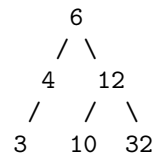


Zadatak 1 Napisati funkciju `int f8(cvor* drvo, int nivo)` koja vraća broj elemenata drveta koji se nalaze na nivou `nivo`. Testirati pozivom u `main-u`.

Primeri:

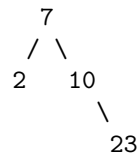
Primer 1:



nivo: 3

rez: 3

Primer 2:



nivo: 3

rez: 1

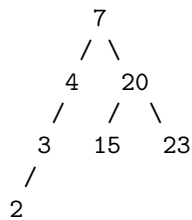
Primer 3:



nivo: 3

rez: 0

Primer 4:

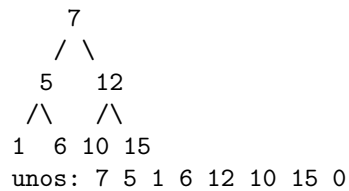


nivo: 2

rez: 2

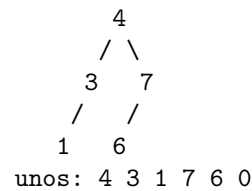
Zadatak 2 Napisati funkciju `int f5(cvor* drvo)` koja vraća broj elemenata stabla drvo koji su jednaki razlici svojih sinova.

Primer 1:



izlaz: 2 (to su cvorovi 7 i 5)

Primer 2:



izlaz: 1 (to je cvor 4)

Primer 3:



Primer 4:



```

    3  12
   /
-1
unos: 6 3 0 12 0

izlaz: 0

```

```

    5  23
   / \ /
  4  9 12
   /
  1
unos: 10 5 4 9 1 5 23 12 0

izlaz: 1

```

Zadatak 3 Napisati funkciju koja iz datoteke `karakter_i.txt` ucitava karaktere i smešta ih u binarno pretraživacko drvo. Uz svaki karakter čuvati i broj pojavljivanja karaktera u datoteci. Ispisati na standardni izlaz karakter koji se pojavljuje najveći broj puta u datoteci.

`karakter_i.txt`: Danas je lep dan.

`karakter_i.txt`: 78665555512

izlaz: a

izlaz: 5

```

karakter_i.txt: U Datoteci
                je probni test
                za Zadatke sa ispita.

```

izlaz: a (izlaz može biti i: t, ' ' (blanko znak))

Zadatak 4 Funkcija vrši rekurzivnu rotaciju drveta oko svih čvorova, dakle dobija se odraz prvobitnog drveta u ogledalu.

```

    7
   / \
  2  10
 / \ / \
4  5 9 17
   /
  12
   \
  13

```

se transformiše u:

```

    7
   / \
  10  2
 / \ / \
17  9 5 4
   \
  12
   /
  13

```

Zadatak 5 Sa standardnog ulaza se zadaje aritmetički izraz zapisan u prefiksnoj notaciji koji može sadržati samo operatore + i * i jednocifrene brojeve. Napisati program koji formira stablo od tako dobijenog izraza, ispisuje izraz u infiksnoj notaciji (sa zagradama) i izračunava vrednost izraza.

Zadatak 6 Napisati program koji iz datoteke čije se ime zadaje kao prvi argument komandne linije čita reči i formira binarno uređeno stablo (uređeno leksikografski) koje će u jednom čvoru sadržati reč koja se pojavljuje i broj pojavljivanja te reči u datoteci. Ispisati reč koja se pojavljuje najčešće u datoteci.

Zadatak 7 Data je struktura:

```
typedef struct _cvor {
    int vrednost;
    struct _cvor *levo, *desno;
} cvor;
```

Napisati funkciju `int br_cvorova(cvor *koren)`, koja računa broj čvorova u stablu koji su veći od svih svojih potomaka.

Zadatak 8 Dva binarna stabla su identična ako su ista po strukturi i sadržaju, odnosno oba korena imaju isti sadržaj i njihova odgovarajuća podstabla su identična. Napisati funkciju koja proverava da li su dva binarna stabla identična.

Zadatak 9 Napisati program koji implementira telefonski imenik. U fajlu, koji se da zadaje kao argument komandne linije, nalazi se niz linija oblika: `Ime Prezime ***/***-****` tj. za svaku osobu je naveden broj telefona (pretpostaviti da je datoteka ispravna). Kada se program pokrene, korisnik treba da unese ime i prezime, a program treba da pronade broj telefona te osobe, ili da obavesti korisnika da se osoba ne nalazi u imeniku. Ovaj postupak treba ponavljati dokle god korisnik ne unese komandu za kraj rada (EOF). Informacije o brojevima telefona uneti u mapu koja je implementirana preko binarnog stabla pretrage.

Zadatak 10 Napisati funkciju `int ravnomerno_izbalansirano(Cvor *stablo)` koja proverava da li je stablo ravnomerno izbalansirano. Stablo je ravnomerno izbalansirano ako za svaki čvor važi da je pozitivna razlika između dubine levog i dubine desnog podstabla najviše 1. Testirati funkciju pozivom u main-u, stablo se učitava sve dok se ne unese 0. Ukoliko jeste izbalansirano ispisati 1, a u suprotnom 0.

Primer 1:

```
      10
     /  \
    7    20
   / \  / \
  5  8 15 23
```

Primer 2:

```
      10
     /  \
    7    11
   /
  5
```

Primer 3:

```
ulaz: 0
izlaz: 1
```

Primer 4:

```
      10
     /  \
    7    11
   / \
  5  8
```

ulaz: 10 7 20 5 8 15 23 0
izlaz: 1

/
4

ulaz: 10 7 11 5 4 0
izlaz: 0

ulaz:
izlaz:

Zadatak 11 Napisati funkciju `int prebroj(cvor* drvo)` koja vraća broj elemenata stabla drvo koji su iste parnosti kao oba svoja sina (sva tri parna ili neparna). Ukoliko je cvor list ili ima samo jednog sina ne ulazi u zbir.

Zadatak 12 Dat je polinom reprezentovan listom, tako da čvorovi liste sadrže koeficijente i stepene polinoma. Dati polinom je nesređen, tj. može sadržati više članova sa istim stepenom. Napisati funkciju `cvor* f4(cvor* lista)` koja sređuje polinom tako da:

- da se svaki stepen pojavljuje najviše jedanput
- da su svi koeficijenti koji su prisutni u listi različiti od 0

U `main`-u se unosi broj elemenata liste, a potom i elementi liste (svaki element liste se dodaje na kraj). Ispisati na standardni izlaz polinom koja je dobijena sređivanjem ulazne liste.

Primer 1:

4
1 1
2 3
-1 1
1 3

3*x^3

Primer 2:

10
-1 2
3 3
1 2
-3 3
7 1
2 5
3 1
-1 5
-1 5
4 4

10*x^1 + 4*x^4

Primer 3:

4
3 1
2 3
5 1
3 3

8*x^1 + 5*x^3

Primer 4:

5
2 3
1 2
4 5
-2 3
2 2

3*x^2 + 4*x^5