

# OPISI ALGORITAMA



Agencija za odgoj i obrazovanje  
Education and Teacher Training Agency



HRVATSKI SAVEZ  
INFORMATIČARA



Ministarstvo znanosti,  
obrazovanja i sporta

### 5.1. Zadatak: Hazena

Autor: Nikola Dmitrović

Ukupan broj postignutih golova na utakmici jednak je zbroju dva učitana broja. Naši su postigli veći od ta dva broja, a to možemo odrediti if naredbom.

### 5.2. Zadatak: Jaja

Autor: Nikola Dmitrović

Naredbama if i else if (odnosno elif u Pythonu) prema tablici iz zadatka ispišemo tražene podatke u odnu na učitani broj.

### 5.3. Zadatak: Sheldon

Autor: Nikola Dmitrović

Najkraće i najjednostavnije rješenje ovog zadatka jest broj zapravo učitati kao string. Zatim za svaku znamenku napišemo if naredbu kako bismo odrediti je li ta znamenka lijepa. Npr. za prvu znamenku if broj[0] == '1': rj += 1

### 6.1. Zadatak: Kovanice

Autor: Nikola Dmitrović

Za svaku vrijednost if naredbom odredimo trebamo li ju ispisati.

### 6.2. Zadatak: Geoid

Autor: Nikola Dmitrović

U ovom zadatku potrebno je znanje for petlje za učitavanje rješenja učenika. Za svako učitano rješenje if naredbama odredimo pripadajući brojač koji trebamo povećati (prava\_formula, kriva\_formula, ne\_zna\_racunati).

### 6.3. Zadatak: Satovi

Autor: Gabrijel Jambrošić

Iterirati ćemo po početnim vremenima satova i odmora te ćemo se zaustaviti kada naiđemo sat ili odmor u kojemu je zadano vrijeme. Sva vremena možemo pretvoriti u minute radi lakšeg računanja. Satovi i odmori se odvijaju naizmjenice pa ćemo tako i dodavati vremena. Još moramo paziti da kada dođemo do velikog odmora dodamo odgovarajuće vrijeme.

*Programski kod (pisan u Python 3)*

```
n = int(input())
m = int(input())
k = int(input())
a = int(input())
b = int(input())
```

```

vrijeme = a * 60 + b
trenutno = 8 * 60
brojac = 0
while trenutno <= vrijeme:
    if brojac % 2 == 0:
        trenutno += 45
    else:
        if (brojac + 1) // 2 == k:
            trenutno += m
        else:
            trenutno += n
        brojac += 1
print((brojac + 1) // 2, end = '. ')
if brojac % 2 == 0:
    print("odmor")
else:
    print("sat")

```

**Potrebno znanje:**

**Kategorija:** ad hoc

## 7.1. Zadatak: Dubioza

Autor: Nikola Dmitrović

Učitamo obje riječi zadane u ulazu i ispišemo prvu riječ počevši od znaka koji je *duljinadrugeriječi*-ti poredu. Npr. ako je prva riječ duga 10 znakova, a druga 4 znaka, ispišemo prvu riječ od 5. znaka do kraja.

## 7.2. Zadatak: Karte

Autor: Gabrijel Jambrošić

Trebamo simulirati igru, pamtit ćemo preostale karte za svakog od tri igrača te provjeriti koju treba baciti. Nakon što igrač baci kartu, moramo zabilježiti da ju više ne možemo odabrati. Možemo npr. karte pamtit u poljima, a kada iskoristimo neku, onda ju prebrišemo s nulom.

*Programski kod (pisan u Python 3)*

```

def daj_kartu(karte, minimum):
    najmanja = 10
    for i in range(len(karte)):
        if karte[i] != 0 and karte[i] < najmanja and karte[i] > minimum:
            najmanja = karte[i]
            ind = i
    if najmanja == 10:
        for i in range(len(karte)):
            if karte[i] != 0 and karte[i] < najmanja:
                najmanja = karte[i]
                ind = i
    karte[ind] = 0
    return najmanja

def ucitaj_karte(polje):
    k1, k2, k3 = map(int, input().split())
    polje.append(k1)
    polje.append(k2)
    polje.append(k3)

karte = [[], [], []]
bodovi = [0, 0, 0]
ucitaj_karte(karte[0])
ucitaj_karte(karte[1])
ucitaj_karte(karte[2])
for krug in range(3):
    najveca = 0
    for igrac in range(3):
        trenutna_karta = daj_kartu(karte[igrac], najveca)
        print(trenutna_karta, end = ' ')
        if trenutna_karta > najveca:
            najveca = trenutna_karta
            tko = igrac
    bodovi[tko] += 1

```

```
print()
for i in bodovi:
    print(i, end = ' ')
print()
```

## Potrebno znanje:

**Kategorija:** ad hoc

### 7.3. i 8.2. Zadatak: Bubnjevi

Autor: Gabrijel Jambrošić

Za odgovor na prvo pitanje za svaki udarac u ritmu for petljom prođemo po svim udarcima prije njega i if naredbom vidimo jesmo li već udarili u taj bubanj. Ako jesmo, zapamtimo to u neku varijablu.

Za odgovor na drugo pitanje idemo redom po udarcima. Ako smo u  $i$ -tom udarcu udarili isti bubanj kao i u  $i-1$  udarcu, povećamo neki brojač za 1. Ako je taj brojač postao  $K$ , *promjenimo*  $i$ -ti udarac, odnosno računamo da je udario bilo koji drugi bubanj, resetiramo brojač na 0 i povećamo rješenje za 1.

### 8.1. Zadatak: Lijep

Autor: Gabrijel Jambrošić

Kao i u 5.3. Sheldon zadatku, učitamo broj kao string. Međutim nećemo (iako bismo mogli) pisati ifove za svaku znamenku već možemo for petlju tako da rješenje glasi

```
for i in range(len(broj)):
    if broj[i]==i+1+ord('0'): rj += i+1+ord('0')
```

### 8.3. Zadatak: Ograda

Autor: Patrick Pavić

Za odgovor na prvo pitanje odredimo najduži segment visina koje se nalaze između dvije nule.

Za odgovor na drugo pitanje treba nam sljedeći algoritam:

1. Pronađemo sve segmente visina koje se nalaze između dvije nule i za svaki od njih povećamo rješenje za jedan
2. U svakom segmentu iz prvog koraka pronademo najmanju visinu i sve pozicije u tom segmentu na kojima se nalazi ta najmanja visina postavimo na 0
3. Ponavljamo korake 1. i 2. dok svi brojevi ne postanu 0, odnosno kada nebude više segmenata

Za odgovor na treće pitanje probamo svaki stupac izbaciti iz liste, a za sve preostale u listi ponoviti algoritam za odgovor na drugo pitanje.