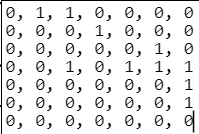
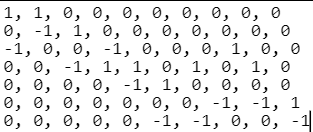
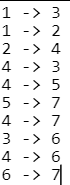
***Задание 1.*** Ориентированный граф **G** взять в соответствии с вариантом. Представить его в отчете в виде матрицы смежности, матрицы инцидентности, списка смежных вершин.



Матрица смежности –

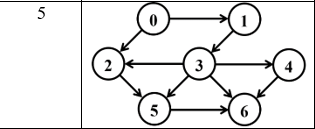


Матрица инцидентности –



Список смежных рёбер –





**Выполнение:**

**Поиск в ширину**

Выбираем стартовую вершину 0. Добавляем ее в очередь.

**Очередь: [0]**

**Посещенная очередь: []**

Посещаем вершину 0 и записываем в стек новые вершины:1,2.

**Очередь: [2,1]**

**Посещенная очередь: [0]**

Посещаем вершину 1 и записываем в стек новые вершины:3.

**Очередь: [3,2]**

**Посещенная очередь: [0,1]**

Посещаем вершину 2 и записываем в стек новые вершины:5.

**Очередь: [5,3]**

**Посещенная очередь: [0,1,2]**

Посещаем вершину 3 и записываем в стек новые вершины:4,6.

**Очередь: [6,4,5]**

**Посещенная очередь: [0,1,2,3]**

Посещаем вершину 5 и записываем в стек новые вершины:ничего.

**Очередь: [6,4]**

**Посещенная очередь: [0,1,2,3,5]**

Посещаем вершину 4 и записываем в стек новые вершины:ничего.

**Очередь: [6]**

**Посещенная очередь: [0,1,2,3,5,4]**

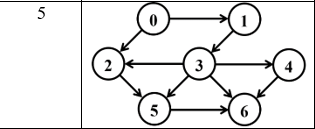
Посещаем вершину 6 и записываем в стек новые вершины:ничего.

**Очередь: []**

**Посещенная очередь: [0,1,2,3,5,4,6]**

**`**

**Поиск в глубину**



Выбираем стартовую вершину 0

**Стек: [0]**

**Посещенные вершины: []**

Посещаем 0. Добавляем в стек:1,2 .

**Стек: [1,2]**

**Посещенные вершины: [0]**

Посещаем 2. Добавляем в стек: 5.

**Стек: [1,5]**

**Посещенные вершины: [0,2]**

Посещаем 5. Добавляем в стек: 6.

**Стек: [1,6]**

**Посещенные вершины: [0,2,5]**

Посещаем 6. Добавляем в стек: 4.

**Стек: [1,4]**

**Посещенные вершины: [0,2,5,6]**

Посещаем 4. Добавляем в стек: ничего.

**Стек: [1]**

**Посещенные вершины: [0,2,5,6,4]**

Посещаем 1. Добавляем в стек: 3.

**Стек: [3]**

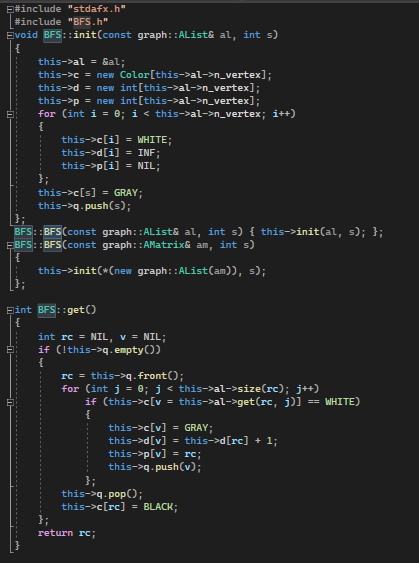
**Посещенные вершины: [0,2,5,6,4,1]**

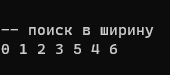
Посещаем 3. Добавляем в стек: ничего.

