**МИНОБРНАУКИ РОССИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №3.3

дисциплина: Дискретная математика

тема: «**Фактор множества**»

|  |  |
| --- | --- |
|  | Выполнил: ст. группы ВТ-22  Макаров Даниил  Проверил: Рязанов Ю.Д. |

Белгород

2018

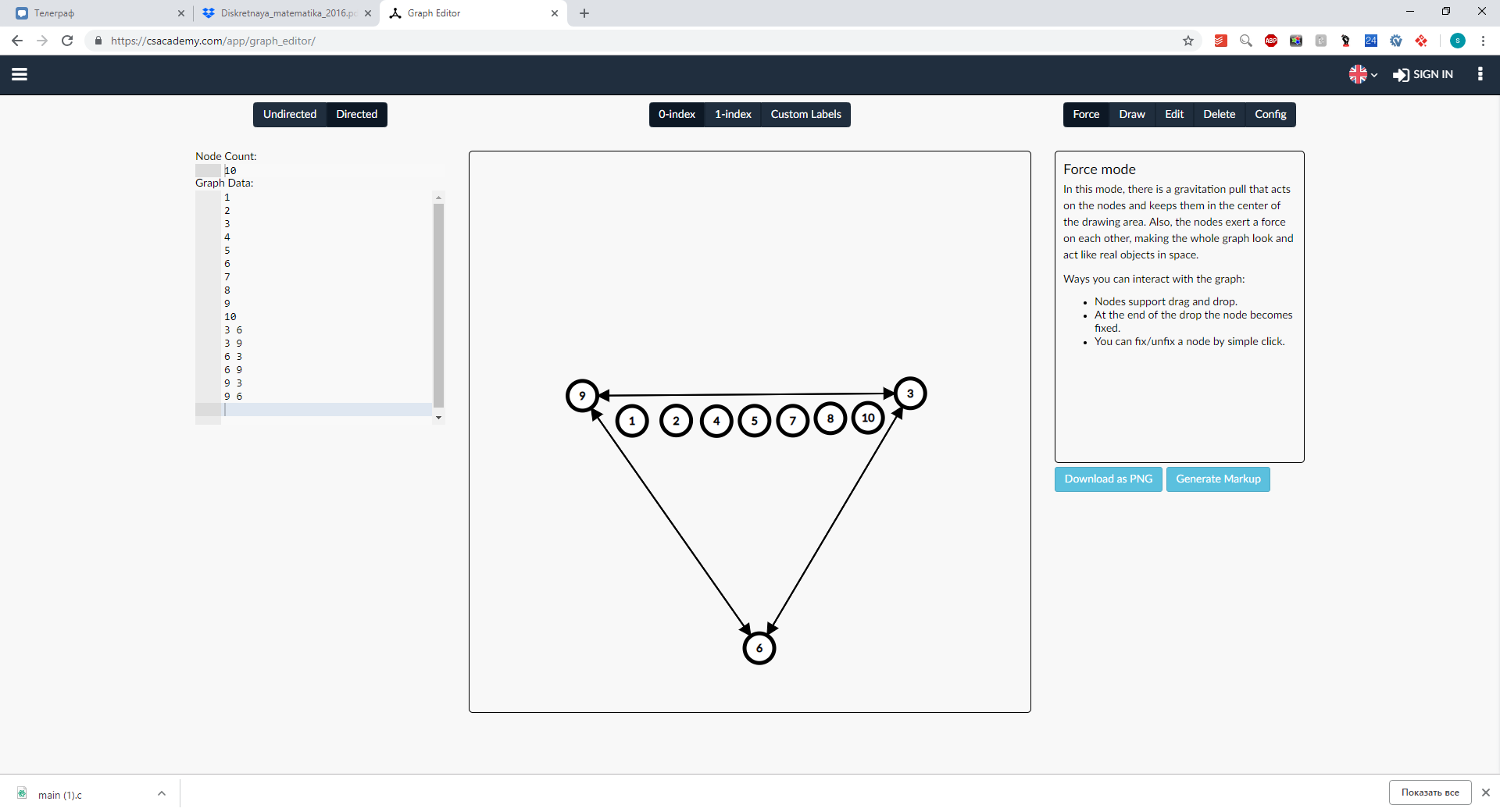
**Цель занятия**: научиться программно формировать фактормножества для заданного отношения эквивалентности и находить отношение эквивалентности для заданного разбиения.

**Вариант 7**

А={(x,y) | xN и yN и x<11 и y<11 и (x и y кратно 3 или x=y)}

**Задания**

1. Отношение на множестве {1,2,3,4,5,6,7,8,9,10} (табл. 3.3) представить графом и характеристической функцией в матричной форме. Найти разбиение Ф, определяемое заданным отношением эквивалентности.



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

1. Программно реализовать алгоритм построения отношения эквивалентности R по разбиению S множества М.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <time.h>

#define N 11

int form\_R (int \*\*a){

int x,y;

for (x=1;x<N;x++)

for (y=1;y<N;y++)

if ((x%3==0 && y%3==0) || (x%5==0 && y%5==0) || (x==y))

a[x][y]=1;

return 0;

}

void output (int \*\*a){

int x,y;

for (x=1;x<N;x++){

for (y=1;y<N;y++)

printf ("%d ",a[x][y]);

printf("\n");

}

}

void clear (int \*\*a){

int x,y;

for (x=0;x<N;x++)

for (y=0;y<N;y++)

a[x][y]=0;

}

void clear\_arr (int \*a){

int x;

for (x=0;x<N;x++)

a[x]=0;

}

void cl\_eq (int \*\*R, int x, int \*clX, int y){

int i,j=1;

for (i=1;i<N;i++)

if (R[x][i])

clX[i]=y;

}

void razb (int \*m, int \*\*r, int \*s){

int a[N],i,c,fl=N,j=1;

for (i=0;i<N;i++)

a[i]=i;

for (i=0;i<N;i++)

s[i]=0;

i=1;

while (pust(a)&&fl){

while (!a[i]&&i<N)

i++;

cl\_eq(r,a[i],s,j);

fl-=del (a,s);

j++;

}

}

int del (int \*a, int \*s){

int i,j=0;

for (i=1;i<N;i++)

if (s[i){

a[i]=a[i]&&(!s[i]);

j++;

}

return j;

}

void postr (int \*s, int \*\*r){

int a[N], g=0,x,y,i=1,j=1;

while (j<N){

while (i<N){

if (s[i]==j){

g++;

a[g]=i;

}

i++;

}

if (g==1)

r[a[g]][a[g]]=1;

if (g>1)

for (x=1;x<=g;x++)

for (y=1;y<=g;y++)

r[a[x]][a[y]]=1;

clear\_arr (a);

j++;

i=1;

g=1;

}

}

int pust (int \*a){

int i;

for (i=1;i<N;i++)

if(a[i])

return (i<N);

}

int main(){

int i, M[N],S[N];

for (i=1;i<N;i++)

M[i]=i;

int \*\*r;

r=(int \*\*)malloc(N\*sizeof(int\*));

for (i=0;i<N;i++)

r[i]=(int \*)malloc(N\*sizeof(int));

form\_R(r); printf ("\nRel - \n"); output (r);

razb (M,r,S);

i=1;

printf ("\nPartition F \n");

while (i<N)

printf ("%d ",S[i++]);

clear(r);

postr(S,r);

output (r);

}