Частное учреждение образования

«Колледж бизнеса и права»

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ  Ведущий методист колледжа  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Е.В. Паскал  « \_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 года |
| Специальность 2-40 01 01 «Программное обеспечение информационных технологий» | Учебная дисциплина «Основы алгоритмизации и программирование» |

**Лабораторная работа № 38**

**Инструкционно-технологическая карта**

Тема: Разработка и отладка программ, реализующих алгоритмы Дж. Краскала и Р. Прима.

Цель: Научиться разрабатывать программы, реализующие алгоритмы Дж. Краскала и Р. Прима; научиться выполнять их отладку.

Время выполнения: 2 часа.

1. **Порядок выполнения работы**
2. Изучить теоретические сведения к лабораторной работе.
3. Разработать на языке С++ программу вывода на экран решения задачи в соответствии с вариантом индивидуального задания, указанным преподавателем.
4. Отлаженную, работающую программу сдать преподавателю. Работу программы показать с помощью самостоятельно разработанных тестов.
5. Ответить на контрольные вопросы.
6. **Теоретические сведения**

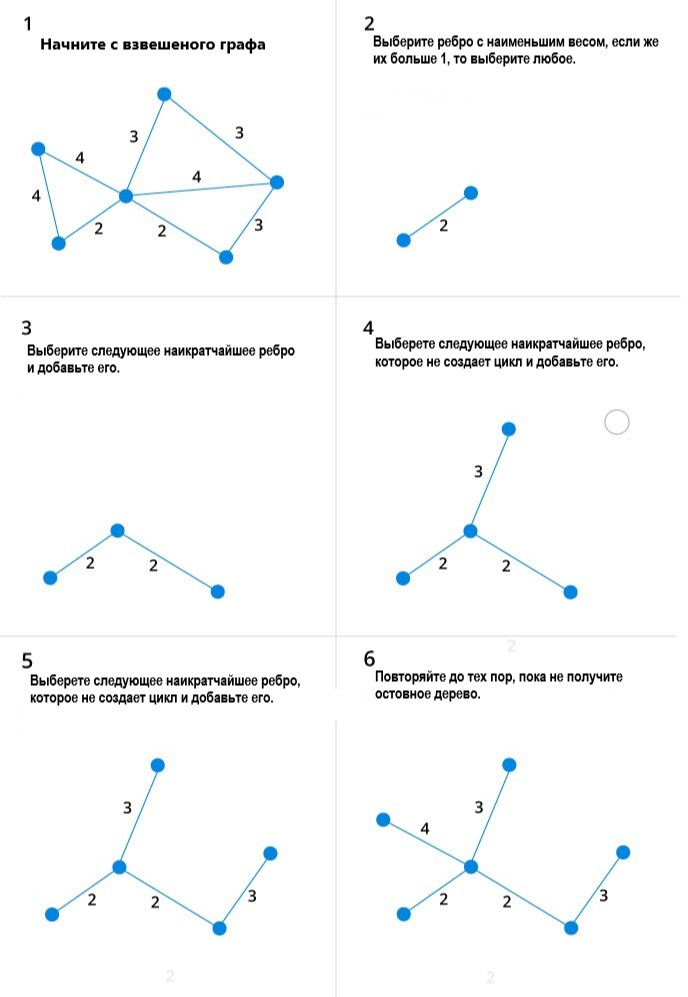
**Алгоритм Дж. Краскала поиска остова графа минимального веса**

Алгоритм Дж. Краскала работает для взвешенного неориентированного графа. **Взвешенный граф** – граф, все ребра которого имеют определенный вес (число). **Неориентированный граф** – это граф, все ребра которого не имеют направления, т.е. по любому ребру можно пройти слева направо и справа налево. **Остов** или **остовное дерево графа** – это дерево, подграф данного графа, с тем же числом вершин, что и у исходного графа. Остовное дерево можно получить из исходного графа путем удаления максимального количества ребер, входящих в циклы, но без нарушения связности графа. **Цикл (циклический граф)** – это ситуация в графе, при которой из данной вершины можно выйти по одному связанному с ней ребру и, идя далее (только вперед) по связным ребрам, можно прийти (вернуться) в данную вершину по другому ребру. Остовное дерево включает в себя все N вершин исходного графа и содержит всегда N – 1 ребро. **Остовное дерево** – это ациклический связный подграф данного связного неориентированного графа, в который входят все его вершины. **Связанный граф** – это граф, у которого нет ни одной вершины, которая бы не была связанна с этим графом хотя бы одним ребром. Алгоритм Дж. Краскала относится к классу алгоритмов, называемых **«жадными» алгоритмами**, которые находят локальный оптимум в надежде найти глобальный оптимум. Согласно алгоритму Дж. Краскала мы начинаем с ребер с наименьшим весом и продолжаем добавлять ребра, пока не достигнем цели.

Шаги для реализации алгоритма Краскала следующие:

1. Сортировать все ребра от малого веса до высокого.
2. Возьмите ребро с наименьшим весом и добавьте его в остовное дерево. Если добавление ребра создало цикл, то отклоните это ребро.
3. Продолжайте добавлять ребра, пока не достигнете всех вершин.

