

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ

10. srpnja 2009.

**Upute:** Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni šalabahter. Kalkulatori, mobiteli, razne neslužbene tablice, papiri i sl., **nisu** dozvoljeni!

Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent.

U svim zadacima zabranjeno je korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Rezultati i uvid u kolokvije: **ponedjeljak, 13. srpnja 2009. u 12 sati.**

## ZADATAK 1

1

(15 + 10 + 10 = 35 bodova.)

- (a) Napišite funkciju koja prima tri argumenta: pokazivač na datoteku već otvorenu za pisanje, te cijele brojeve  $x$  i  $y$ . Funkcija treba vratiti vrijednost  $f(x, y)$  definiranu s

$$f(x, y) = \begin{cases} xy, & \text{ako } x + y \text{ sadrži znamenku } 7, \\ f(x + \text{inv}(y - 2x), x) + 7, & \text{ako je } x > \text{inv}(y), \\ f(\text{ds}(x + y), \text{ds}(x - y + 1)), & \text{inače,} \end{cases}$$

gdje  $\text{inv}(z)$  vraća broj dobiven zapisivanjem  $|z|$  unatrag, a  $\text{ds}(z)$  vraća sumu znamenaka broja  $z$  zapisanog u bazi 7. Dodatno, funkcija treba u **svakom** koraku zapisati  $x$ ,  $y$  i rezultat u datoteku u CSV formatu, koristeći znak `;` kao separator.

- (b) Napišite program koji kao argumente komandne linije prima cijele brojeve  $x$  i  $y$ . Program treba otvoriti tekstualnu datoteku s imenom `"recursion.txt"` i ispisati koliko je  $f(x, y)$ . Pazite da program ispravno otvori i zatvori datoteku.

Za dodatnih 10 bodova, ispis pojedinih koraka u datoteku i vraćanje vrijednosti funkcije riješite pomoću **jedne** parametrizirane preprocesorske direktive (makro-naredbe).

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 2

10. srpnja 2009.

(10 + 20 + 15 = 45 bodova.)

Svaki element  $m \times n$  matrice znakova je znak '.' (točka), ili znak '\*' (zvjezdica). Zvjezdica predstavlja zid. Pri tome se sve zvjezdice mogu poredati u ovakav niz: prva zvjezdica se nalazi u gornjem desnom kutu matrice, druga zvjezdica se nalazi 1 mjesto lijevo, ili 1 mjesto ispod prve, treća zvjezdica je 1 mjesto lijevo, ili 1 mjesto ispod druge, itd. (vidi sliku). Zid spaja gornji desni s donjim lijevom kutem matrice.

```

.....**
.....**
.****..
**.....
*.....

```

- (a) Napišite funkciju `broj` koja prima neku takvu matricu, te putem varijabilnih parametara vraća 2 broja: koliko se znakova '.' nalazi iznad zida i koliko se znakova '.' nalazi ispod zida. Na slici je iznad zida 10 točaka, a ispod 14.
- (b) Napišite funkciju `zid` koja prima neku takvu matricu, te stvara i vraća novi string koji se sastoji od znakova 'D' i 'L'. String opisuje nastanak niza zvjezdica kako je navedeno gore: znak 'D' predstavlja pomak prema dolje ("ispod"), a 'L' pomak lijevo. Za matricu na slici treba vratiti string "LDLDDLLDLD", jer je druga zvjezdica lijevo od prve, treća je ispod druge, četvrta je lijevo od treće, itd. Za string treba alocirati točno onoliko memorije koliko je najmanje potrebno.
- (c) Napišite glavni program u kojem ćete učitati dimenzije matrice i samu matricu, te pozvati funkcije `broj` i `zid`, i ispisati njihove povratne vrijednosti. Nemojte uvoditi ograničenja na dimenzije matrice.

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 3

10. srpnja 2009.

(10 + 5 + 25 = 40 bodova.)

Za opis zadataka koje treba napraviti definiran je tip `Zadatak` na sljedeći način

```
typedef struct _Zadatak {
    int sat, trajanje;
    char opis[32];
    struct _Zadatak *next;
} Zadatak;
```

gdje `sat` označava početak zadatka, a `trajanje` se mjeri u satima od početka.

