

Işleri ýerine ýetirmek (BOI1997)

Bizde birnäçe ýerine ýetirmeli işlerimiz bar. Nähili hem bolsa, iki sany iş biri-birine baglanşykly. İş 2 iş 1-e bagly, egerde iş 2-ni ýerine ýetirmek üçin iş 1-iň gutarmagyna garaşmaly bolsa.

Käbir işler duş gelýär, ýagny olary parallel (köp işi birden ýerine ýetirmek) şekilde ýerine ýetirip bolýar, wagty tygşytlamak üçin. İşleriň sany we baglylyklar berlen. Hemme işleri ýerine ýetirmek üçin gerek bolan iň gysga wagty tapmaly. İşleri ýerine ýetirmek üçin isledigiňçe ýerine ýetiriji enjam (ÝÝE) bar. Sizden soralyan ikinji zat bolsa, iň gysga wagtda ýerine ýetirmek üçin gerek ÝÝE-leriň sany. Bir işi ýerine ýetirmek üçin 1 sekunt wagt gerek. Her işler 1-den başlap N-e çenli nomerlenen. ($1 \leq N \leq 200$)

GIRIŞ

Siziň programmaňyz girişi "INPUT.TXT" faýlyndan aşakda görkezilşi ýaly okamaly. Birinji setirde iki sany bitin san berlen, N we M. N – ýerine ýetirmeli işleriň sany we M – baglanşyklaryň sany. Ikinji setirden başlap, faýlyň soňuna çenli, baglanşyklar barada maglumatlar berlen. Meselem, bir setirde "2 3" bar bolsa, 3-nji işe başlamadan öň 2-nji işi gutarmaly diýmek bolýar. Giriş hemme wagt dogrydyr we hemme wagt çözüwi bardyr.

ÇYKYS

Siziň programmaňyz jogaplary "OUTPUT.TXT" atly faýla ýazmaly. Ol faýlda bir setir bolmaly. Setirde iki sany bitin san, birinjisi işi gutarmak üçin gerek iň gysga wagt, ikinji san bolsa iň gysga wagtda işi gutarmak üçin gerek ÝÝE-niň sany.

MESELEM

INPUT .TXT	OUTPUT .TXT
6 6	4 2
1 4	
2 5	
3 6	
4 6	
4 5	
5 6	

Her test üçin 3 sekunt berilýär.

Çözgüdi

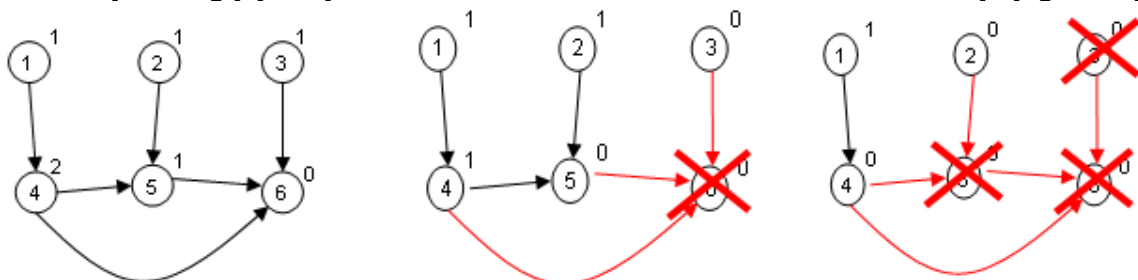
Bu soragda size gönükdirilen siklsiz graf (DAG – Directed Acyclic Graph) berilýär. ÝSG-lerden bolsa hemme wagt topological sorting tapyp bolýar. Biz algoritmi birazajyk üýtgedeli. Goý $DEG[i]$ -da i -nji dügümden daşyna çykýan gönükdirilen baglansyklaryň sany bolsun. Eger $DEG[i]=0$ bolsa, onda i -nji işi yerine ýetirmek üçin hiç zada garaşmaly däl. Şu wagt $DEG[i]=0$ bolan hemme i -leri parallel şekilde yerine ýetirip bolar. Ol işleri yerine ýetirýän we ol dügümleri grafdan çykarýän. Graf boş bolýança, bu işi gaýtalamaly.