**Визуализация работы алгоритма Краскала**

[](https://evileg.com/users/mafulechka/albums/photo/825/)

**Алгоритм Р. Прима поиска остова графа минимального веса**

Алгоритм Прима (нахождения «ближайшего» соседа) отличается от алгоритма Дж. Краскала тем, что на каждом этапе не требует ни сортировки, ни проверки на цикличность. Алгоритм Р. Прима:

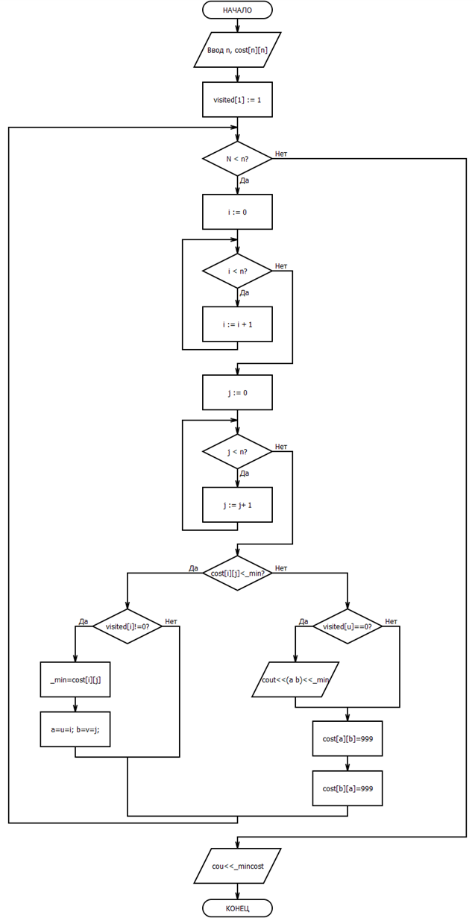
1. Построение остовного дерева T начинается с произвольной вершины V.
2. Среди ребер, инцидентных (связанных напрямую) вершине V, выбирается ребро V-U наименьшего веса и включается в дерево T.
3. Повторяя процесс, выполняется поиск наименьшего по весу ребра, соединяющего вершины V или U с некоторой другой вершиной графа, соответствующей условию из пункта 2.
4. Процесс включения ребер продолжается до тех пор, пока все вершины исходного графа G не будут включены в дерево T. Построенное дерево T будет минимальным остовным деревом.

Алгоритм Р. Прима следует схеме алгоритма построения минимального остовного дерева: на каждом шаге мы добавляем к строящемуся остову безопасное ребро. Алгоритм Прима относится к группе алгоритмов наращивания минимального остова: на каждом шаге существует не более одной нетривиальной (не состоящей из одной вершины) компоненты связности, и каждый раз к ней добавляется ребро наименьшего веса, соединяющее вершины компоненты с остальными вершинами. По теореме такое ребро является безопасным.

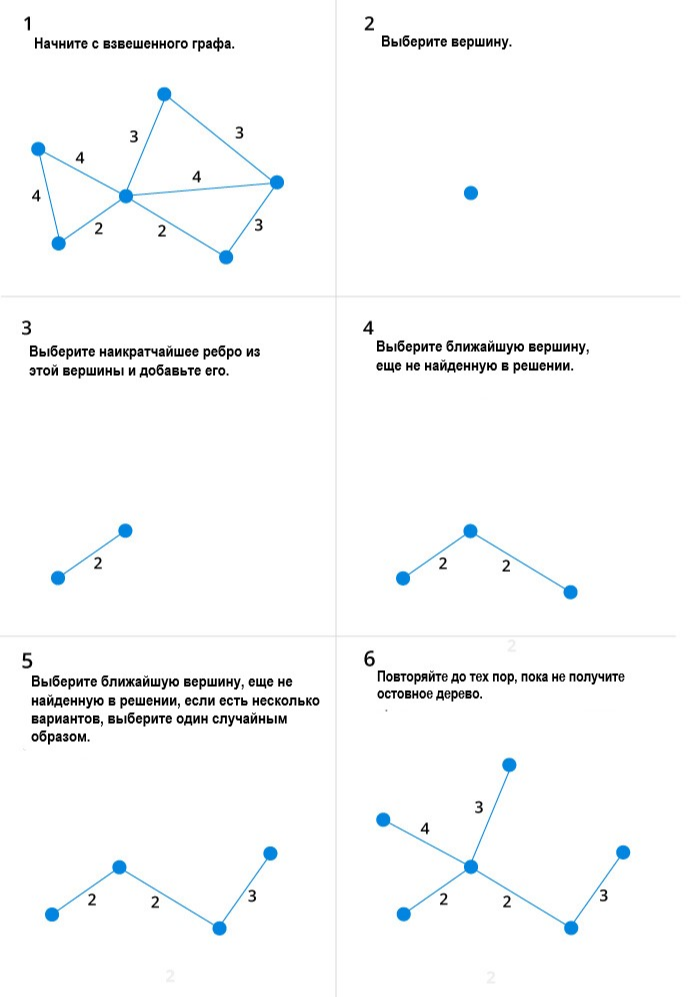
Неформальное описание алгоритма:

1. Выбрать корневую вершину V;
2. Пометить вершину V как посещенную;
3. Для каждой смежной с V вершины установить затраты кратчайшего ребра;
4. Выбрать не посещённую вершину с наименьшим весом ребра в качестве текущей вершины;
5. Пометить вершину V как посещеннуюи добавить связывающее ребро в остовное дерево;
6. Повторить все шаги со второго, пока не будут посещены все вершины.

