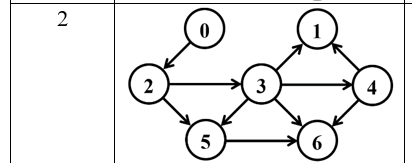
**Лабораторная работа 6. АЛГОРИТМЫ НА ГРАФАХ**

**ЦЕЛЬ РАБОТЫ:** Освоить сущность и программную реализацию: а) способов представления графов; б) алгоритмов поиска в ширину и глубину; в) алгоритма топологической сортировки графов. Разобрать алгоритм Прима и алгоритм Крускала

***Задание 1.*** Ориентированный граф **G** взять в соответствии с вариантом. Представить его в отчете в виде матрицы смежности, матрицы инцидентности, списка смежных вершин.



e10

e9

e8

e7

e6

e5

e4

e3

e2

e1

Матрица смежности

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| v\v | v0 | v1 | v2 | v3 | v4 | v5 | v6 |
| v0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| v1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| v2 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| v3 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| v4 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| v5 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| v6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Матрица инцидентности

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| v\e | e1 | e2 | e3 | e4 | e5 | e6 | e7 | e8 | e9 | e10 |
| v0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| v1 | 0 | 0 | 0 | -1 | 0 | 0 | 0 | -1 | 0 | 0 |
| v2 | -1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| v3 | 0 | -1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| v4 | 0 | 0 | 0 | 0 | -1 | 0 | 0 | 1 | 1 | 0 |
| v5 | 0 | 0 | -1 | 0 | 0 | -1 | 0 | 0 | 0 | 1 |
| v6 | 0 | 0 | 0 | 0 | 0 | 0 | -1 | 0 | -1 | -1 |

Список смежных вершин

S0 = {2}

S1 = {}

S2 = {3,5}

S3 = {1, 5, 6}

S4 = {1,6}

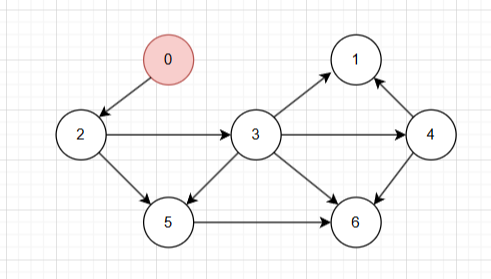
S5 = {6}

S6 = {}

***Задание 2.*** Осуществить алгоритмы поиска в ширину и глубину, а также алгоритма топологической сортировки аналогично примерам, рассмотренным на лекциях. Оформить отчет, включив в него **каждый** шаг выполнения алгоритмов.

**BFS – Поиск в ширину (очередь) 0 2 3 5 1 4 6**

**Шаг1**

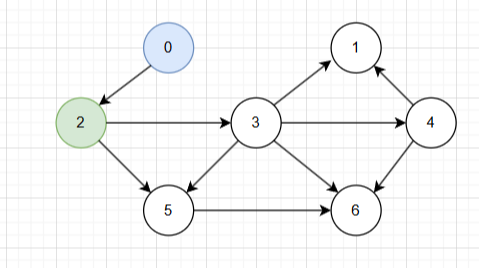
****

Посещенные вершины:

Очередь:

Текущая вершина: 0

**Шаг 2**

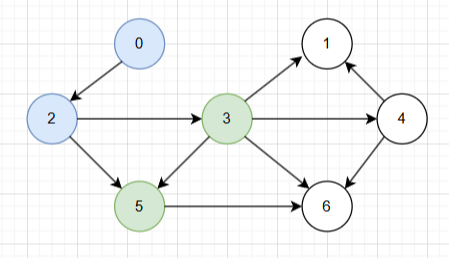
****

Посещенные вершины: 0

Очередь: 0

Текущая вершина: 2

**Шаг 3**

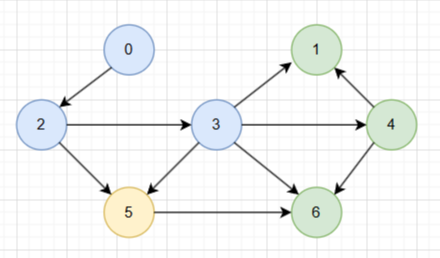
****

Посещенные вершины: 0 2

Очередь: 0 2 3 5

Текущая вершина: 3

**Шаг 4**

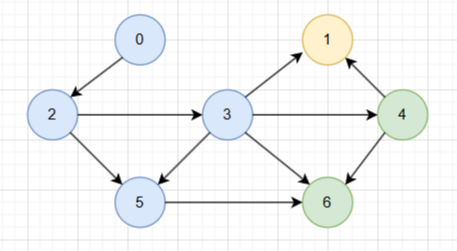
****

Посещенные вершины: 0 2 3

Очередь: 0 2 3 5

Текущая вершина: 5

**Шаг 5**

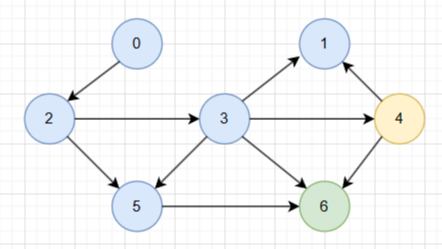
****

Посещенные вершины: 0 2 3 5

Очередь: 0 2 3 5 1 4 6

Текущая вершина: 1

**Шаг 6**

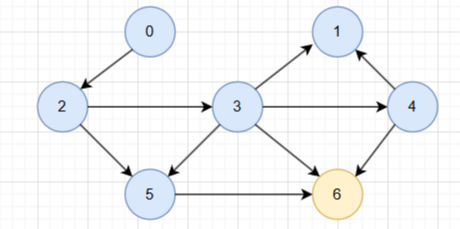
****

Посещенные вершины: 0 2 3 5 1

Очередь: 0 2 3 5 1 4 6

Текущая вершина: 4

**Шаг 7**

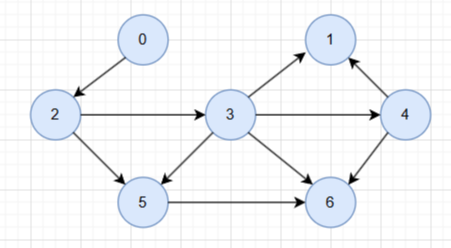
****

Посещенные вершины: 0 2 3 5 1 4

Очередь: 0 2 3 5 1 4 6

Текущая вершина: 6

**Шаг 8**

****

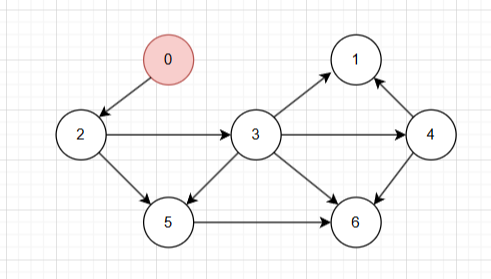
Посещенные вершины: 0 2 3 5 1 4 6

Очередь: 0 2 3 5 1 4 6

Текущая вершина:

**DFS – Поиск в глубину (стек)**

**Шаг 1**

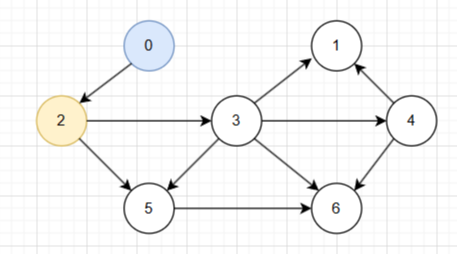
****

Посещенные вершины:

Стек:

Текущая вершина: 0

**Шаг 2**

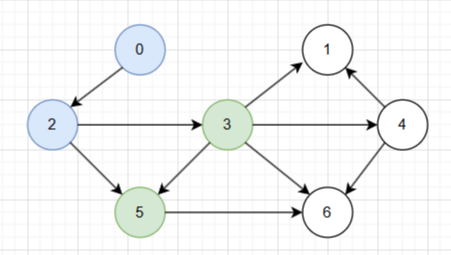
****

Посещенные вершины: 0

Стек: 2

Текущая вершина: 2

**Шаг 3**

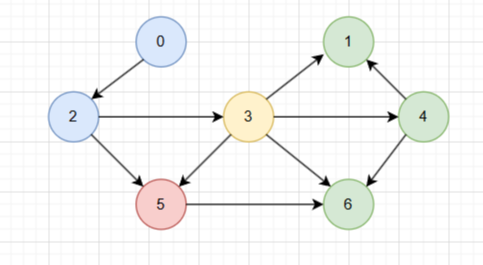
****

Посещенные вершины: 0 2

Стек: 3 5

Текущая вершина:

**Шаг 4**

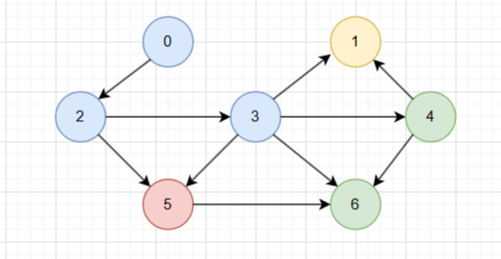
****

Посещенные вершины: 0 2

Стек: 1 4 6 5

Текущая вершина: 3

**Шаг 5**

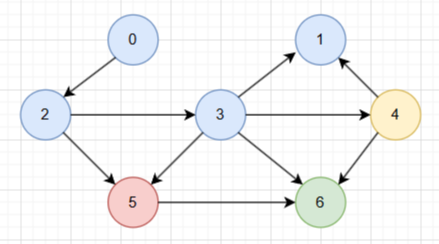
****

Посещенные вершины: 0 2 3

Стек: 4 6 5

Текущая вершина: 1

**Шаг 6**

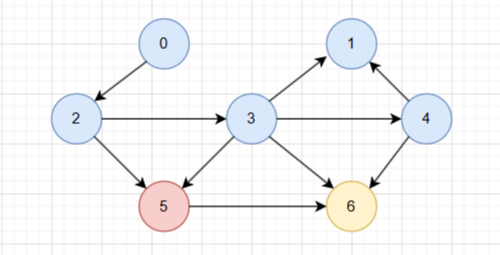
****

Посещенные вершины: 0 2 3 1

Стек: 6 5

Текущая вершина: 4

**Шаг 7**

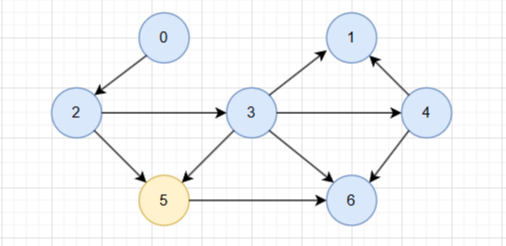
****

Посещенные вершины: 0 2 3 1 4

Стек: 5

Текущая вершина: 6

**Шаг 8**

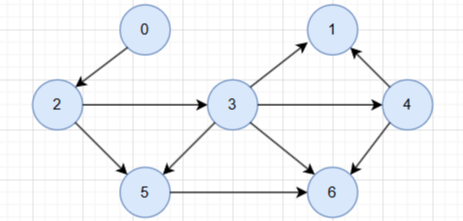
****

Посещенные вершины: 0 2 3 1 4 6

Стек:

Текущая вершина: 5

**Шаг 9**

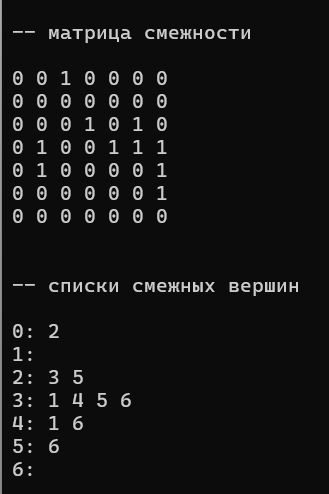
****

Посещенные вершины: 0 2 3 1 4 6 5

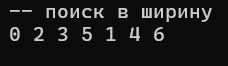
Стек:

Текущая вершина:

***Задание 3.*** Осуществить программную реализацию алгоритмов на C++. Разработать структуры AMatrix и АList для представления ориентированного графа матричным и списковым способом. Разработать функции преобразования из одного способа представления в другой. Разработать функцию BFS обхода вершин графа, используя метод поиска в ширину. Продемонстрировать работу функции. Копии экрана вставить в отчет.

****

***Задание 4.*** Разработать функцию DFS обхода вершин графа, используя метод поиска глубину. Продемонстрировать работу функции. Копии экрана вставить в отчет.

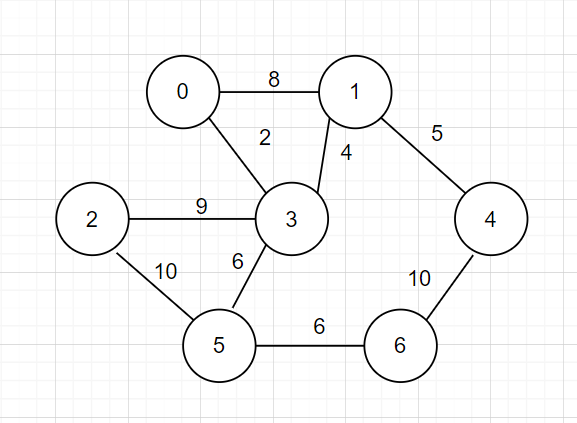
****

***Задание 5.*** Доработайте функцию **DFS**,для выполнения топологической сортировки графа. Продемонстрировать работу функции. Копии экрана вставить в отчет.