Graf boş bolýança

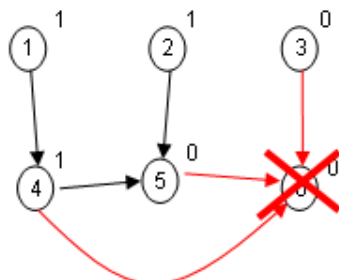
```
{
  P=0;
  For i=1 to n do
    If(deg[i]==0 && not removed[i])
    {
      P++;
      NOW[P]=I;
    }
  IF (P > PP) PP=P; //ÝÝE sany

  For i=1 to P do
  {
    Removed[NOW[i]]=true;
    <DEG täzele>
  }
  T=T+1;
}
WRITE (T, PP)
```

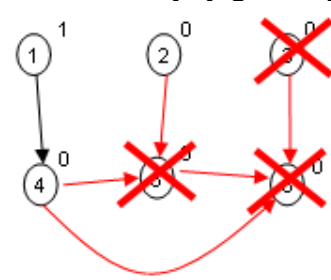
Indi bolsa bu algoritmanyň işleýşini göreli. Dügümleriň içindäki sanlar iş nomeri, gönükdirijiler baglylyklary we dügümleriň üstündäki sanlar DEG bahasyny görkezýär.



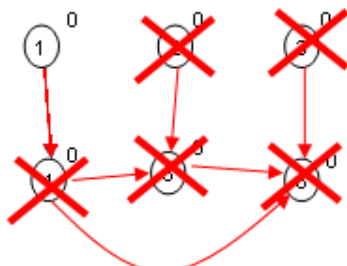
T=0 PP=0



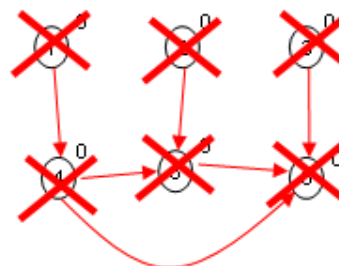
T=1 PP=1



T=2 PP=2



T=3 PP=2



T=4 PP=2

Ine Programmamyz:

```
/*
    5th Balkan Olympiad in Informatics, 1997
    Day 1 Problem 2 Task Execution
    (C) Copyright Atamyrat Hezretgulyev, 2000-2004
*/
#include<stdio.h>
#include<string.h>

#define infile "task.in"
#define outfile "task.out"
#define maxn 110

int deg[maxn];
int a[maxn][maxn];

int n,t,p; // n-number of vertices t-time p-number of processors

int main()
{
    int m,i,u,v,deg0,j;
    int deg0l[maxn], removed[maxn];

    FILE *fin=fopen(infile,"r");
    fscanf(fin,"%d%d",&n,&m);
    memset(a,0,sizeof(a));
    memset(deg,0,sizeof(deg));
    memset(removed,0,sizeof(removed));
    for(i=1; i<=m; i++)
    {
        fscanf(fin,"%d%d",&u,&v);
        deg[u]++; a[v][0]++; a[v][a[v][0]]=u;
    }
    fclose(fin);

    t=0; p=0; printf("\n\n");
    for(;;)
    {
        deg0=0;
        for(i=1; i<=n; i++) //list vertices with 0 out-degree
            if(deg[i]==0 && !removed[i])
            {
                deg0++;
                deg0l[deg0]=i;
            }
        if(deg0>p) p=deg0; // update number of proc-s.
        if(!deg0) break; // all tasks executed.
        // remove 0 degree vertices
        for(i=1; i<=deg0; i++)
        {
            printf("%d ",deg0l[i]);
            removed[deg0l[i]]=1;
            for(j=1; j<=a[deg0l[i]][0]; j++)
                deg[a[deg0l[i]][j]]--;
        }
        printf("\n");
        t++;
    }

    FILE *fout=fopen(outfile,"w");
    fprintf(fout,"%d %d\n",t,p);
}
```

```
fclose(fout);  
return 0;  
}
```

Atamyrat Hezretgulyev
atamyrat_hezretgulyev@yahoo.com