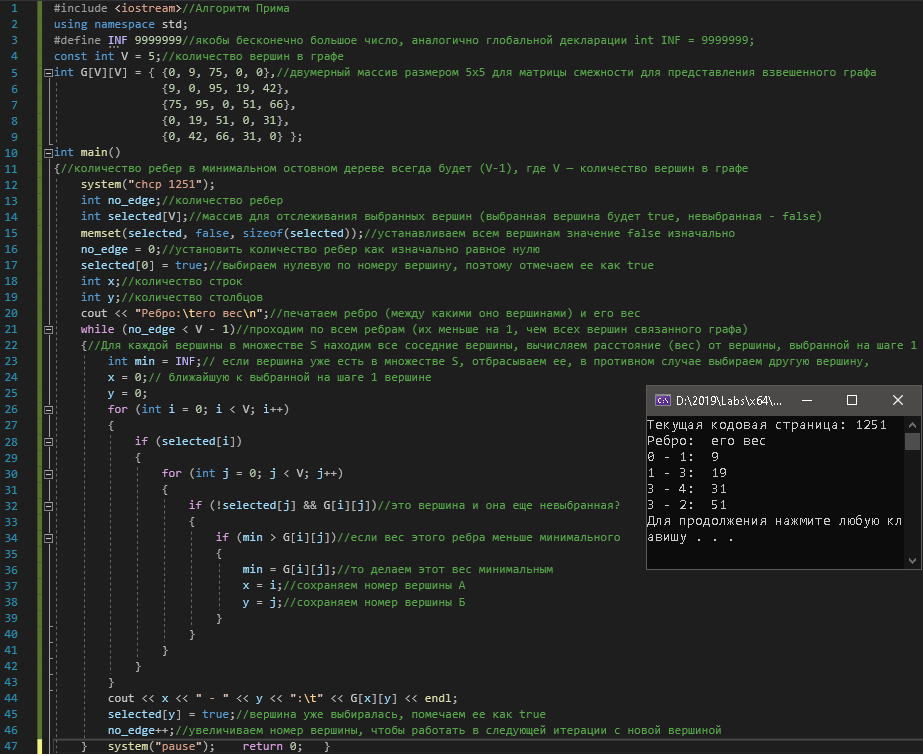
На рисунке представлена блок-схема алгоритма Прима.

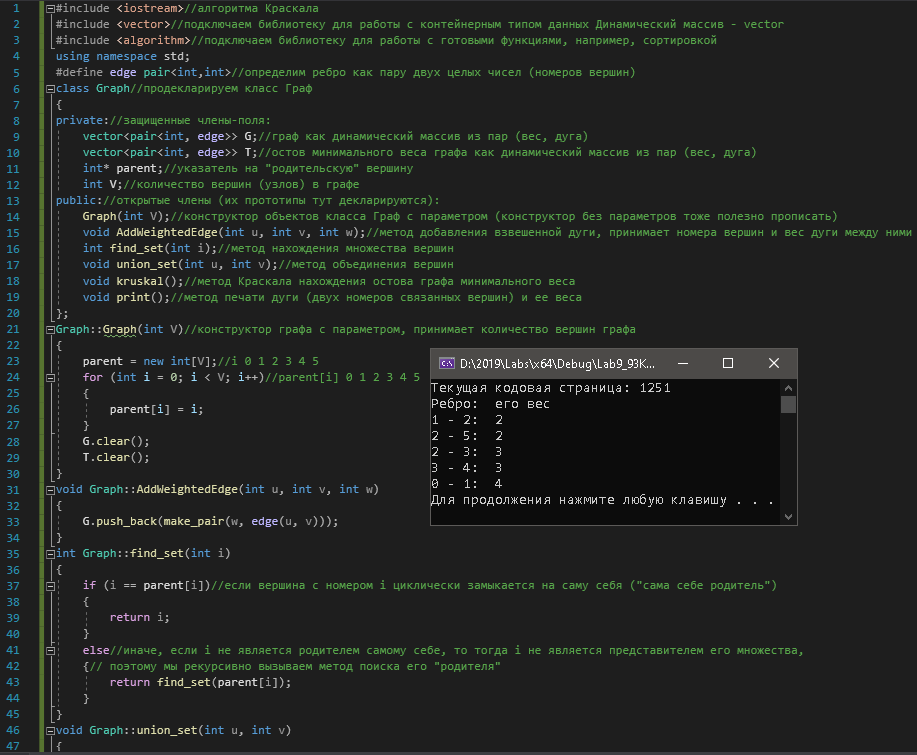


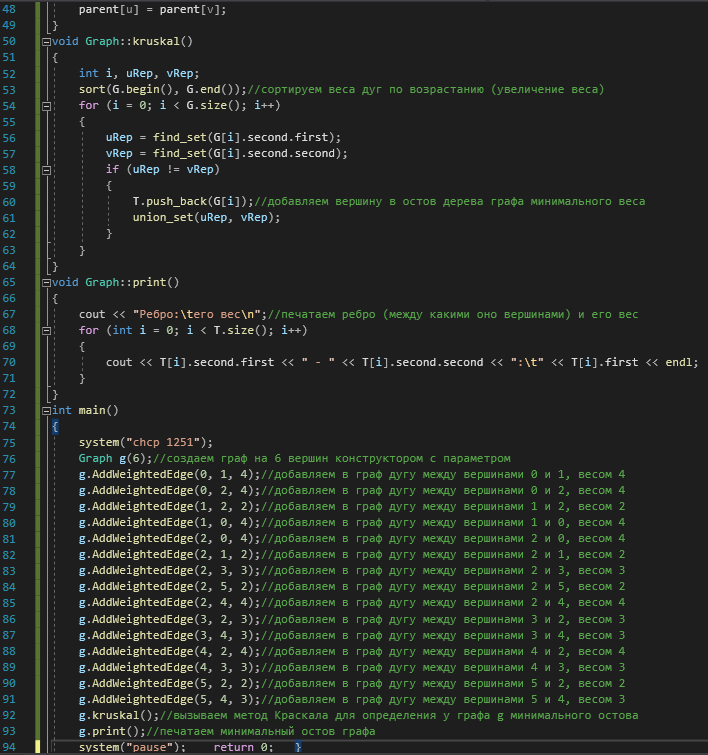
**Визуализация работы алгоритма Р. Прима**

