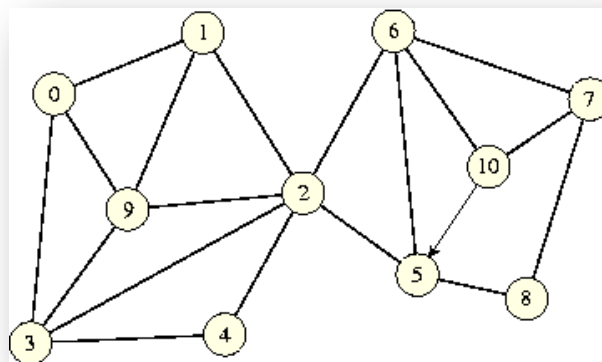


Program za ispitivanje cvora grafa,artikulacione tacke

Definicija:

U povezanom grafu G cvor predstavlja artikulacionu tacku ako brisanjem tog cvora, I svih grana koje polaze iz njega, graf postaje nepovezan.



Na ovom primeru sa slike,artikulaciona tacka je cvor "2",jer njegovim brisanjem dati graf postaje nepovezan.

Kako program radi:

Unosimo n-broj cvorova u grafu, zatim predstavimo graf pomocu matrice susedstva. Unosimo k-cvor koji zelimo da ispitamo da li predstavlja artikulacionu tacku, brisemo k-tu vrstu I k-tu kolonu iz martice susedstva da bi dobili novu matricu koja je $(n-1) \times (n-1)$ dimenzije, i tu matricu saljemo f-ji koja ce da ispita da li je taj graf dobijen izbacivanjem jednog cvora povezan ili ne. U slucaju ako graf i dalje ostaje povezan, cvor k nije artikulaciona tacka,a ako je graf nepovezan cvor k je artikulaciona tacka.

Kod:

```
#include<stdio.h>
#include<stdlib.h>
#define BROJCLANOVA 100

void mnozi(int a[BROJCLANOVA][BROJCLANOVA],int b[BROJCLANOVA][BROJCLANOVA],int
c[][BROJCLANOVA], int n)
{
    int i,j,k,z;
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            {
                c[i][j]=0;
                for(k=0;k<n;k++)c[i][j]+=a[i][k]*b[k][j];
            }

    for(j=1;j<n;j++)
        {
            z=0;
            for(i=0;i<j;i++)
                z+=c[i][j];
            if(z>0)c[n][j]=1;else c[n][j]=0;
        }
}

int main()
{
    int matrica[BROJCLANOVA][BROJCLANOVA],nova_matrica[BROJCLANOVA][BROJCLANOVA],
i,j,n,p,k,t,z=0;
    int pom[BROJCLANOVA][BROJCLANOVA],niz[BROJCLANOVA],pom1[BROJCLANOVA][BROJCLANOVA];
    //unos matrice
    printf("\nUnesite broj cvorova n:\n");
    scanf("%d",&n);
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            {
                printf("\nUnesite %d. clan %d. vrste matrice susedstva:\t",j+1,i+1);
                scanf("%d",&matrica[i][j]);
            }
    printf("Unesite koji cvor zelite da ispitajte \n");
    scanf("%d",&k);
    k=k-1;
    p=n-1;
    //kopiramo staru matricu u novi
```

```

for(i=0;i<n;i++)
  for(j=0;j<n;j++)
    nova_matrica[i][j]=matrica[i][j];
//prvo izbacimo kolonu
for(i=0;i<n;i++)
  for(j=k;j<p;j++)
    nova_matrica[i][j]=nova_matrica[i][j+1];
// zatim izbacimo vrstu
for(j=0;j<n;j++)
  for(i=k;i<p;i++)
    nova_matrica[i][j]=nova_matrica[i+1][j];

//stampanje matrice susedstva grafa
printf("MATRICA SUSEDSTVA GRAFA\n\n");
for(i=0;i<n;i++)
{
  for(j=0;j<n;j++)
    printf("%d\t",matrica[i][j]);
  printf("\n");
}
printf("=====\n\n");
//stampanje matrice susedstva bez k-te kolone i vrste
printf("MATRICA SUSEDSTVA BEZ %d ELEMENTA\n\n",k+1);
for(i=0;i<p;i++)
{
  for(j=0;j<p;j++)
    printf("%d\t",nova_matrica[i][j]);
  printf("\n");
}
printf("=====\n\n");

//provera da li je graf povezan
n=p;
for(i=0;i<n;i++)for(j=0;j<n;j++){if(i==j)pom1[i][j]=1;else pom1[i][j]=0;}
for(i=0;i<n-1;i++)
{
  mnozi(nova_matrica,pom1,pom,n);
  for(j=0;j<n;j++)for(t=0;t<n;t++)pom1[j][t]=pom[j][t];
  for(t=0;t<n;t++)niz[t]+=pom[n][t];
}
z=0;
for(j=1;j<n;j++){if(niz[j]>0)niz[j]=1;else niz[j]=0;z+=niz[j];}
if(z==n-1)printf("\nTACKA %d NIJE ARTIKULACIONA!\n",k+1);
else printf("\nTACKA %d JE ARTIKULACIONA!\n",k+1);
system("pause"); }

```