

PROGRAMIRANJE II (06.07.2017. - K2)

❶ (10 bodova) Neka je definisan tip:

```
typedef struct cvor{
    double k; // koeficijent clana
    unsigned s; // stepen clana
    struct cvor *sljedeci;
} CVOR;
```

kojim se reprezentuje čvor jednostruko povezane liste. Svaki čvor liste reprezentuje jedan član realnog polinoma $P(x)=a_nx^n+a_{n-1}x^{n-1}+...+a_1x+a_0$. Pri tome, broj čvorova liste jednak je broju članova polinoma za koje je koeficijent $\neq 0$. Takođe, u listi ne postoje dva člana istog stepena. Lista je uređena u opadajućem redoslijedu po vrijednosti stepena člana.

Neka je definisana funkcija koja dodaje novi čvor na kraj liste:

```
void dodaj_kraj(CVOR **pglava, double k, unsigned s);
```

Definisati funkciju (dozvoljeno je korištenje samo funkcije *dodaj_kraj*) koja sabira dva polinoma (predstavljena odgovarajućim listama) i vraća rezultujući polinom (predstavljen odgovarajućom listom). Prototip funkcije je:

```
CVOR* saberi(CVOR *p1, CVOR *p2);
```

Na primjer, polinom $P(x)=5x^4-3x^2+7$ se predstavlja na sljedeći način:



❷ (10 bodova) Neka je definisan tip:

```
typedef struct buffer{
    void *start, *front, *rear;
    int size, capacity;
} BUFFER;
```

koji predstavlja kružni bafer podataka proizvoljnog tipa, pri čemu je: *start* – adresa početka bafera u memoriji, *front* – adresa prvog podatka u baferu, *rear* – adresa posljednjeg podatka u baferu, *size* – veličina jednog podatka u baferu, *capacity* – kapacitet bafera tj. maksimalan broj podataka u baferu. Pri tome, ako je bafer prazan, onda je $front = NULL \wedge rear = NULL$, te ako bafer ima tačno jedan element, onda je $front \neq NULL \wedge front = rear$.

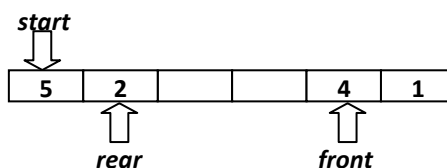
Napisati funkciju koja dodaje novi podatak u bafer bez prepisivanja, a vraća informaciju o tome da li je podatak uspješno dodan u bafer (0 = podatak nije dodan u bafer, 1 = podatak je dodan u bafer). Prototip funkcije je:

```
int add_norewrite(BUFFER *b, const void *data);
```

Napisati funkciju koja dodaje novi podatak u bafer sa prepisivanjem, a vraća informaciju o tome da li je prilikom dodavanja podatka došlo do prepisivanja (0 = nije bilo prepisivanja, 1 = bilo je prepisivanja). Prototip funkcije je:

```
int add_rewrite(BUFFER *b, const void *data);
```

Primjer kružnog bafera kapaciteta šest elemenata sa elementima 4, 1, 5 i 2:



❸ (15 bodova) Neka je definisan tip:

```
typedef struct cvor{
    int info, nivo;
    struct cvor *lijevi, *desni;
} CVOR;
```

Napisati funkciju koja ima sljedeći prototip:

```
void dopuni_nivo (CVOR *korijen, int nivo);
```

Funkcija treba da dopuni nivo binarnog stabla, pri čemu je prvi argument funkcije korijen stabla a drugi argument nivo koji je potrebno dopuniti. Potrebno je dodati maksimalan broj čvorova u dati nivo, tako da svi čvorovi iz prethodnog nivoa imaju stepen 2. Informacioni sadržaj svakog novog čvora treba da bude jednak informacionom sadržaju njegovog roditelja.

❹ (15 bodova) Napisati program koji omogućava iterativni obilazak grafa po dubini (u implementaciji nije dozvoljen rekurzivni poziv) pri čemu je potrebno ispisivati informacioni sadržaj čvorova grafa. Potrebno je koristiti matricnu reprezentaciju grafa. Informacioni sadržaj čvora je cijeli broj. Prilikom iterativnog obilaska obavezno je koristiti stek. Funkcije za rad sa stekom su definisane u biblioteci stek.h. Prototipovi funkcija za rad sa stekom su sljedeći:

```
void push(CVOR **tos, int *info);
int pop(CVOR **tos, int *info);
```