

# UvR

## Dodatne naloge

### 3. del

#### Naloga 1

Napiši algoritem, ki na vhodu dobi celoštevilsko vrednost  $k \geq 0$  in seznam  $k$  števil  $N_1, N_2, \dots, N_k$ . Algoritem naj obrne vrstni red števil na seznamu. Če začetni seznam (dolžine 5,  $k = 5$ ) vsebuje števila:

$$N_1 = 5, N_2 = 13, N_3 = 8, N_4 = 27, N_5 = 10,$$

naj bo vsebina seznama po koncu izvajanja algoritma:

$$N_1 = 10, N_2 = 27, N_3 = 8, N_4 = 13, N_5 = 5.$$

#### Naloga 2

Napiši algoritem za iskanje mediane ( $M$ ) števil v seznamu, ki vsebuje  $N$  različnih števil. Pri razvoju si pomagaj z uporabo že obstoječega algoritma za iskanje največjega števila. Mediana seznama je število iz danega seznama, od katerega je (približno) polovica števil v seznamu manjša in (približno) polovica števil v seznamu večja.

Na primer, mediana seznama:

$$26, 50, 83, 44, 91, 20, 55.$$

je število 50, saj so tri števila (20, 26, 44) manjša in tri števila večja (55, 83, 91). Če je  $N$  sodo, naj bo število manjših števil za ena manjše od števila večjih števil. Na primer, mediana seznama:

$$26, 50, 44, 91, 20, 55$$

je 44, saj sta dve števili manjši (20, 26), tri (50, 55, 91) pa so večja od 44.

#### Naloga 3

V psevdokodi zapiši iterativno in rekurzivno verzijo Evklidovega algoritma za izračun največjega skupnega delitelja danih števil  $a$  in  $b$ .

## Naloga 4

Podan imamo algoritem:

```
i = 0
a = 3
b = 2
while i < n do
  j = 1
  b = b · i
  while j > 0 do
    j = j - 1
    c = a + b
  end while
  a = a · a
  i = i + 1
end while
while i > 0 do
  j = 1
  b = b/i
  while j < n do
    j = j · 2
  end while
  i = i - 2
end while
```

- (a) Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]
- (b) Koliko operacij množenja (v odvisnosti od  $n$ ) se zgodi v enem izvajanju algoritma? [5 točk]
- (c) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvajanju algoritma? [5 točk]

## Naloga 5

Na procesorju z 8 bitnim registrom imamo naslednji program v zbirniku. [15 točk]

```
LOAD ENA
STORE X
LOAD NIC
STORE Y
Z: LOAD X
  COMPARE Y
  JUMPEQ K
  INC X
  JUMP Z
K: STORE N
  HALT
N: .DATA 100
X: .DATA 0
Y: .DATA 0
ENA: .DATA 1
NIC: .DATA 0
```

- Kakšna je časovna zahtevnost zgornjega programa v odvisnosti od  $N$ ?
- Kolikokrat se izvrši ukaz COMPARE Y?
- Kakšna je desetiška vrednost pomnilniške lokacije  $N$  ob koncu izvajanja programa?

## Naloga 6

Podan je algoritem:

```
i = 0
a = 1
b = 2
while i < n do
  j = 1
  while j < n do
    j = j · 2
    a = aj
  end while
  k = 8
  while k ≥ 0 do
    k = k - 4
    b = b + 3
  end while
  i = i + 1
end while
c = a + b
```

(a) Predpostavimo, da je  $n$  pozitivno celo število. Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]

(b) Kakšna je vrednost spremenljivk  $a$ ,  $b$  in  $c$  (v odvisnosti od  $n$ ) po izvajanju algoritma? [5 točk]

## Naloga 7

Kompleksnost [15 točk]

```
i = n
a = 1
b = 0
while i ≥ 1 do
  j = 1
  while j < n do
    j = j · 2
  end while
  k = a
  while k ≥ 0 do
    k = k - 1
    b = b + 3
  end while
  i = i/2
end while
c = a + b
```

(a) Predpostavimo, da je  $n$  pozitivno celo število. Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]

(b) Kolikokrat se izvede operacija seštevanja? [5 točk]

(c) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvajanju algoritma? [5 točk]

## Naloga 8

Algoritma A in B rešujeta isti problem. Za katere vrednosti  $n$  je – glede na časovno zahtevnost – bolj smiselno uporabljati algoritem A oz. B? Upoštevajte operacije, ki so omenjene v kodi. Podajte število operacij v odvisnosti od  $n$ , ki se bodo izvedle v prvem oz. drugem primeru, in ugotovite kateri algoritem je smiselno uporabiti za katere vrednosti  $n$ . [15 točk]

```
Algoritem A:
for(i=0; i<n; i+=c) {
  c*n operacij
}
i=0
while i<100000 {
  i++; // (1 operacija)
}
```

```
Algoritem B:
for(i=n; i>0; i-=1) {
  1100 operacij
}
```

