Primene računara, septembar 2009.

1. Neka n ljudi čeka u redu da kupi karte za predstavu, pri čemu je $\mathbf{t_i}$ vreme koje je i-tom kupcu potrebno da kupi kartu. Ako se po dvoje suseda u redu udruži da kupi karte - na primer k-ti i k+1-vi kupac, onda vreme potrebno da oni kupe karte je $\mathbf{p_k}$, k=1..n-1. Udruživanjem kupaca može da se ubrza kupovina karata. Ulazni podaci su broj kupaca **n** i nizovi **t** i **p**. Konstruisati algoritam koji određuje takav način udruživanja da bude minimalno ukupno vreme potrebno da svih **n** kupaca kupi kartu.

2. a) Odrediti izgled implicitno predstavljenog hipa koji se dobija umetanjem redom brojeva 7, 3, 5, 11, 8, 6, 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. Traži se prva pojava uzorka **ananas** u tekstu **anaibananasunasi**. Izračunati brojeve pomeranja uzorka do konačnog odgovora (prikazati tabelu koja se koristi za algoritam KMP).

4. Zadat je regularni graf (graf čiji svi čvorovi imaju isti stepen) i prirodan broj k. Dokazati NP-kompletnost problema koji utvrđuje da li ovaj graf sadrži kliku veličine k.

REŠENJE:
1.
Neka je niz **a** takav da **a[k]** je ušteda u vremenu kupovine karata koja nastaje udruživanjem **k**-tog i **k+1**-vog kupca, tj:
a[k]=t[k]+t[k+1]-p[k], k=1..n-1
Jasno da članovi niza **a** u opštem slučaju su i pozitivni i negativni celi brojevi.
Dalje, **k**-ti čovek se ne može istovremeno udružiti sa **k+1**-vim i **k-1**-vim čovekom,
tj. odatle sledi da u nizu ne mogu istovremeno i a[k-1] i a[k] biti uštede,
tj.potrebno je pronaći podniz niza **a (**a[1]...a[n-1]) koji ima najveću sumu u kojoj ne učestvuju susedni članovi

Neka je a niz od m=n-1 celih brojeva.
Neka je R=R(A) jedan podniz iz formulacije zadatka koji odgovara nizu a.
Neka je A_k niz koji se sastoji od prvih k elemenata niza a.

Podniz niza A_0 je prazan niz, dok je podniz niza A_1 element a[1] ako je a[1]>0.

1. Ako element a[m] ne pripada podnizu R=R(A), onda je R
podniz iz formulacije zadatka koji odgovara nizu A_{m-1}.
2. Ako element a[m] pripada podnizu R=R(A), onda je R \ a[m]
podniz iz formulacije zadatka koji odgovara nizu A_{m-2}.

Dakle, za i=2..m, važi da $R(A_i)$ je ili $R(A_{i-1})$ ili $R(A_{i-2})$* {a[i]}, tj.
uzima se onaj od podnizova koji ima veci zbir.

Neka je niz s takav da:
s(i) je suma podniza $R(A_i)$.
Iz gore izlozenog, jasno je da:
s(0)=0,
s(1)=max(0, a[1]),
s(i)=max{ s(i-1), s(i-2) + a[i] }, i=2..m

Iz niza s će se ispisati rešenje R.

```
Algoritam MSN (a,m)
ulaz: a, m   /* niz a duzine m, BSO clanovi su a[1]...a[m] */
izlaz: podniz niza a cija suma je maksimalana,
gde podniz ne sadrzi uzastopne elemente iz a
{
  s[0]=0;      /* s[j] = suma elemenata podniza b koji je resenja zadatka za podniz A(j) */
               /* s[0]=0, jer suma praznog niza je 0, po dogovoru */
if (a[1] > 0)  s[1]=a[1];
           else s[1]=0;
```

```
for (i=2;i <= m; i++)  /*uzimamo da indeks prvog clana niza je 1, drugog je 2,...*/

     if  (s[i-2] + a[i]  > s[i-1])
             s[i]= s[i-2] + a[i];
      else s[i]= s[i-1];

MSN_ispis (m);
}


procedure  MSN_ispis (m)
ulaz: m
izlaz: ispis clanova podniza koji sadrzi nesusedne clanove niza,
         ali tako da je suma podniza maksimalna
{
if (m > 0)
   if (s[m] = = s[m-1)  MSN_ispis(m-1);
   else
      {  MSN_ispis(m-2);
         ispisati  a[m];   /*jer je clan niza b */
      }

}
```

2.

a)

umetni 7

1

umetni 3

7	3

umetni 5

7	3	5

umetni 11

7	3	5	11

umetni 8

7	3	5	11	8
11	8	5	3	7

umetni 6

11	8	5	3	7	6
11	8	6	3	7	5

umetni 9

11	8	6	3	7	5	9
11	8	9	3	7	5	6

umetni 10

11	8	9	3	7	5	6	10
11	8	9	10	7	5	6	3
11	10	9	8	7	5	6	3

UKLANJANJE ELEMENTA 11

3	10	9	8	7	5	6
10	3	9	8	7	5	6
10	8	9	8	7	5	6

b) videti zadatak sa vežbi
3.

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

pokušaj  1:
anaibananasunasi
ANAn..
Pomeri za: 3 (3-kmpNext[3])

pokušaj  2:
anaibananasunasi
   a.....
Pomeri za: 1 (0-kmpNext[0])

pokušaj  3:
anaibananasunasi
    a.....
Pomeri za: 1 (0-kmpNext[0])

pokušaj  4:
anaibananasunasi
     ANANAS
Pomeri za: 6 (6-kmpNext[6])


anaibANANASunasi
POKUŠAJA  4
    Broj poređenja karaktera:  12
```

4.
Može se svesti opšti klika problem na klika problem za regularne grafove.
Neka je G=(V,E) proizvoljan graf i k proizvoljan broj.
Kontruiše se pravilan graf H takav da graf G ima kliku veličine veću ili jednaku k ako i samo ako H ima kliku veličine veću ili jednaku k.
Neka je n broj čvorova grafa G .
Neka je d maksimalni stepen grafa G ako je taj broj paran, inače maksimalni stepen grafa G plus jedan (tj. vrednost d je uvek parna).
Za svaki čvor v grafa G stepena d(v)<d dodaje se d-d(v) novih čvorova i povezuju se sa čvorom v . Ukupan broj novih čvorova je
$\sum_{i=1..n}(d - d(v_i)) = dn - \sum_{i=1..n} d(v_i) = dn - 2|E|$

Ovaj broj je paran, jer je broj d paran.
Cilj nam je da povežemo nove čvorove tako da svi imaju stepen d.

Nove klike za sada nisu napravljene (zašto?). Skup novih čvorova se, dalje, deli na dva dela sa jednakim brojem čvorova (zašto je to moguce?). Svaki čvor iz jedne grupe povežimo sa tačno d-1 čvorova iz druge.

Tako dobijen graf je regularan i on ima kliku veličine veću ili jednaku k ako i samo ako graf G ima kliku veličine veću ili jednaku k .

Time je klika problem sveden na klika problem za regularnene grafove. Kako je klika problem NP-kompletan problem, sledi da je i klika problem za regularne grafove NP-kompletan.