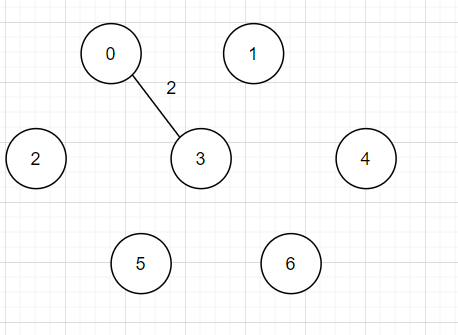


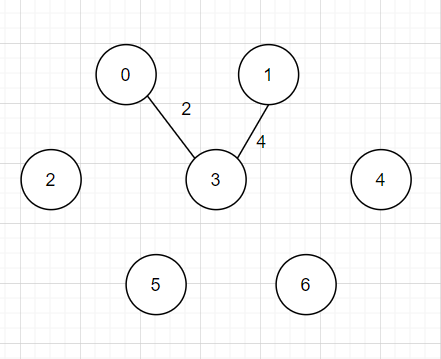
***Задание 6.*** По графу, соответствующему варианту составить минимальное остовное дерево по алгоритму Прима. Шаги построения отразить в отчете.

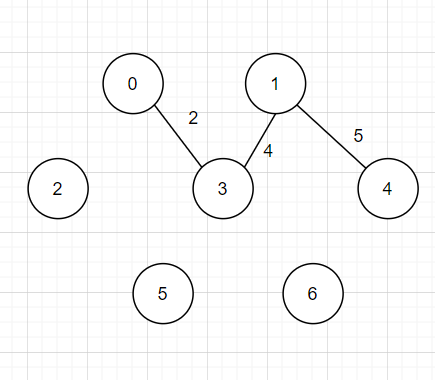
Алгоритм Прима

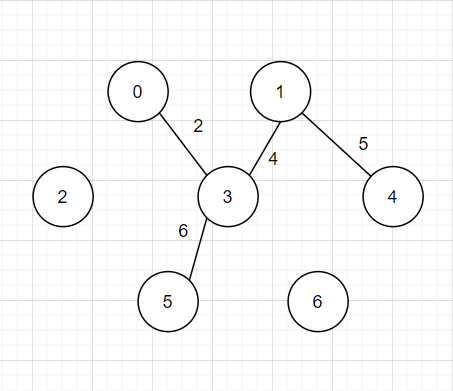


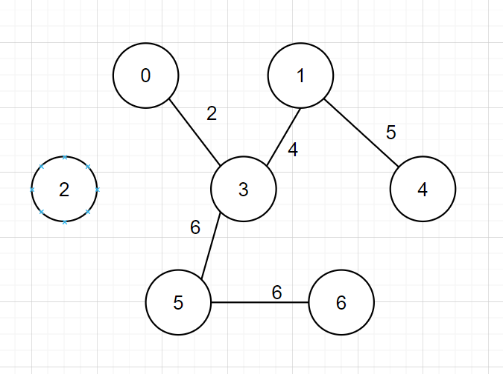
Этапы

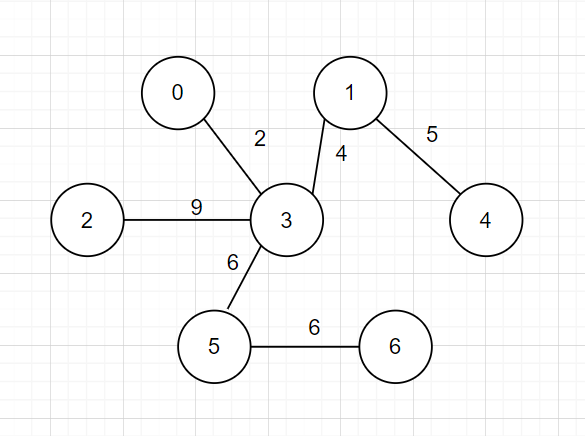












***Задание 7.*** По графу, соответствующему варианту составить минимальное остовное дерево по алгоритму Крускала. Шаги построения отразить в отчете.

Алгоритм Краскала