(a) Napišite funkciju s prototipom

```
Zadatak *ubaci(Zadatak *first, Zadatak *novi)
```

koja ubacuje novi zadatak u vezanu listu zadataka s početkom `first`, tako da nova lista ostane sortirana uzlazno po satu u kojem zadaci započinju. Pretpostavite da je polazna lista već uzlazno soritrana po satu, od početka prema kraju liste.

(b) Definirajte tip `Rokovnik` kao matricu s 4 retka i 7 stupaca, čiji su elementi pokazivači na početke vezanih listi zadataka. Recite matrice predstavljaju tjedne, a stupci dane unutar tjedna, počev od ponedjeljka.

U nastavku zadatka, pretpostavlja se da su sve liste u rokovniku sortirane uzlazno po satu.

(c) Napišite funkciju koja prima `Rokovnik`, redni broj dana u mjesecu (između 1 i 28) i sat u danu. Funkcija treba obrisati sva zaduženja za taj dan koja počinju najkasnije u zadani sat, a traju najviše 3 sata.

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ

10. srpnja 2009.

**Upute:** Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni šalabahter. Kalkulatori, mobiteli, razne neslužbene tablice, papiri i sl., **nisu** dozvoljeni!

Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent.

U svim zadacima zabranjeno je korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Rezultati i uvid u kolokvije: **ponedjeljak, 13. srpnja 2009. u 12 sati.**

## ZADATAK 1

1

(15 + 10 + 10 = 35 bodova.)

- (a) Napišite funkciju koja prima tri argumenta: pokazivač na datoteku već otvorenu za pisanje, te cijele brojeve  $x$  i  $y$ . Funkcija treba vratiti vrijednost  $f(x, y)$  definiranu s

$$f(x, y) = \begin{cases} x + y, & \text{ako } xy \text{ sadrži znamenku } 3, \\ f(\text{inv}(x + y), y + 1) + 3, & \text{ako je } \text{inv}(x) < y, \\ f(\text{md}(x + y + 1, 17), \text{md}(x - y, 19)), & \text{inače,} \end{cases}$$

gdje  $\text{inv}(z)$  vraća broj dobiven zapisivanjem  $|z|$  unatrag, a  $\text{md}(z, b)$  vraća najveću znamenku broja  $z$  zapisanog u bazi  $b$ . Dodatno, funkcija treba u **svakom** koraku zapisati  $x$ ,  $y$  i rezultat u datoteku u CSV formatu, koristeći znak `#` kao separator.

- (b) Napišite program koji kao argumente komandne linije prima cijele brojeve  $x$  i  $y$ . Program treba otvoriti tekstualnu datoteku s imenom `"recursion.txt"` i ispisati koliko je  $f(x, y)$ . Pazite da program ispravno otvori i zatvori datoteku.

Za dodatnih 10 bodova, ispis pojedinih koraka u datoteku i vraćanje vrijednosti funkcije riješite pomoću **jedne** parametrizirane preprocesorske direktive (makro-naredbe).

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 2

10. srpnja 2009.

(10 + 20 + 15 = 45 bodova.)

Svaki element  $m \times n$  matrice znakova je znak '.' (točka), ili znak '#' (povisilica). Povisilica predstavlja zid. Pri tome se sve povisilice mogu poredati u ovakav niz: prva povisilica se nalazi u donjem lijevom kutu matrice, druga povisilica se nalazi 1 mjesto desno, ili 1 mjesto iznad prve, treća povisilica je 1 mjesto desno, ili 1 mjesto iznad druge, itd. (vidi sliku). Zid spaja donji lijevi s gornjim desnim kutem matrice.

```

.....#
....###
...##..
.###...
##.....

```

- (a) Napišite funkciju `broj` koja prima neku takvu matricu, te putem varijabilnih parametara vraća 2 broja: koliko se znakova '.' nalazi lijevo od zida i koliko se znakova '.' nalazi desno od zida. Na slici je lijevo od zida 14 točaka, a desno 10.
- (b) Napišite funkciju `zid` koja prima neku takvu matricu, te stvara i vraća novi string koji se sastoji od znakova 'D' i 'G'. String opisuje nastanak niza povisilica kako je navedeno gore: znak 'D' predstavlja pomak prema desno, a 'G' pomak prema gore ("iznad"). Za matricu na slici treba vratiti string "DGDDGDGDDG", jer je druga povisilica desno od prve, treća je iznad druge, četvrta je desno od treće, itd. Za string treba alocirati točno onoliko memorije koliko je najmanje potrebno.
- (c) Napišite glavni program u kojem ćete učitati dimenzije matrice i samu matricu, te pozvati funkcije `broj` i `zid`, i ispisati njihove povratne vrijednosti. Nemojte uvoditi ograničenja na dimenzije matrice.

