispisi(1, 10)  
print()  
ispisi(10, 1)  
print()  
ispisi(5, 5)
```

## Zadatak 16

Napišite funkciju koja prima string sastavljen isključivo od malih slova. Neka funkcija sva uzastopna pojavljivanja nekog slova  $S$  ispiše skraćeno u obliku  $NS$ , gdje je  $N$  broj uzastopnog pojavljivanja slova  $S$ . Ako se slovo pojavi samo jednom, neka ga ispiše normalno. Primjerice, ako funkcija primi string SAAARRMMMMMA, treba ispisati: S3A2R5MA (jer imamo jedno S pa ga samo ispišemo, zatim imamo tri slova A pa pišemo 3A, dva slova R pa pišemo 2R, pet slova M pa pišemo 5M i jedno slovo A koje samo ispišemo).

*Moguće rješenje:*

```
def ispisi(s):
    prethodni_znak = s[0]
    broj_ponavljanja = 1

    for i in range(1, len(s)):
        if s[i] == prethodni_znak:
            # Nastavljamo niz
            broj_ponavljanja += 1
        else:
            # Niz je prekinut, ispisemo
            if broj_ponavljanja == 1:
                print(prethodni_znak, end="")
            else:
                print(str(broj_ponavljanja) + prethodni_znak, end="")

            broj_ponavljanja = 1

        prethodni_znak = s[i]

    # Na kraju ispisemo zadnje slovo
    if broj_ponavljanja == 1:
        print(prethodni_znak, end="")
    else:
        print(str(broj_ponavljanja) + prethodni_znak, end="")

    print()

ispisi("SAAARRMMMMMA")
ispisi("SARMA")
ispisi("SSAARRMMAA")
ispisi("SARRRRRRRRRRRRRRRRRRRMA")
```

## Zadatak 17

Napišite funkciju koja prima string i znak i vraća koliko se puta u stringu pojavljuje taj znak. Pozovite funkciju iz glavnog dijela programa i ispišite rezultat.

*Moguće rješenje:*

---

```
def prebroji(s, znak):
    n = 0

    for i in range(len(s)):
        if s[i] == znak:
            n += 1

    return n

recenica = "a common mistake that people make when trying to design something
completely foolproof is to underestimate the ingenuity of complete fools"
znak = "o"

kolicina = prebroji(recenica, znak)

print("Pronadjeno znakova:", kolicina)
```

## Zadatak 18

Napišite funkciju koja prima string koji predstavlja MAC adresu i vraća pozivatelju je li MAC adresa ispravna. MAC adresu čine 12 heksadecimalnih brojeva, a iza svaka dva broja dolazi crtica, osim iza zadnjeg. Pozovite funkciju s nekoliko ispravnih i neispravnih MAC adresa iz glavnog dijela programa. Primjer ispravne MAC adrese: 00-B0-D0-63-C2-26

*Moguće rješenje:*

```
def heksadecimalni_broj(c):
    # Provjerimo je li c znamenka 0-9 ili slovo a-f ili A-F.
    return c >= '0' and c <= '9' or c >= 'a' and c <= 'f' or c >= 'A' and c <= 'F'

def je_li_ispravna(mac):
    if len(mac) != 17:
        return False

    for i in range(len(mac)):
        if i == 2 or i == 5 or i == 8 or i == 11 or i == 14:
            # Trazimo crticu
            if mac[i] != '-':
                return False
        else:
            if not heksadecimalni_broj(mac[i]):
                return False

    return True

print(je_li_ispravna("00-B0-D0-63-C2-26"))
print(je_li_ispravna("00-B0-D0-63-C2-2"))
print(je_li_ispravna("FF-B1-D0-63-C2-26"))
print(je_li_ispravna("00-G0-D0-63-C2-26"))
print(je_li_ispravna("00-B0-D0-63-C2"))
print(je_li_ispravna("50-81-40-FC-92-70"))
print(je_li_ispravna("00-15-5D-12-FD-9C"))
print(je_li_ispravna("0A-00-27-00-00-16"))
print(je_li_ispravna("54-14-F3-FF-0E-E8"))
```

## Zadatak 19

Napišite jednu funkciju koja prima potrebne parametre te izračunava i vraća gustoću, masu ili volumen, prema korisnikovoj želji. Koristite sljedeće formule:

- Gustoća  $\rho = \frac{m}{V}$
- Masa  $m = \rho V$
- Volumen  $V = \frac{m}{\rho}$

Pozovite funkciju tri puta kako biste dobili i ispisali prvo gustoću pa masu pa volumen.

*Moguće rješenje:*

```
def izracunaj(sto, ro, m, v):  
    if sto == 'R': return m / v  
    elif sto == 'M': return ro * v  
    elif sto == 'V': return m / ro  
  
print("Gustoca je", izracunaj('R', 0.0, 2.75, 0.35), " kg/m3")  
print("Masa je", izracunaj('M', 7.857, 0.0, 0.35), " kg")  
print("Volumen je", izracunaj('V', 7.857, 2.75, 0.0), " m3")
```

## Zadatak 20

Napišite funkciju koja prima dva stringa jednake duljine i ispisuje tri retka:

- U prvom retku ispisuje znakove koji su u oba stringa nalaze na istom indeksu i jednaki su. Umjesto ostalih znakova ispisuju se podvlake.
- U drugom retku ispisuje znakove iz prvog stringa koji na istom indeksu u drugom stringu nemaju jednak znak. Umjesto ostalih znakova ispisuju se podvlake.
- U trećem retku ispisuje znakove iz drugog stringa koji na istom indeksu u prvom stringu nemaju jednak znak. Umjesto ostalih znakova ispisuju se podvlake.

Primjerice, ako su dva stringa „easefulnesses“ i „easygoingness“, funkcija treba ispisati:

```
eas__n__s
__eful_esse_
__ygoi_gnes_
```

*Moguće rješenje:*

```
def ispisi(a, b):
    # Prvi redak
    for i in range(len(a)):
        if a[i] == b[i]:
            print(a[i], end="")
        else:
            print("_", end="")
    print()

    # Drugi redak
    for i in range(len(a)):
        if a[i] != b[i]:
            print(a[i], end="")
        else:
            print("_", end="")
    print()

    # Treci redak
    for i in range(len(a)):
        if a[i] != b[i]:
            print(b[i], end="")
        else:
            print("_", end="")
    print()

ispisi("easefulnesses", "easygoingness")
print()
ispisi("branka", "krafna")
print()
ispisi("funny", "funny")
print()
```