

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Тихоокеанский государственный университет»

Кафедра «Программное обеспечение вычислительной техники и
автоматизированных систем»

Генерация случайных графов

Лабораторная работа № 4 по дисциплине «Алгоритмы дискретной
математики»

Выполнил студент

Чекулаев В.Ю.

Факультет, группа

ФКФН, ПО(аб)-81

Проверил

Резак Е.В.

Хабаровск – 2020г.

Исходная информация: $n = 5$ — число вершин исходного графа; $m = 7$ — число ребер неориентированного графа; матрица S размером n на n , состоящая из нулей.

Требуется:

1. Выполнить программную реализацию алгоритма построения случайного неориентированного графа без петель, содержащего 5 вершин и 7 ребер.
2. По полученной матрице смежности построить неориентированный граф.

Листинг

```
#include<iostream>
#include<cstdlib>
#include<ctime>
#include<cmath>
#include<iomanip>

using namespace std;

int getRand(int b){
    return rand()%b;
}

int main(int argc, char* argv[]){
    srand(time(0));

    int m = atoi(argv[1]), n = atoi(argv[2]);
    int N, i, j, r;
    int** S = new int*[n];
    for(i = 0; i < n; i++){
        S[i] = new int[n];
    }
    for(i = 0; i < n; i++){
        for(j = 0; j < n; j++){
            S[i][j] = 0;
        }
    }

    cout << "k" << setw(3) << "|" << setw(2) << "N" << setw(2) << "|" << setw(2) << "i" <<
        setw(2) << "|" << setw(2) << "j" << setw(2) << "|" << setw(5) << "S" << setw(10) << "\n";
    cout << setfill('-') << setw(30) << "\n";
    for(int k = 0; k < m; k++){
```

```

N = getRand(n*n);
i = floor(N/n);
j = (N - (i)*n);
cout << setfill(' ') << setw(2) << k << setw(2) << "|" << setw(2) << N << setw(2) <<
"|" << setw(2) << i+1 << setw(2) << "|" << setw(2) << j+1 << setw(2) << "|" << "\n";
if((i == j) || (S[i][j] == 1)){
    m++;
} else {
    S[i][j] = 1;
    S[j][i] = 1;
}
for(i = 0; i < n; i++){
    cout << setw(4) << "|" << setw(4) << "|" << setw(4) << "|" << setw(4) << "|" <<
setw(2);
    for(j = 0; j < n; j++){
        cout << S[i][j] << " ";
    }
    cout << "\n";
}
cout << setfill('-') << setw(30) << "\n";
}

return 0;
}

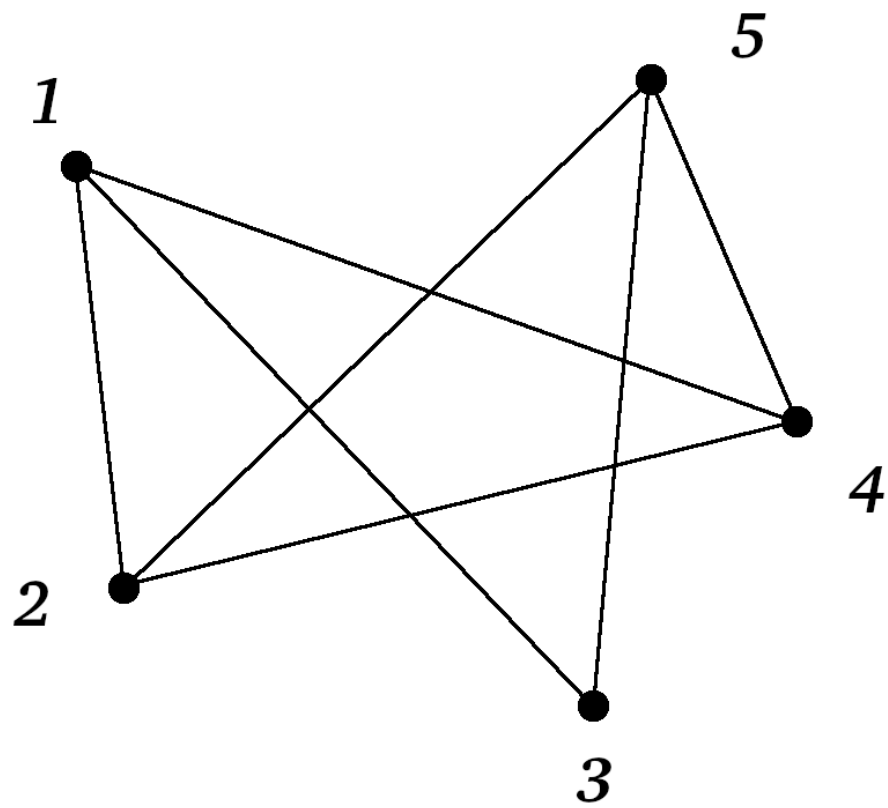
```

Результат работы программы

k	N	i	j	S
0	22	5	3	<div>0 0 0 0 0</div> <div>0 0 0 0 0</div> <div>0 0 0 0 1</div> <div>0 0 0 0 0</div> <div>0 0 1 0 0</div>
1	12	3	3	<div>0 0 0 0 0</div> <div>0 0 0 0 0</div> <div>0 0 0 0 1</div> <div>0 0 0 0 0</div> <div>0 0 1 0 0</div>
2	19	4	5	<div>0 0 0 0 0</div> <div>0 0 0 0 0</div> <div>0 0 0 0 1</div> <div>0 0 0 0 1</div> <div>0 0 1 1 0</div>
3	3	1	4	<div>0 0 0 1 0</div> <div>0 0 0 0 0</div> <div>0 0 0 0 1</div> <div>1 0 0 0 1</div> <div>0 0 1 1 0</div>
4	16	4	2	<div>0 0 0 1 0</div> <div>0 0 0 1 0</div> <div>0 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 0 1 1 0</div>

5	21	5	2	<div>0 0 0 1 0</div> <div>0 0 0 1 1</div> <div>0 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 1 1 1 0</div>
6	18	4	4	<div>0 0 0 1 0</div> <div>0 0 0 1 1</div> <div>0 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 1 1 1 0</div>
7	6	2	2	<div>0 0 0 1 0</div> <div>0 0 0 1 1</div> <div>0 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 1 1 1 0</div>
8	2	1	3	<div>0 0 1 1 0</div> <div>0 0 0 1 1</div> <div>1 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 1 1 1 0</div>
9	14	3	5	<div>0 0 1 1 0</div> <div>0 0 0 1 1</div> <div>1 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 1 1 1 0</div>
10	1	1	2	<div>0 1 1 1 0</div> <div>1 0 0 1 1</div> <div>1 0 0 0 1</div> <div>1 1 0 0 1</div> <div>0 1 1 1 0</div>

Граф по полученной матрице смежности



Вывод

В ходе лабораторной работы был рассмотрен алгоритм генерации случайного неориентированного графа. На основе теоретических данных была написана программа на языке C++. По результатам работы программы был построен граф, вид которого подтверждает правильность работы программы.