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 3

10. srpnja 2009.

(10 + 5 + 25 = 40 bodova.)

Za opis zadataka koje treba napraviti definiran je tip `Zadatak` na sljedeći način

```
typedef struct _Zadatak {
    int sat, trajanje;
    char opis[32];
    struct _Zadatak *next;
} Zadatak;
```

gdje `sat` označava početak zadatka, a `trajanje` se mjeri u satima od početka.

- (a) Napišite funkciju koja prima pokazivač na početak vezane liste zadataka, sortirane uzlazno po satu, i vraća pokazivač na prvi zadatak iz liste koji završava nakon što sljedeći zadatak započinje. Ako takvog zadatka nema, funkcija vraća `NULL`.
- (b) Definirajte tip `Rokovnik` kao matricu s 4 retka i 7 stupaca, čiji su elementi pokazivači na početke vezanih listi zadataka. Reciproci matrice predstavljaju tjedne, a stupci dane unutar tjedna, počev od ponedjeljka.

U nastavku zadatka, pretpostavlja se da su sve liste u rokovniku sortirane uzlazno po satu.

- (c) Napišite funkciju koja prima `Rokovnik`, redni broj dana u mjesecu (između 1 i 28) i sat u danu. Funkcija treba obrisati sva zaduženja za taj dan koja počinju u zadani sat ili kasnije, a traju barem 2 sata.

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ

10. srpnja 2009.

**Upute:** Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni šalabahter. Kalkulatori, mobiteli, razne neslužbene tablice, papiri i sl., **nisu** dozvoljeni!

Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent.

U svim zadacima zabranjeno je korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Rezultati i uvid u kolokvije: **ponedjeljak, 13. srpnja 2009. u 12 sati.**

## ZADATAK 1

1

(15 + 10 + 10 = 35 bodova.)

- (a) Napišite funkciju koja prima tri argumenta: pokazivač na datoteku već otvorenu za pisanje, te cijele brojeve  $x$  i  $y$ . Funkcija treba vratiti vrijednost  $f(x, y)$  definiranu s

$$f(x, y) = \begin{cases} x - y, & \text{ako } xy \text{ sadrži znamenku } 5, \\ f(\text{inv}(3y) + 1, x), & \text{ako je } x < \text{inv}(y), \\ f(\text{md}(x + y + 1, 13), \text{md}(x - y, 23)), & \text{inače,} \end{cases}$$

gdje  $\text{inv}(z)$  vraća broj dobiven zapisivanjem  $|z|$  unatrag, a  $\text{md}(z, b)$  vraća najveću znamenku broja  $z$  zapisanog u bazi  $b$ . Dodatno, funkcija treba u **svakom** koraku zapisati  $x$ ,  $y$  i rezultat u datoteku u CSV formatu, koristeći znak `$` kao separator.

- (b) Napišite program koji kao argumente komandne linije prima cijele brojeve  $x$  i  $y$ . Program treba otvoriti tekstualnu datoteku s imenom `"recursion.txt"` i ispisati koliko je  $f(x, y)$ . Pazite da program ispravno otvori i zatvori datoteku.

Za dodatnih 10 bodova, ispis pojedinih koraka u datoteku i vraćanje vrijednosti funkcije riješite pomoću **jedne** parametrizirane preprocesorske direktive (makro-naredbe).

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 2

10. srpnja 2009.

(10 + 20 + 15 = 45 bodova.)

