



## Pravila

Kolokvij se piše 120 minuta. Ukupno je moguće ostvariti 100 bodova. Rezultati će biti objavljeni na Teams kanalu kolegija. Sve kodove spremite u jedinstvenu zip ili rar datoteku s imenom "KOL\_1\_PREZIME\_IME" te ju predajte na e-mail adresu tprusina@mathos.hr.

---

## Teorijski zadaci

---

**Zadatak 1 (15 bodova).** Primjenom master teorema asimptotski omeđite sljedeće funkcije.

- a)  $T(n) = 9T(n/3) + n^2$ ,
- b)  $T(n) = 9T(n/3) + n^3$ ,
- c)  $T(n) = 2T(n/2) + n$ ,
- d)  $T(n) = 2T(n/2) + c$ ,
- e)  $T(n) = 3T(n/2) + n$ ,
- f)  $T(n) = 16T(n/4) + n^4$ .

**Zadatak 2 (15 bodova).** Dan je algoritam 1.1.

---

PROGRAM 1.1. Dvije for petlje

---

```
procedure LOOP( $A, n$ )  
  for  $i = 0 \dots n$   
    for  $j = i + 1 \dots n$   
      if ( $A[i] > A[j]$ )  
        SWAP( $A, i, j$ );
```

```
  return
```

---

Izrazite vrijeme izvršavanja algoritma 1.1 u  $\Theta$  notaciji i svoju tvrdnju argumentirajte. Pretpostavite da procedura SWAP radi u vremenu  $\Theta(1)$ .



**Zadatak 3 (10 + 10 bodova).** Standardni merge sort algoritam (dan kao procedura merge-sort) sortira polje tako da ga razdvoji na dva polja jednake duljine, rekurzivno ih sortira te ih spoji koristeći proceduru merge. U ovom zadatku potrebno je analizirati algoritam koji polje razdvoji na **četiri** polja jednakih duljina, rekurzivno ih sortira i spaja u polazni niz.

- Napišite pseudokod procedure merge-sort opisane u tekstu zadatka. Procedura merge i dalje prima dva sortirana polja. Pretpostavite da vam je takva merge procedura dana na korištenje i nju ne morate dodatno pojašnjavati.
- Dajte asimptotske granice na vrijeme izvršavanja ovako definiranog algoritma. Tvrdnje dokažite koristeći stablo rekurzije.

## Programerski zadaci

**Zadatak 4 (10 + 5 bodova).** Dan je vektor duljine  $n$  čiji elementi su brojevi od 0 do  $k$  ali točno jedan broj iz tog intervala nedostaje. Napišite mali program koji će pronaći koji točno broj nedostaje.

```
int missing(vector<int> A, int k);
```

Pretpostavite da će uvijek biti dan  $k \leq n$ . Za dodatnih 5 bodova veše rješenje mora raditi u vremenu  $\Theta(n)$ .

Primjeri	
Input	Output
$A = \{ 2, 2, 3, 0, 3 \}$ $k = 3$	1
$A = \{ 2, 1, 3, 0, 3 \}$ $k = 4$	4
$A = \{ 1, 2, 3, 4, 5, 6, 7 \}$ $k = 7$	0

Tablica 1.1: Test primjeri za zadatak 4



**Zadatak 5 (10 + 5 bodova).** Dan je vektor  $A$  od  $n$  brojeva. Implementirajte funkciju

```
int closest(vector<int> A);
```

koja vraća najmanju apsolutnu razliku između dva elementa na različitim pozicijama vektora, tj. vratite  $\min_{i \neq j} |A[i] - A[j]|$ . Za dodatnih 5 bodova vaše rješenje se mora izvršavati u vremenu  $\Theta(n \log n)$ .

Primjeri	
Input	Output
$A = \{ 4, 2, 5, 10 \}$	1
$A = \{ 1, 3, 5, 10, 1 \}$	0
$A = \{ 2000, 100, 1000, 10 \}$	90

Tablica 1.2: Test primjeri za zadatak 4

**Zadatak 6 (10 bodova).** Implementirajte funkciju

```
int subarray_sum(vector<int> A, int x);
```

koja za dani vektor prirodnih brojeva  $A$  od  $1 \leq n \leq 2 \cdot 10^5$  elemenata  $1 \leq A[i] \leq 10^9$  i prirodan broj  $1 \leq x \leq 10^9$ , izbroji koliko postoji parova  $(i, j)$  takvih da vrijedi  $i \leq j$

$$\left( \sum_{k=i}^j A[k] \right) == x.$$

Vrijeme izvršavanja dane funkcije mora biti  $\Theta(n)$ .

Primjeri	
Input	Output
$A = \{ 2, 4, 1, 2, 7 \}$ $x = 7$	3
$A = \{ 1, 1, 1, 1, 1, 1, 1, 1, 3, 4 \}$ $k = 7$	4
$A = \{ 1, 2, 3, 4, 5, 6, 7 \}$ $x = 8$	0

Tablica 1.3: Test primjeri za zadatak 5