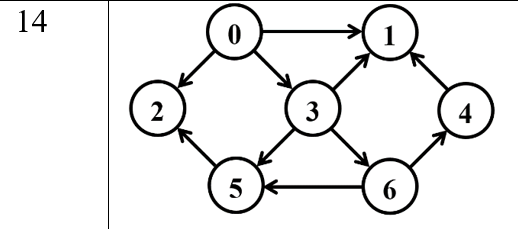
**Лабораторная работа 6. АЛГОРИТМЫ НА ГРАФАХ**

**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:



Матрица смежности:

0, 1, 1, 1, 0, 0, 0, 0

0, 0, 0, 0, 0, 0, 0, 1

0, 0, 0, 0, 0, 0, 0, 2

0, 1, 0, 0, 0, 1, 1, 3

0, 1, 0, 0, 0, 0, 0, 4

0, 0, 1, 0, 0, 0, 0, 5

0, 0, 0, 0, 1, 1, 0, 6

Матрица инцидентности:

1, 1, 1, 0, 0, 0, 0, 0, 0, 0 0

-1, 0, 0, 0, 0, 0, 0, 0, -1, -1 1

0, -1, 0, -1, 0, 0, 0, 0, 0, 0 2

0, 0, -1, 0, 1, 1, 0, 0, 1, 0 3

0, 0, 0, 0, 0, 0, 0, -1, 0, 1 4

0, 0, 0, 1, -1, 0, -1, 0, 0, 0 5

0, 0, 0, 0, 0, -1, 1, 1, 0, 0 6

Вертикально вершины, горизонтально – ребра

Список смежных ребер:

V0:v1,v2,v3

V1: 0

V2:0

V3:v1,v5,v6

V4:v1

V5:v2

V6:v4,v5

Задание 2:

Поиск в ширину: 0->1->2->3->5->6->4. По принципу очереди

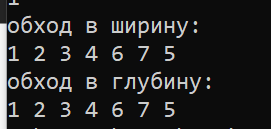
Поиск в глубину: 0->1->2->3->5->6->4. По принципу стека

Топологическая сортировка

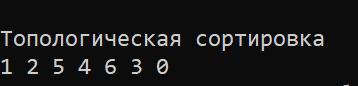
[1,2,5,4,6,3,0]

Заходим в вершину, если из нее нет путей – добавляется в результат, если есть – обход в глубину по ней, пока не найдем ту, которую уже посетили или из которой нет путей.

Задание 3+4. Отнять 1 от индексов вершин

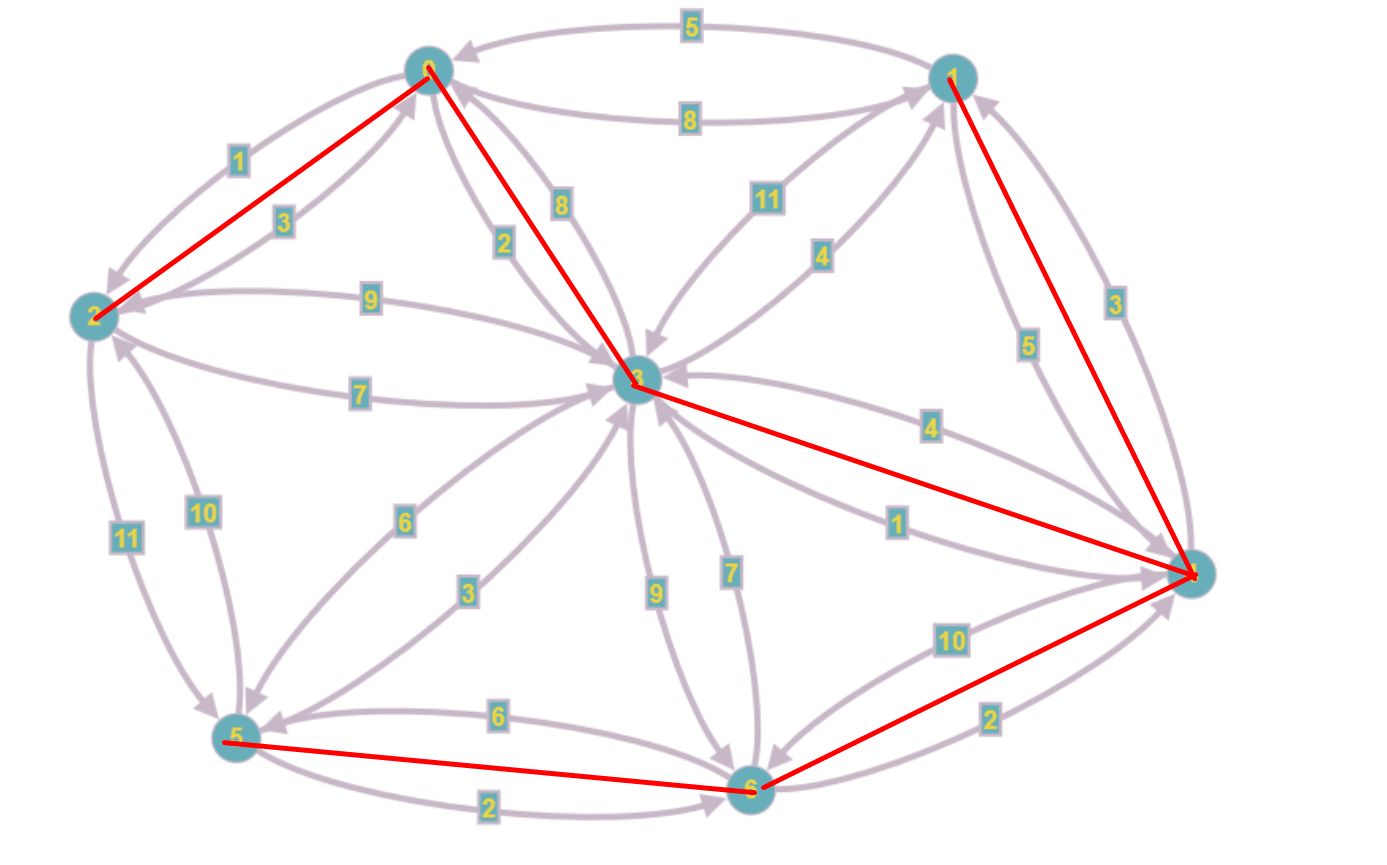


Задание 5:



Задание 6. Алгоритм прима

Добавляем 1 вершину во множество u, ищем ребро с минимальной стоимостью, которое связывает вершины из множества u и вершину не из множества.



Остовное дерево состоит из ребер 0->3, 3->4, 4->1, 0->2, 5->6, 4->6

Задание 7. Алгоритм Крускала

Берем ребро минимальной стоимости, которое соединяет 2 разные компоненты на каждом шаге