[](https://evileg.com/users/mafulechka/albums/photo/827/)

1. **Пример выполнения программы**







1. **Задания по вариантам**
2. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Краскала (отдельной функцией).

**1**

**2**

**8**

**3**

**42**

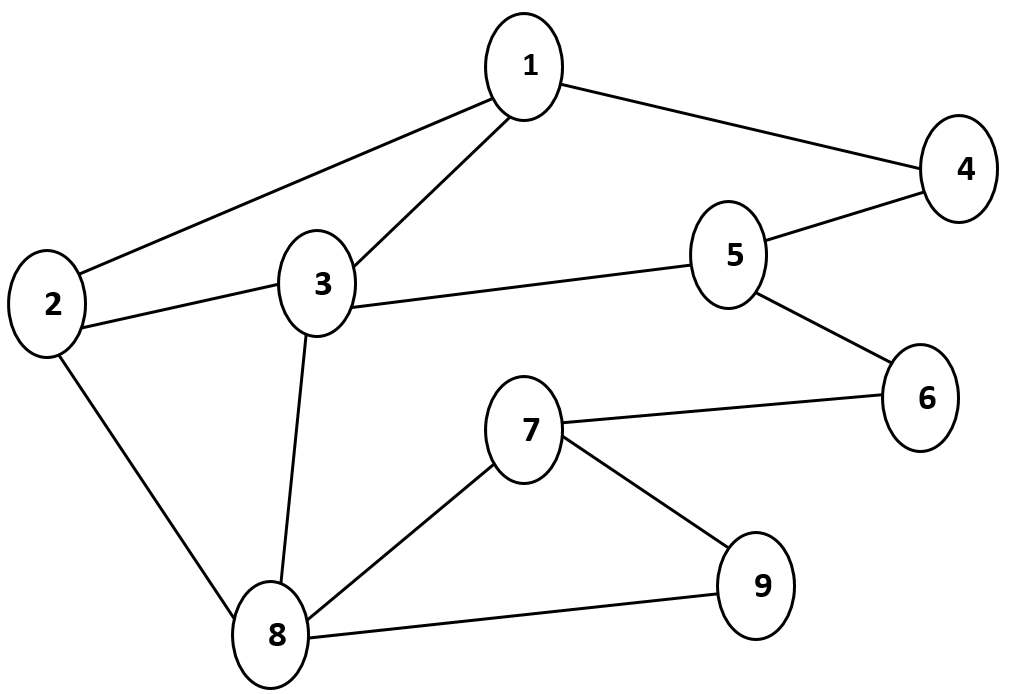
**5**

**9**

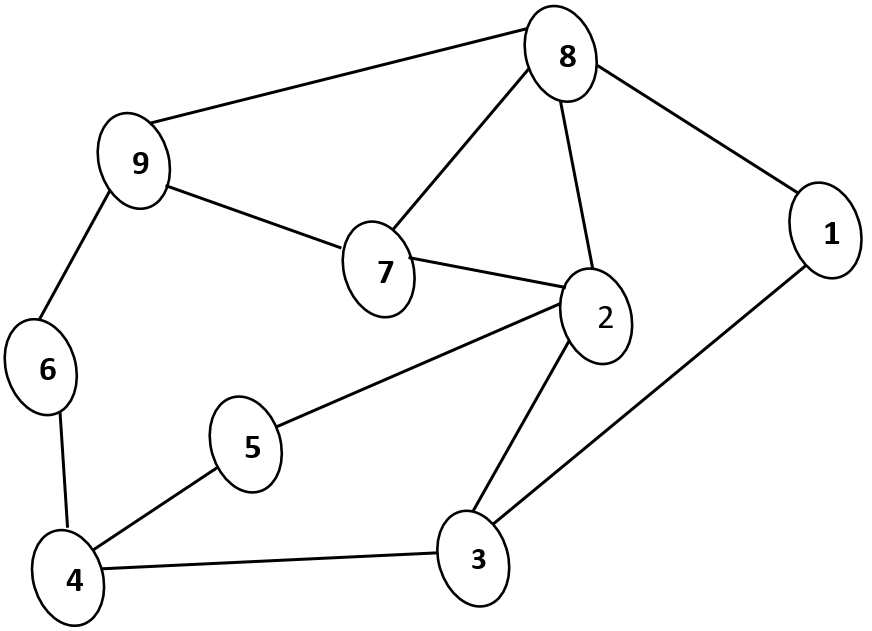
**7**

**6**

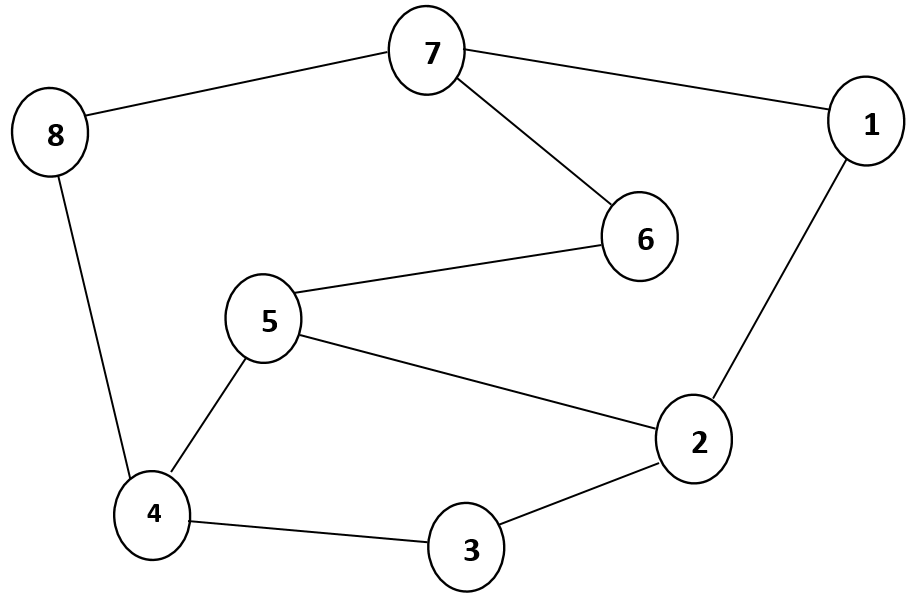
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Прима (отдельной функцией).



