

Algoritmi i strukture podataka - pismeni deo ispita, septembar 2009.

1. Traži se prva pojava uzorka **marama** u tekstu **marijasemaloodmaramadaimadauči**. Izračunati brojeve pomeranja uzorka do konačnog odgovora (prikazati tabelu koja se koristi za algoritam KMP).
2. a) Odrediti izgled implicitno predstavljenog hipa koji se dobija umetanjem redom brojeva 5, 7, 9, 11, 8, 3, 9, 10 polazeći od praznog hipa. Prikazati izgradnju hipa tabelom, čiji svaki red odgovara jednoj promeni hipa. Zatim odrediti izgled hipa dobijenog uklanjanjem najvećeg elementa. Smatrajte da je hip organizovan tako da je ključ svakog čvora veći ili jednak od ključeva njegovih sinova.
b) Konstruisati algoritam za formiranje hipa koji sadrži sve elemente dva hipa veličine n i m . Hipovi su predstavljeni eksplicitno (svaki čvor ima pokazivače na svoja dva sina). Vremenska složenost treba da bude u najgorem slučaju $O(\log(m + n))$. Obavezno je obrazložiti vremensku složenost konstruisanog algoritma.
3. Dat je usmereni graf $G=(V,E)$ i njegov čvor v . Konstruisati algoritam složenosti $O(|E| + |V|)$ koji ispituje da li G jeste korensko stablo sa korenom v . Smatrati da su u korenskom stablu grane usmerene od korena ka listovima.
4. Konstruisati algoritam koji za datu sumu P i dati niz v koji sadrži vrednosti novčanica pronalazi sva rešenja za rasitnjavanje sume P . Pretpostaviti da svake novčanice ima proizvoljno mnogo.

1.

kmpNext:

```
0 1 2 3 4 5 6
-1 0 0 0 -1 0 2
```

POKUŠAJ 1:

marijasemaloodmaramadaimadauči

MARa..

Pomeri se za: 3 (3-kmpNext[3])

POKUŠAJ 2:

marijasemaloodmaramadaimadauči

m.....

Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 3:

marijasemaloodmaramadaimadauči

m.....

Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 4:

marijasemaloodmaramadaimadauči

m.....

Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 5:

marijasemaloodmaramadaimadauči

m.....

Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 6:

marijasemaloodmaramadaimadauči

m.....

Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 7:

marijasemaloodmaramadaimadauči

MAR..

Pomeri se za: 2 (2-kmpNext[2])

POKUŠAJ 8:
marijasemaloodmaramadaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 9:
marijasemaloodmaramadaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 10:
marijasemaloodmaramadaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 11:
marijasemaloodmaramadaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 12:
marijasemaloodmaramadaimadauči
MARAMA
Pomeri se za: 4 (6-kmpNext[6])

POKUŠAJ 13:
marijasemaloodMARAMAdaimadauči
..r...
Pomeri se za: 2 (2-kmpNext[2])

POKUŠAJ 14:
marijasemaloodMARAMAdaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 15:
marijasemaloodMARAMAdaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 16:
marijasemaloodMARAMAdaimadauči
m.....
Pomeri se za: 1 (0-kmpNext[0])

POKUŠAJ 17:
marijasemaloodMARAMAdaimadauči
MAr...
Pomeri se za: 2 (2-kmpNext[2])

marijasemaloodMARAMAdaimadauči

POKUŠAJA: 17
POREĐENJA: 29

2. a) videti sličan zadatak sa vežbi
- b) videti identičan zadatak sa vežbi
- 3.

Korensko stablo jeste usmereno stablo koje poseduje posebno izdvojen čvor koji se zove koren, a sve grane su usmerene od korena.

Na zadati usmeren graf $G=(V,E)$ treba primeniti algoritam za obilazak stabla smatrajući pri tome da je čvor v njegov koren i brojeći čvorove kroz koje se prolazi.

Ukoliko pri tom obilasku, broj čvorova kroz koje se prošlo bude veći od $|V|$, graf G ima ciklus(e), te nije stablo; ukoliko se obilazak završi sa brojem čvorova kroz koje se prošlo manjim od $|V|$, graf G nije povezan, te nije stablo; inače graf G jeste stablo.

```
Algoritam obilazak1(G,v)
```

```
ulaz: G=(V,E) (usmeren graf), v (cvor grafa G)
izlaz: broj ( brojac cvorova kroz koje se proslo)

{
markiraj v;
broj=broj+1 ;
ako je broj manji ili jednak od broja cvorova grafa G onda
    za sve grane (v,w) do
        if markiran(w) return -1;
        else
            obilazak1(G,w);
return broj;
}
```

```
Algoritam provera1(G,v)
```

```
ulaz: G=(V,E) (usmeren graf), v(cvor grafa G)
izlaz: odgovor na pitanje da li je graf G sa korenom v stablo
{
    broj:=0 ;
    obilazak1(G,v);
    ako je broj manji od broja cvorova grafa G onda
        write('graf G nije povezan');
    ako je broj  veci od broja cvorova grafa G onda write('graf G ima
ciklus(e) ')
    inace write('graf G je stablo');
}
```

4.

```
#include <stdio.h>
```

```
int P; /* novcani iznos koji treba razbiti */
```

```
int n; /* broj razlicitih novcanica */
```

```
int v[10]; /* vrednosti novcanica/apoena */
```

```
int x[10]; /* kolicina pojedinih apoena */
```

```
void pisi()
```

```
{ int i;
```

```
printf("\nRazmena: \n");
```

```
for(i=0; i<n; i++) printf("%d dinara:%d puta ", v[i],x[i]);
```

```
printf("\n");
```

```
}
```

```
void razmeni(int k, int P)
```

```
{
```

```
//k= redni broj apoena
//P = tekuca suma za rasiitnjavaenje
int i;
if(k>=n)
{ if(P==0) pisi();}
else
for(i=0; i<=P/v[k];i++)
{ x[k]=i;
razmeni(k+1, P -i*v[k]);
}
}

main()
{
int i;
printf("Unesite sumu i broj novcanica\n");
scanf("%d%d",&P,&n);
printf("\nUnesite vrednost za %d novcanica\n",n);
for(i=0; i<n;i++) scanf("%d",&v[i]);
razmeni(0,P);
}
```