**Стек: []**

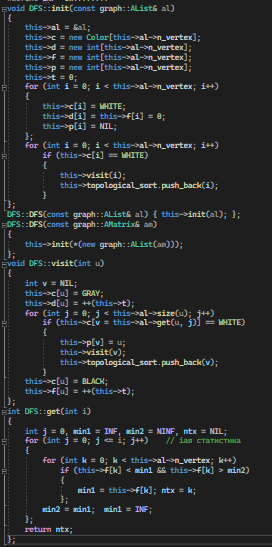
**Посещенные вершины: [0,2,5,6,4,1,3]**

***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.





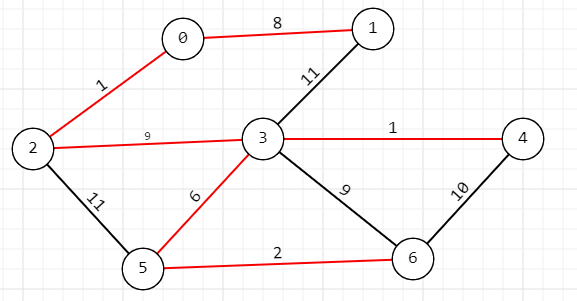
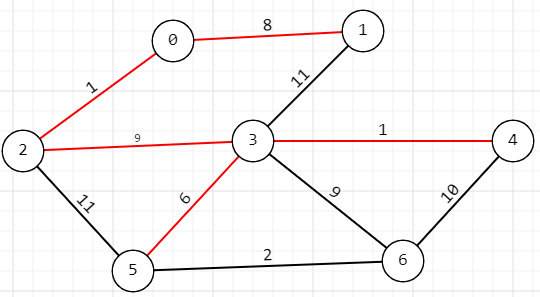
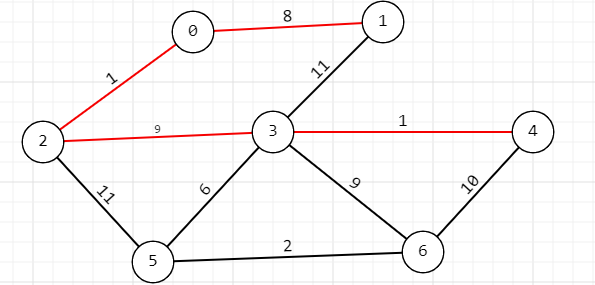
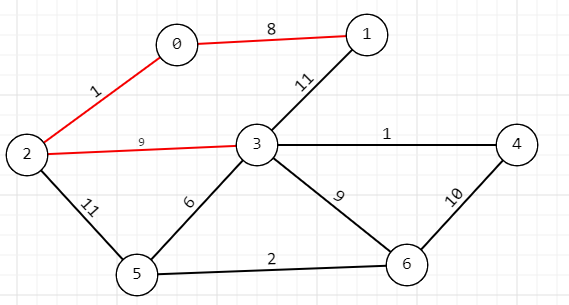
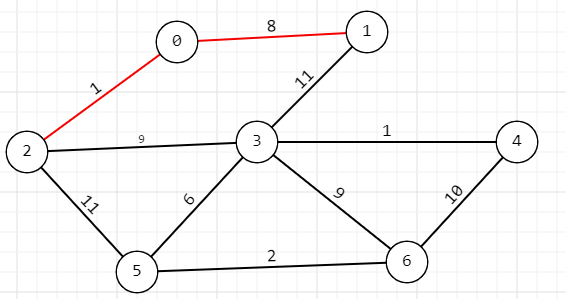
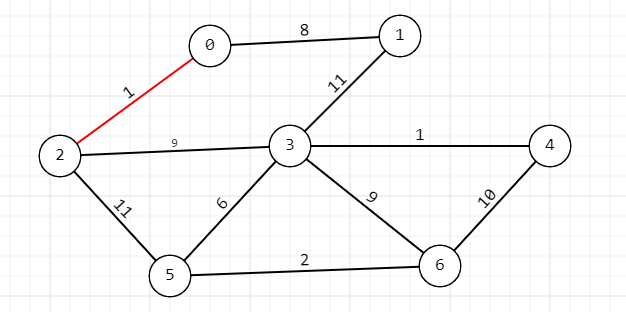
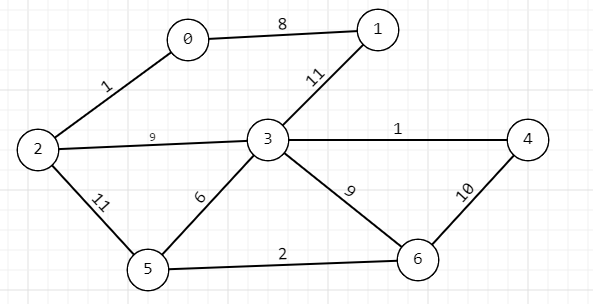
***Задание 4.*** Разработать функцию **DFS**  обхода вершин графа, используя метод поиска глубину. Продемонстрировать работу функции. Копии экрана вставить в отчет.

***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.



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