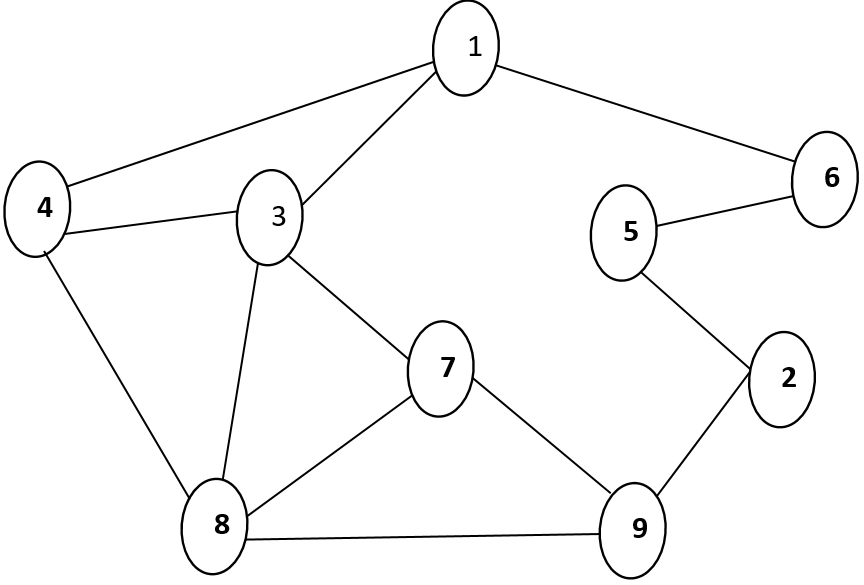
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Найти граф минимального веса по алгоритму Краскала.



1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Прима (отдельной функцией).



1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Найти граф минимального веса по алгоритму Прима



1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Краскала (отдельной функцией).