## Naloga 9

Algoritma A in B rešujeta isti problem. Pri vходу velikosti  $n$  algoritem A izvrši  $20n^2$  operacij, algoritem B pa  $10000n$  operacij. Za katere vrednosti  $n$  je algoritem A bolj učinkovit? [10 točk]

## Naloga 10

Podan je algoritem:

```
i = 0
a = 2
b = 0
while i < n do
  a = a · i
  b = b + 1
  i = i + 2
  j = 256
  while j > 0 do
    j = j - 1
  end while
  k = n
  while k > 2 do
    k = k/2
  end while
end while
```

- (a) Predpostavimo, da je  $n$  pozitivno parno število. Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]
- (b) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvedbi algoritma? [5 točk]
- (c) Spremenite čim manj vrstic algoritma, da bo njegova časovna zahtevnost linearna. [5 točk]

## Naloga 11

Podan je algoritem:

```
a = 0
b = 1
i = n/2
L = [ ]
while i > 0 do
  j = n
  while j < 0 do
    L.append(a)
    j = j/3
  end while
  i = i - 1
end while
while i < n do
  L.append(b)
  j = 0
  while j < n do
    j = j + 1
  end while
  a = a + i
  b = b * a
  i = i + 1
end while
```

- (a) Predpostavimo, da je  $n$  pozitivno celo število. Kakšen je razred časovne zahtevnosti tega algoritma glede na osnovne aritmetične operacije? Operacija  $“/”$  predstavlja celoštevilsko deljenje. [7 točk]
- (b) Kakšen je razred prostorske zahtevnosti tega algoritma?  $L = [ ]$  pomeni, da je  $L$  prazen seznam, funkcija  $L.append(b)$  pa doda element  $b$  na konec seznama  $L$ . [6 točk]
- (c) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvedbi algoritma? [4 točke]
- (d) Spremenite eno vrstico algoritma, v kateri se ne pojavi spremenljivka  $b$ , tako da bo vrednost spremenljivke  $b$  po izvedbi algoritma  $b = 1 * 2 * \dots * n = n!$ . [3 točke]

## Naloga 12

Podana imamo dva algoritma:

A) 

```
i = 0
while i < n do
  j = 100
  i = i + 1
  while j > 0 do
    j = j - 1
  end while
end while
```

B) 

```
i = 0
while i < n do
  j = n
  i = i + 2
  while j ≥ 0 do
    j = j - 3
  end while
end while
```

- (a) Kakšna je časovna zahtevnost algoritmov glede na osnovne aritmetične operacije? [8 točk]  
A) \_\_\_\_\_ B) \_\_\_\_\_
- (b) Poiščite tak  $n$ , kjer se spremeni izbira učinkovitejšega algoritma. Navedite za katere vrednosti  $n$  je bolj učinkovit kateri algoritem. [7 točk]

# REŠITVE

## Rešitev - Naloga 1

Preberi  $k$ .  
Preberi  $N_1, N_2, \dots, N_k$ .  
Nastavi  $i$  na 1.  
Dokler ( $i < k/2$ )  
     $\text{temp} = N_i$   
     $N_i = N_{k+1-i}$   
     $N_i = \text{temp}$   
Ustavi se.

## Rešitev - Naloga 2

Nekaj možnih rešitev:

1. Najprej uredi seznam, potem vrni število z indeks  $n//2$
2. Za vsako število v seznamu, preštej koliko je večjih in manjših števil
3. Sortiraj seznam dokler ne najdeš  $n//2 + 1$  najmanjših števil in zadnjo število na seznamu bo medijana

## Rešitev - Naloga 3

Rekurzivno:

```
gcd(a, b)
    if (b == 0)
        return a;
    return gcd(b, a mod b)
```

Iterativno:

```
gcd(a, b)
    while (b != 0) {
        t := b;
        b := a mod b;
        a := t;
    }
    return a;
```

#### Rešitev - Naloga 4

- (a) Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]

$$O(n \cdot \log n)$$

- (b) Koliko operacij množenja (v odvisnosti od  $n$ ) se zgodi v enem izvajanju algoritma? [5 točk]

$$2n + \frac{n \cdot \log n}{2}$$

- (c) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvajanju algoritma? [5 točk]

$$a = 3^{(2^n)}$$

$$b = 0$$

#### Rešitev - Naloga 5

- Kakšna je časovna zahtevnost zgornjega programa v odvisnosti od  $N$ ?  
Ker pride do preliva se ustavi in zahtevnost je  $O(1)$ .
- Kolikokrat se izvrši ukaz COMPARE Y?  
256-krat.
- Kakšna je desetiška vrednost pomnilniške lokacije  $N$  ob koncu izvajanja programa?  
0

#### Rešitev - Naloga 6

- (a) Predpostavimo, da je  $n$  pozitivno celo število. Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]

$$O(n \log n)$$

- (b) Kakšna je vrednost spremenljivk  $a$ ,  $b$  in  $c$  (v odvisnosti od  $n$ ) po izvajanju algoritma? [5 točk]

$$a = 1, b = 2 + 9n, c = 3 + 9n$$

## Rešitev - Naloga 7

- (a) Predpostavimo, da je  $n$  pozitivno celo število. Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije?

