**МИНОБРНАУКИ РОССИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕНЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.Шухова»**

**(БГТУ им. В.Г.Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа № 4.2

По дисциплине: Дискретная математика

Тема: «Циклы»

Выполнил: Богатырёв В.

Белгород 2018

**Цель занятия:** изучить разновидности циклов в графах, научиться генерировать случайные графы, определять их принадлежность к множеству эйлеровых и гамильтоновых графов, находить все эйлеровы и гамильтоновы циклы в графах.

**Задания**

1. Разработать и реализовать алгоритм генерации случайного графа,

содержащего n вершин и m ребер.

2. Написать программу, которая:

а) в течение десяти секунд генерирует случайные графы, содержащие n вершин и m ребер;

б) для каждого полученного графа определяет, является ли он эйлеровым или гамильтоновым;

в) подсчитывает общее количество сгенерированных графов и количество графов каждого типа.

Результат работы программы представить в виде таблицы

(табл. 4.6).

**Результат работы программы**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| количество вершин | количество ребер | количество графов | | |
| эйлеровых | гамильтоновых | всех |
| n | n |  |  |  |
| n | n+1 |  |  |  |
| n | n+2 |  |  |  |
|  |  |  |  |  |
| n |  |  |  |  |

3. Выполнить программу при n = 8, 9, 10 и сделать выводы.

4. Привести пример диаграммы графа, который является эйлеровым, но не гамильтоновым. Найти в нем все эйлеровы циклы.

5. Привести пример диаграммы графа, который является гамильтоновым, но не эйлеровым. Найти в нем все гамильтоновы циклы.

6. Привести пример диаграммы графа, который является эйлеровым и гамильтоновым. Найти в нем все эйлеровы и гамильтоновы циклы.

7. Привести пример диаграммы графа, который не является ни эйлеровым, ни гамильтоновым.

**Задание 1.**

void GenRandGraph(int \*\*\*G, int n, int m)

{

srand(clock());

int x = 0, y = 0, k = 0;

\*G = calloc(n,sizeof(int\*));

for (int i = 0; i < n; i++)

(\*G)[i] = calloc(n,sizeof(int));

while (k < m)

{

x = rand() % n;

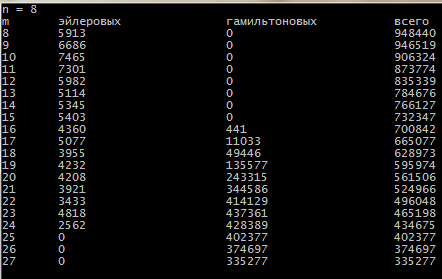
y = rand() % n;

if (!(\*G)[x][y] && x != y)

k += (\*G)[x][y] = (\*G)[y][x] = 1;

}

}

**Задание 2.**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <windows.h>

void freemem(int \*\*a, int n)

{

for (int i = 0; i < n; i++)

free(a[i]);

free(a);

}

int \*\*GenRandGraph(int n, int m)

{

int \*\*G;

srand(clock());

int x = 0, y = 0, k = 0;

G = calloc(n,sizeof(int\*));

for (int i = 0; i < n; i++)

G[i] = calloc(n,sizeof(int));

while (k < m)

{

x = rand() % n;

y = rand() % n;

if (!G[x][y] && x != y)

k += G[x][y] = G[y][x] = 1;

}

return G;

}

int \*\*CreateIncidMatr(int \*\*G,int n, int m)

{

int \*\*E = calloc(n,sizeof(int\*)), k = 0;

for (int i = 0; i < n; i++)

E[i] = calloc(m,sizeof(int));

for (int i = 0; i < n - 1; i++)

for (int j = i + 1; j < n; j++)

if (G[i][j])

E[i][k] = E[j][k++] = 1;

return E;

}

int check\_gamilton(int \*\*E, int n, int m)

{

for (int i = 0; i < n; i++)

{

int s = 0;

for (int j = 0; j < m; j++)

s += E[i][j];

if (s < n/2)

return 0;

}

return 1;

}

int check\_eiler(int \*\*E, int n, int m)

{

for (int i = 0; i < n; i++)

{

int s = 0;

for (int j = 0; j < m; j++)

s += E[i][j];

if (s % 2 == 1)

return 0;

}

return 1;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int \*\*G, \*\*E;

srand(time(NULL));

int n,m;

int k = 0, k1 = 0, k2 = 0;

n = 8;

printf("n = %i\n",n);

printf("m\tэйлеровых\t\tгамильтоновых\t\tвсего\n");

for (int m = n; m < n\*(n-1)/2; m++)

