

PROGRAMIRANJE II (05.07.2016.)

- 1 (25 bodova) Napisati program u kojem treba iz binarne datoteke, čiji je naziv prvi argument komandne linije, pročitati binarno upisane nepakovane četvorocifrene BCD podatke. Prva četiri bajta (neoznačeni cijeli broj) u datoteci predstavljaju broj upisanih BCD podataka. U pročitanoj nizu podataka, pronaći najmanji i najveći, a zatim sortirati samo one podatke u nizu koji se nalaze između najmanjeg i najvećeg podatka. Za sortiranje podataka, u rastućem poretku, koristeći *insert-sort* algoritam, potrebno je definisati i koristiti funkciju čiji je prototip:

```
void insert_sort(int *niz, unsigned n);
```

Rezultujući niz BCD podataka upisati binarno u datoteku čiji je naziv drugi argument komandne linije, ako je naveden. Ako drugi argument komandne linije nije naveden podatke upisati u datoteku čiji je naziv DEFAULT.DAT.

- 2 (25 bodova) Potrebno je implementirati jednostruko povezanu listu koja sadrži statističke podatke o nastupima jednog tenisera na turnirima. Neka je definisan tip:

```
typedef struct cvor
{
    char *sifra;
    int broj_bodova;
    struct cvor *sljedeći;
} CVOR;
```

kojim se reprezentuje čvor jednostruko povezane liste. Čvor liste sadrži šifru turnira (dinamički string) i broj osvojenih bodova na tom turniru.

Definisati rekurzivnu funkciju koja vraća ukupan broj bodova jednog tenisera na svim turnirima. Funkcija ima sljedeći prototip:

```
int broj_bodova(CVOR *glava);
```

Definisati funkciju koja dodaje novi čvor u listu tako da je poredak elemenata u listi uvijek u opadajućem redoslijedu po broju osvojenih bodova, a čiji je prototip:

```
void dodaj(CVOR **pglava, char *s, int br);
```

pri čemu *s* predstavlja šifru, a *br* broj osvojenih bodova.

Definisati funkciju koja ispisuje podatke o *n* najboljih nastupa na turnirima (ispisati šifru turnira i broj bodova), a čiji je prototip:

```
void top_n(CVOR *glava, int n);
```

U glavnom programu učitati podatke (sa standardnog ulaza) o nastupima jednog tenisera i formirati listu, koristeći funkciju *dodaj*, a zatim ispisati ukupan broj bodova, koristeći funkciju *broj_bodova*, kao i podatke o pet najboljih nastupa koristeći funkciju *top_n*.

Napomena: Nije dozvoljena upotreba globalnih, niti statičkih promjenljivih.

- 3 (25 bodova) Neka je definisan tip:

```
typedef struct graf
{
    int n; // broj cvorova
    void *info[MAX]; // inf.sadržaj cvorova
    int ms[MAX][MAX]; // matrica susjednosti
} GRAF;
```

kojim se reprezentuje usmjeren graf.

Napisati funkciju koja za zadati graf vraća stepen grafa, odnosno najveći stepen nekog od čvorova u grafu. Prototip funkcije je:

```
int stepen(GRAF *g);
```

Napisati funkciju sa promjenljivim brojem argumenata koja, koristeći funkciju *stepen*, pronalazi i vraća adresu grafa koji sadrži čvor najvećeg stepena. Funkcija prihvata broj grafova *n*, te *n* pokazivača na grafove. Prototip funkcije je:

```
GRAF *f(int n, ...);
```

- 4 (25 bodova) Neka su definisani tipovi:

```
typedef struct cvor_s
{
    int info;
    struct cvor_s *lijevi, *desni;
} CVOR_S; // cvor binarnog stabla
```

```
typedef struct cvor_r
{
    CVOR_S *cs;
    struct cvor_r *sljedeći;
} CVOR_R; // cvor reda
```

i funkcije za dodavanje i brisanje iz reda:

```
void dodaj(CVOR_R **pf, CVOR_R **pr, CVOR_S *cs);
int brisi(CVOR_R **pf, CVOR_R **pr, CVOR_S **cs);
```

Funkcija *dodaj* omogućava dodavanje adrese čvora binarnog stabla (*cs*) u odgovarajući red. Prvi i drugi argument funkcije predstavljaju adrese odgovarajućih pokazivača na prvi (*front*) i posljednji (*rear*) element reda.

Funkcija *brisi* omogućava dohvaćanje (preko parametra *cs*) i brisanje prvog elementa iz reda, a vraća informaciju o tome da li je element uspješno obrisao iz reda (element nije uspješno obrisao iz reda ako je red prazan). Prvi i drugi argument funkcije predstavljaju adrese odgovarajućih pokazivača na prvi (*front*) i posljednji (*rear*) element reda.

Definisati funkciju koja, na osnovu primljenih podataka, formira kompletno (ili skoro kompletno) binarno stablo. Funkcija vraća korijen formiranog stabla. Prototip funkcije je:

```
CVOR_S *kompletno_stablo(int *niz, int n);
```