**1**

**2**

6

**42**

**5**

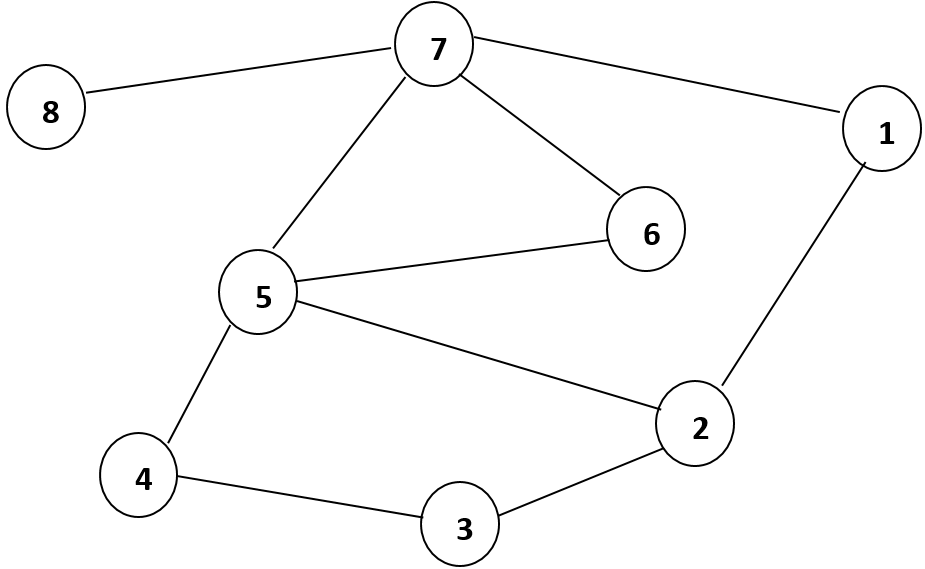
**9**

**7**

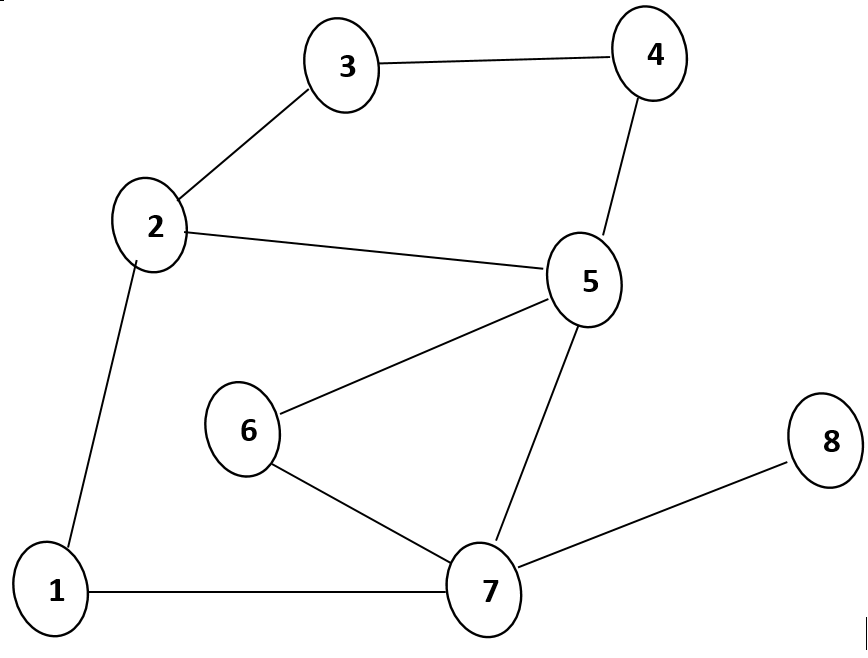
**3**

8

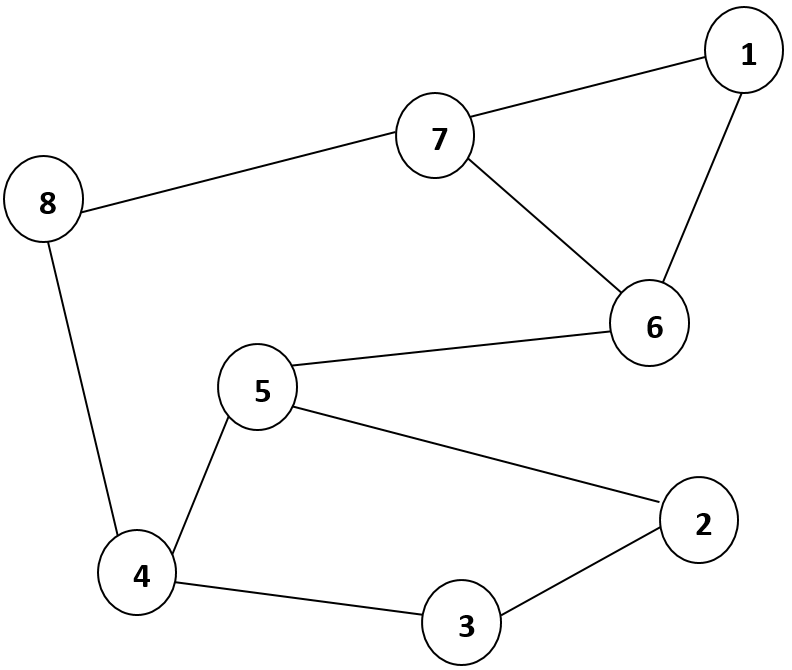
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Краскала (отдельной функцией).

****

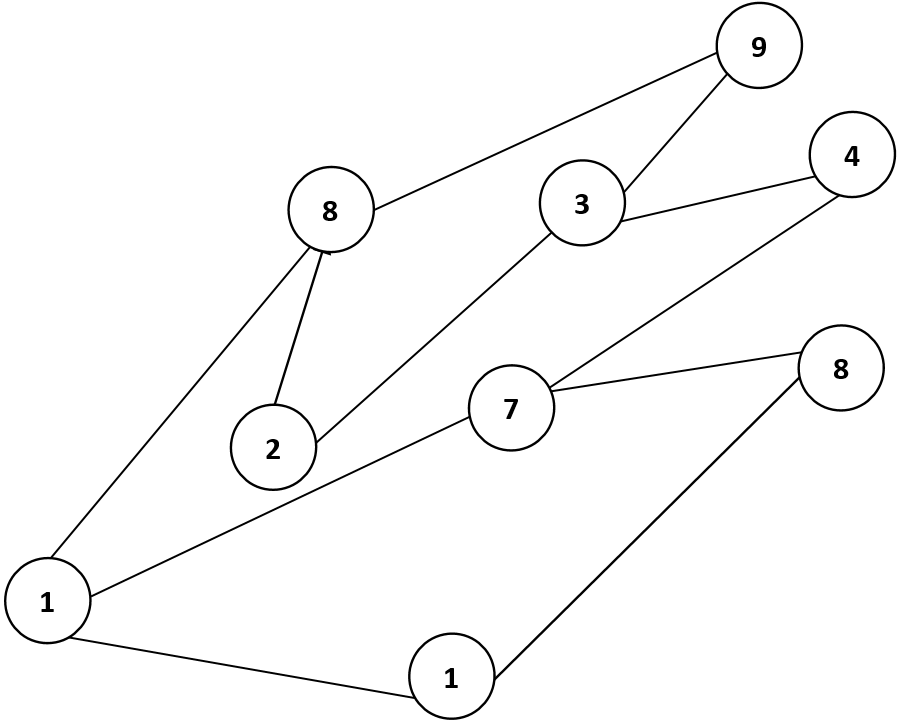
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Прима (отдельной функцией).