[5 točk]

$O(\log^2 n)$

- (b) Kolikokrat se izvede operacija seštevanja? [5 točk]

$(2\log(n) + 1)$ –krat

- (c) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvajanju algoritma? [5 točk]

$a = 1, b = 6\log(n)$

## Rešitev - Naloga 8

REŠITEV:

Treba je prešteti št. operacij in rešit enostavno kvadratno enačbo:

A:  $c \cdot n \cdot n / c + 100000 = n^2 + 100000$  operacij

B:  $1100 \cdot n$  operacij

$$n^2 + 100000 = 1100 \cdot n$$

$$(n-100) \cdot (n-1000) = 0$$

$$n_1 = 100, n_2 = 1000$$

Za  $n < 100$  je boljši B, za  $n = 100$  sta enaka, na  $100 < n < 1000$  je boljši A, pri  $n = 1000$  sta spet enaka, nato pa je za  $n > 1000$  B vedno boljši.

n= 2 A: 100004 B: 2200

n= 10 A: 100100 B: 11000

n= 50 A: 102500 B: 55000

n= 90 A: 108100 B: 99000

n= 100 A: 110000 B: 110000

n= 200 A: 140000 B: 220000

n= 500 A: 350000 B: 550000

n= 900 A: 910000 B: 990000

n= 1000 A: 1100000 B: 1100000

n= 1100 A: 1310000 B: 1210000

n= 1200 A: 1540000 B: 1320000

## Rešitev - Naloga 9

REŠITEV:

```
A      B
20 n^2 = 10000 n
20 n = 10000    // lahko delimo z n, ker privzamemo, da n>0
n = 500
```

A je bolj učinkovit za  $0 < n < 500$ .

## Rešitev - Naloga 10

- (a) Predpostavimo, da je  $n$  pozitivno parno število. Kakšna je časovna zahtevnost tega algoritma glede na osnovne aritmetične operacije? [5 točk]

$O(n \log n)$

- (b) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izva-  
janju algoritma? [5 točk]

$a = 0, b = n/2 + 1$

- (c) Spremenite čim manj vrstic algoritma, da bo njegova časovna za-  
htevnost linearna. [5 točk]

npr  $k = 8$

## Rešitev - Naloga 11

- (a) Predpostavimo, da je  $n$  pozitivno celo število. Kakšen je razred časovne zahtevnosti tega algoritma glede na osnovne aritmetične op-  
eracije? Operacija  $"/$  predstavlja celoštevilsko deljenje. [7 točk]

$\Theta(n^2)$

- (b) Kakšen je razred prostorske zahtevnosti tega algoritma?  $L = [ ]$  pomeni, da je  $L$  prazen seznam, funkcija  $L.append(b)$  pa doda ele-  
ment  $b$  na konec seznama  $L$ . [6 točk]

$\Theta(n)$

- (c) Kakšna je vrednost spremenljivk  $a$  in  $b$  (v odvisnosti od  $n$ ) po izvedbi  
algoritma? [4 točke]

$a = 0 + 1 + 2 + \dots + n - 1 = n(n - 1)/2, b = 0$

- (d) Spremenite eno vrstico algoritma, v kateri se ne pojavi spremenljivka  
 $b$ , tako da bo vrednost spremenljivke  $b$  po izvedbi algoritma  $b =$   
 $1 * 2 * \dots * n = n!$ . [3 točke]

$a = a + i \rightarrow a = i + 1$  ali  $a = a + 1$



## Rešitev - Naloga 12

- (a) Kakšna je časovna zahtevnost algoritmov glede na osnovne aritmetične operacije? [8 točk]

A)  $O(n)$

B)  $O(n^2)$

- (b) Poiščite tak  $n$ , kjer se spremeni izbira učinkovitejšega algoritma. Navedite za katere vrednosti  $n$  je bolj učinkovit kateri algoritem.

[7 točk]  $100 \cdot n = \frac{n}{2} \cdot \frac{n}{3}$

$$100 \cdot n = \frac{n^2}{6}$$

$$100 = \frac{n}{6}$$

$$600 = n$$

Do  $n = 600$  je učinkovitejši B, po tem pa A.