{

time\_t timer = time(NULL);

k = 0;

k1 = 0;

k2 = 0;

while (time(NULL) - timer < 10)

{

G = GenRandGraph(n,m);

E = CreateIncidMatr(G,n,m);

k1 += check\_eiler(E,n,m);

k2 += check\_gamilton(E,n,m);

k++;

freemem(E,n);

freemem(G,n);

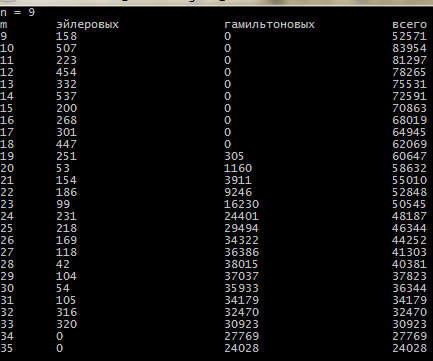
}

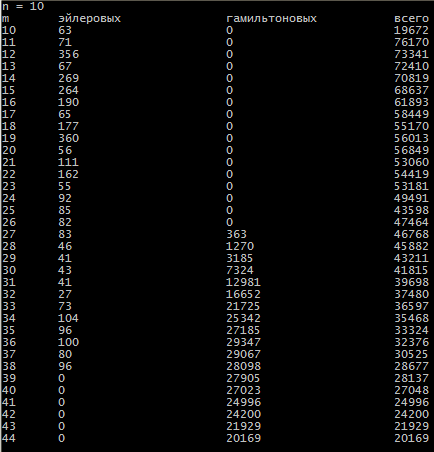
printf("%i\t%i\t\t\t%i\t\t\t%i\n",m,k1,k2,k);

}

return 0;

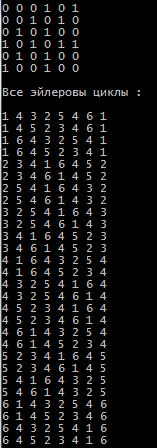
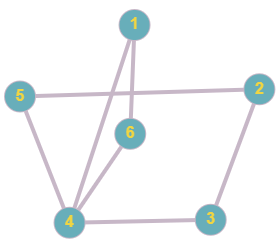
}

**Задание 3.**



**Задание 4.**

int write\_eiler\_cycles(int \*\*a, int \*w, int \*\*e, int m, int n, int i)

{

for (int x = 0; x < n; x++)

{

if (a[w[i-1]][x] && !e[w[i-1]][x])

{

w[i] = x;

if (w[0] == x && i == m)

{

for (int t = 0; t <= m; t++)

printf("%i ",w[t] + 1);

printf("\n");

}

else

{

e[w[i-1]][x] = e[x][w[i-1]] = 1;

write\_eiler\_cycles(a,w,e,m,n,i+1);

e[w[i-1]][x] = e[x][w[i-1]] = 0;

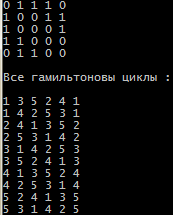
}

}

}

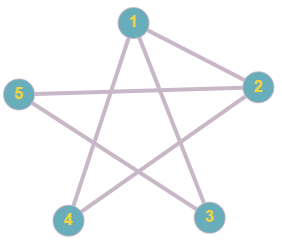
}

**Задание 5.**

int write\_gamiltons\_cycles(int \*\*a, int \*w, int \*v, int n, int i, int k)

{

for (int x = 0; x < n; x++)

 {

if (a[w[i-1]][x] && !search(v,x,k))

{

w[i] = x;

if (x == w[0] && i == n)

{

for (int t = 0; t <= k + 1; t++)

printf("%i ",w[t] + 1);

printf("\n");

}

else

{

v[k] = x;

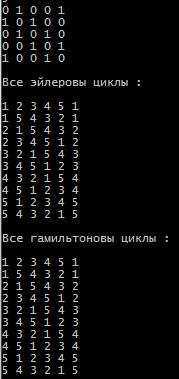
write\_gamiltons\_cycles(a,w,v,n,i+1,k+1);

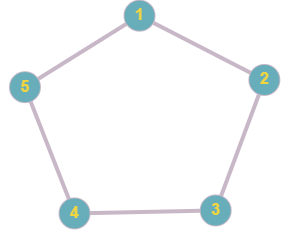
}

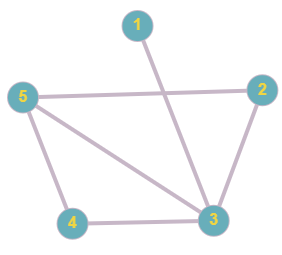
}

}

}

****

**Задание 6.**

**Задание 7.**