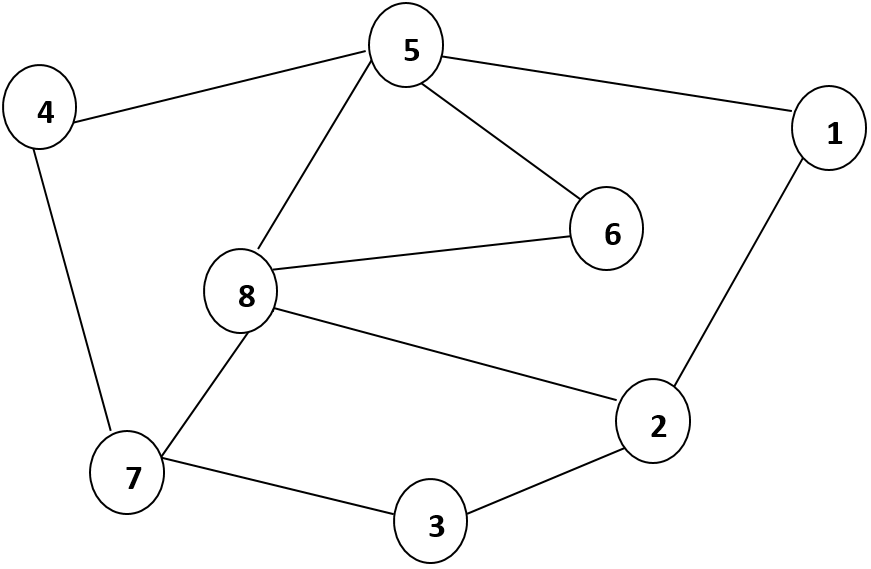
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Найти граф минимального веса алгоритмом Краскала.



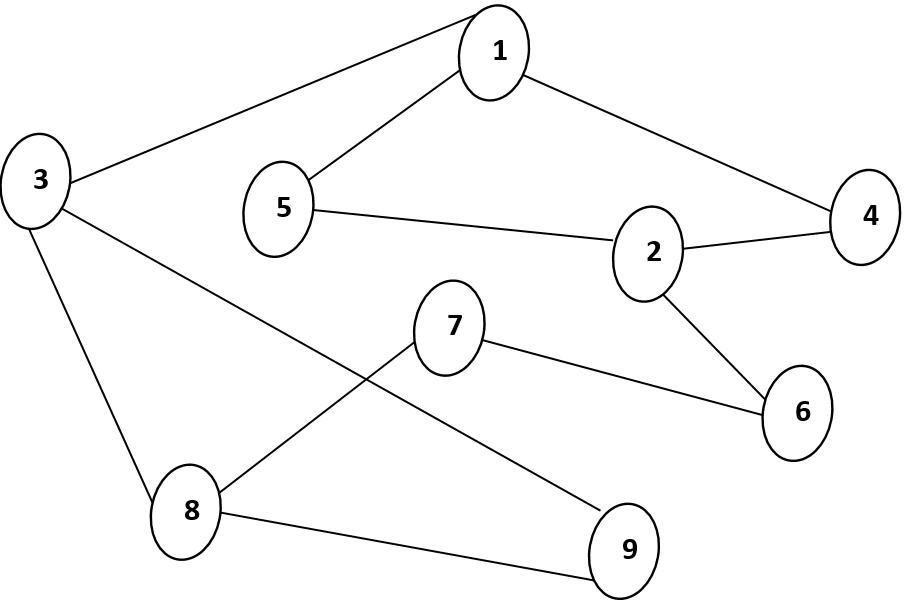
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Прима (отдельной функцией).



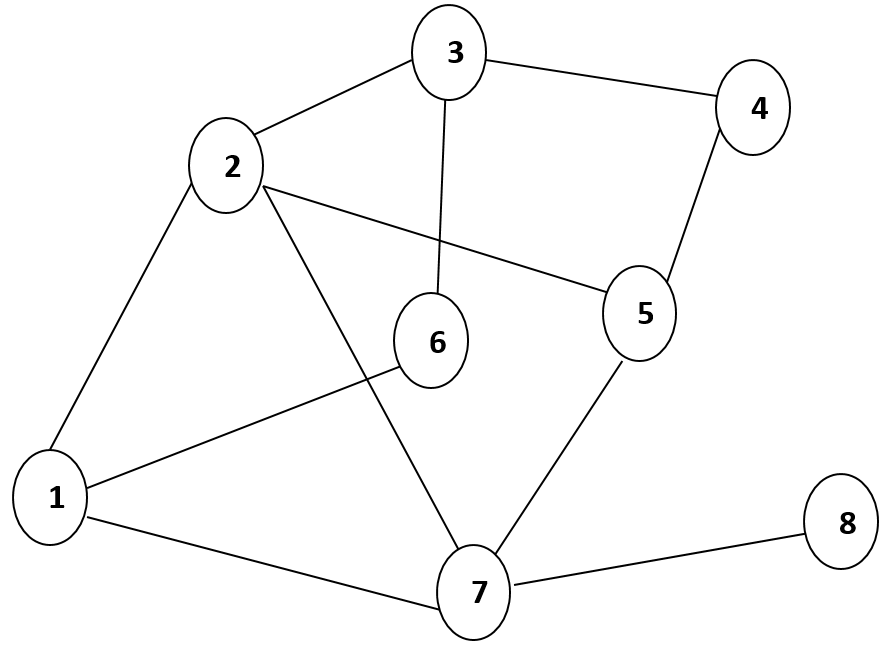
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Краскала (отдельной функцией).