Svaki element  $m \times n$  matrice znakova je znak '.' (točka), ili znak '\*' (zvjezdica). Zvjezdica predstavlja zid. Pri tome se sve zvjezdice mogu poredati u ovakav niz: prva zvjezdica se nalazi u donjem desnom kutu matrice, druga zvjezdica se nalazi 1 mjesto lijevo, ili 1 mjesto iznad prve, treća zvjezdica je 1 mjesto lijevo, ili 1 mjesto iznad druge, itd. (vidi sliku). Zid spaja donji desni s gornjim lijevim kutem matrice.

```
*.....
**.....
.****..
....**
.....**
```

- (a) Napišite funkciju `broj` koja prima neku takvu matricu, te putem varijabilnih parametara vraća 2 broja: koliko se znakova '.' nalazi lijevo od zida i koliko se znakova '.' nalazi desno od zida. Na slici je lijevo od zida 10 točaka, a desno 14.
- (b) Napišite funkciju `zid` koja prima neku takvu matricu, te stvara i vraća novi string koji se sastoji od znakova 'G' i 'L'. String opisuje nastanak niza zvjezdica kako je navedeno gore: znak 'G' predstavlja pomak prema gore ("iznad"), a 'L' pomak lijevo. Za matricu na slici treba vratiti string "LGLGLLGLG", jer je druga zvjezdica lijevo od prve, treća je iznad druge, četvrta je lijevo od treće, itd. Za string treba alocirati točno onoliko memorije koliko je najmanje potrebno.
- (c) Napišite glavni program u kojem ćete učitati dimenzije matrice i samu matricu, te pozvati funkcije `broj` i `zid`, i ispisati njihove povratne vrijednosti. Nemojte uvoditi ograničenja na dimenzije matrice.



## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 3

10. srpnja 2009.

(10 + 5 + 25 = 40 bodova.)

Za opis zadataka koje treba napraviti definiran je tip `Zadatak` na sljedeći način

```
typedef struct _Zadatak {
    int sat, trajanje;
    char opis[32];
    struct _Zadatak *next;
} Zadatak;
```

gdje `sat` označava početak zadatka, a `trajanje` se mjeri u satima od početka.

- (a) Napišite funkciju koja prima pokazivač na početak vezane liste zadataka i vraća 1, ako je lista sortirana uzlazno po satu, a inače vraća 0.
- (b) Definirajte tip `Rokovnik` kao matricu s 4 retka i 7 stupaca, čiji su elementi pokazivači na početke vezanih listi zadataka. Redovi matrice predstavljaju tjedne, a stupci dane unutar tjedna, počev od ponedjeljka.

U nastavku zadatka, pretpostavlja se da su sve liste u rokovniku sortirane uzlazno po satu.

- (c) Napišite funkciju koja prima `Rokovnik`, dan u tjednu (između 1 i 7) i dnevnu normu sati. Funkcija treba pronaći, za sva četiri tjedna posebno, prvi zadatak u danu čije bi izvršenje premašilo zadanu normu, i obrisati sve zadatke u danu, od tog zadatka nadalje.

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ

10. srpnja 2009.

**Upute:** Na kolokviju je dozvoljeno koristiti samo pribor za pisanje i brisanje, te službeni šalabahter. Kalkulatori, mobiteli, razne neslužbene tablice, papiri i sl., **nisu** dozvoljeni!

Sva rješenja napišite isključivo na papire sa zadacima, jer jedino njih predajete. Ne zaboravite se **potpisati** na svim papirima! Skice smijete raditi i na drugim papirima koje će vam dati dežurni asistent.

U svim zadacima zabranjeno je korištenje dodatnih nizova i standardne matematičke biblioteke (zaglavlje `math.h`), osim ako je u zadatku drugačije navedeno.

Rezultati i uvid u kolokvije: **ponedjeljak, 13. srpnja 2009. u 12 sati.**

## ZADATAK 1

1

(15 + 10 + 10 = 35 bodova.)

- (a) Napišite funkciju koja prima tri argumenta: pokazivač na datoteku već otvorenu za pisanje, te cijele brojeve  $x$  i  $y$ . Funkcija treba vratiti vrijednost  $f(x, y)$  definiranu s

$$f(x, y) = \begin{cases} x - y, & \text{ako } xy \text{ sadrži znamenku } 2, \\ f(x - 2, \text{inv}(2x - y)), & \text{ako je } \text{inv}(x) \geq y, \\ f(\text{ds}(x - y), \text{ds}(x + y)), & \text{inače,} \end{cases}$$

gdje  $\text{inv}(z)$  vraća broj dobiven zapisivanjem  $|z|$  unatrag, a  $\text{ds}(z)$  vraća sumu znamenaka broja  $z$  zapisanog u bazi 17. Dodatno, funkcija treba u **svakom** koraku zapisati  $x$ ,  $y$  i rezultat u datoteku u CSV formatu, koristeći znak `|` kao separator.

- (b) Napišite program koji kao argumente komandne linije prima cijele brojeve  $x$  i  $y$ . Program treba otvoriti tekstualnu datoteku s imenom `"recursion.txt"` i ispisati koliko je  $f(x, y)$ . Pazite da program ispravno otvori i zatvori datoteku.

Za dodatnih 10 bodova, ispis pojedinih koraka u datoteku i vraćanje vrijednosti funkcije riješite pomoću **jedne** parametrizirane preprocesorske direktive (makro-naredbe).

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 2

10. srpnja 2009.

(10 + 20 + 15 = 45 bodova.)

Svaki element  $m \times n$  matrice znakova je znak '.' (točka), ili znak '#' (povisilica). Povisilica predstavlja zid. Pri tome se sve povisilice mogu poredati u ovakav niz: prva povisilica se nalazi u gornjem lijevom kutu matrice, druga povisilica se nalazi 1 mjesto desno, ili 1 mjesto ispod prve, treća povisilica je 1 mjesto desno, ili 1 mjesto ispod druge, itd. (vidi sliku). Zid spaja gornji lijevi s donjim desnim kutem matrice.

```
##.....  
.###...  
...##..  
....###  
.....#
```

- (a) Napišite funkciju `broj` koja prima neku takvu matricu, te putem varijabilnih parametara vraća 2 broja: koliko se znakova '.' nalazi iznad zida i koliko se znakova '.' nalazi ispod zida. Na slici je iznad zida 10 točaka, a ispod 14.
- (b) Napišite funkciju `zid` koja prima neku takvu matricu, te stvara i vraća novi string koji se sastoji od znakova 'D' i 'R'. String opisuje nastanak niza povisilica kako je navedeno gore: znak 'D' predstavlja pomak prema dolje ("ispod"), a 'R' pomak prema desno. Za matricu na slici treba vratiti string "RDRDRDRRD", jer je druga povisilica desno od prve, treća je ispod druge, četvrta je desno od treće, itd. Za string treba alocirati točno onoliko memorije koliko je najmanje potrebno.
- (c) Napišite glavni program u kojem ćete učitati dimenzije matrice i samu matricu, te pozvati funkcije `broj` i `zid`, i ispisati njihove povratne vrijednosti. Nemojte uvoditi ograničenja na dimenzije matrice.

## PROGRAMIRANJE 2 — POPRAVNI KOLOKVIJ — ZADATAK 3

10. srpnja 2009.

(10 + 5 + 25 = 40 bodova.)

Za opis zadataka koje treba napraviti definiran je tip `Zadatak` na sljedeći način

```
typedef struct _Zadatak {
    int sat, trajanje;
    char opis[32];
    struct _Zadatak *next;
} Zadatak;
```

gdje `sat` označava početak zadatka, a `trajanje` se mjeri u satima od početka.

- (a) Napišite funkciju koja prima pokazivač na početak vezane liste zadataka i vraća ukupno trajanje zadataka u listi, ako ono ne prelazi 24 sata, a inače vraća  $-1$ .
- (b) Definirajte tip `Rokovnik` kao matricu s 4 retka i 7 stupaca, čiji su elementi pokazivači na početke vezanih listi zadataka. Recipro matrice predstavljaju tjedne, a stupci dane unutar tjedna, počev od ponedjeljka.

U nastavku zadatka, pretpostavlja se da su sve liste u rokovniku sortirane uzlazno po satu.

- (c) Napišite funkciju koja prima `Rokovnik`, redni broj tjedna (između 1 i 4) i tjednu normu sati. Funkcija treba pronaći, počev od ponedjeljka u zadanom tjednu nadalje, prvi zadatak čije bi izvršenje premašilo zadanu normu i obrisati sve zadatke u tjednu, od tog zadatka nadalje.