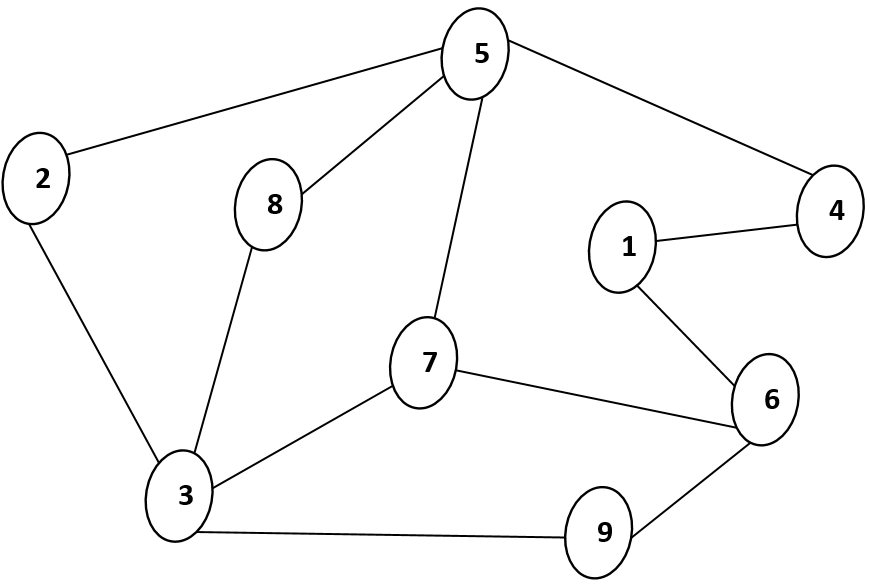
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Прима (отдельной функцией).



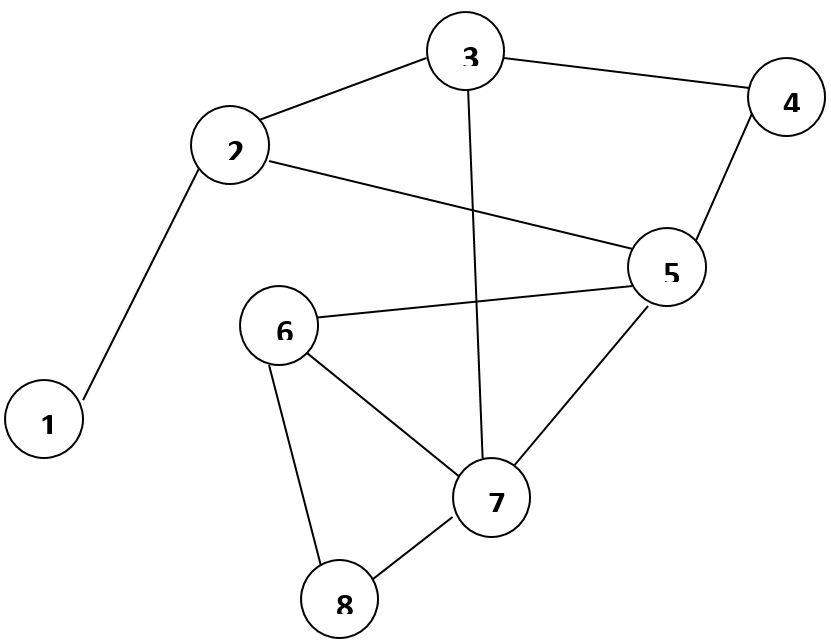
1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Краскала (отдельной функцией).



1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Реализовать алгоритм Прима (отдельной функцией).



1. Вывести матрицу смежности для графа, представленного на рисунке, где вес ребер задается случайным образом. Найти граф минимального веса для графа по алгоритму Краскала.



1. **Контрольные вопросы**
2. Дайте определение понятию «взвешенный граф».
3. Дайте определение понятию «неориентированный граф».
4. Дайте определение понятию «ориентированный граф».
5. Дайте определение понятию «циклический граф».
6. Дайте определение понятию «ациклический граф».
7. Дайте определение понятию «связный граф».
8. Дайте определение понятию «несвязный граф».
9. Чем класс «жадных» алгоритмов отличается от класса алгоритмов «ближайшего соседа»?
10. Опишите алгоритм Р. Прима.
11. Опишите алгоритм Дж. Краскала.

**Литература**

**Дейтел,** Х.М. Как программировать на С++ / Х.М. Дейтел, П.Дж. Дейтел . – М. : Бином-Пресс , 2018 . – 1456 с.

**Павловская**, Т.А. С++. Объектно-ориентированное программирование : практикум / Т.А. Павловская, Ю.А. Щупак . – СПб. : Питер , 2019 . – 265 с.

**Страуструп**, Б. Язык программирования С++ / Б. Страуструп . – СПб. : Бином-Пресс , 2019 . – 1054 с.

Преподаватель Шаляпин Ю.В.

|  |
| --- |
| Рассмотрено на заседании цикловой  комиссии ПОИТ № 10  Протокол №\_\_\_\_от «\_\_\_»\_\_\_\_\_\_\_\_\_\_2021 г.  Председатель ЦК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |