

Спецификация

Цель работы:

Разработка программного приложения на языке Java, предназначенного для визуализации алгоритма Прима.

Описание алгоритма:

На вход алгоритма подаётся связный неориентированный граф. Для каждого ребра задаётся его стоимость.

Сначала берётся произвольная вершина и находится ребро, инцидентное данной вершине и обладающее наименьшей стоимостью. Найденное ребро и соединяемые им две вершины образуют дерево. Затем, рассматриваются рёбра графа, один конец которых — уже принадлежащая дереву вершина, а другой — нет; из этих рёбер выбирается ребро наименьшей стоимости. Выбираемое на каждом шаге ребро присоединяется к дереву. Рост дерева происходит до тех пор, пока не будут исчерпаны все вершины исходного графа.

Результатом работы алгоритма является остовное дерево минимальной стоимости.

Входные данные:

Данные представлены списком инцидентности в файле “input.txt”. Входные данные должны быть типа int, максимальный вес ребра 1000. Вершины нумеруются цифрами в произвольном порядке. В первой строке задается количество вершин, во второй строке номер вершины и количество инцидентных вершин (“номер вершины” пробел “количество инцидентных вершин”), затем в следующих строках перечисляются инцидентные вершины с соответствующими весами (“номер вершины” пробел “вес ребра”).

Выходные данные :

На экран выводится граф, ребра, которые входят в минимальное остовное дерево, будут окрашены цветом отличным от цвета исходного графа. В поле Text illustration будут выводиться пошаговые комментарии к работе алгоритма.

Требования к программе:

Программа реализует алгоритм Прима с визуализацией графа, и пошаговой визуализацией работы алгоритма.

Требования к пользователю:

Пользователь должен обладать базовыми знаниями в теории графов.

При запуске проекта будет создаваться отдельное окно (рисунок 1). Данная программа будет иметь 2 окна. Первое окно представляет собой меню.

Второе представляет собой окно по работе с графом (рисунок 2). Сверху располагаются кнопки, слева находится окно вывода графа, справа текстовые пояснения. :

- 1) Поле интерфейса создания графа. Пока данные не будут загружены, будут активны только кнопки загрузить данные и выйти из программы.
 - a) Upload input. При нажатии левой кнопкой мыши этой кнопки, данные будут загружены из файла. Если данные будут заданы неправильно, выведется окно с сообщением об ошибке
 - b) Enter the starting node. При нажатии этой кнопки, нужно будет ввести стартовую вершину, с которой начнет работу алгоритм
 - c) Start algorithm. При нажатии этой кнопки программа запускает алгоритм Прима.
 - d) Next step. При нажатии этой кнопки программа переходит к обработке следующей вершины, все рассматриваемые ребра окрашиваются в синий цвет. В окне "Текстовые пояснения" будут подробно описываться алгоритм на данном шаге.
 - e) Finish algorithm. При нажатии этой кнопки можно завершить работу алгоритма для данного графа, и снова загрузить данные.

При нажатии этой кнопки рабочая область очищается от рисунка и текста.

f) Exit. При нажатии этой кнопки , пользователь совершает выход из программы

2) Меню (рисунок 1)

a) Start. При нажатии левой кнопкой мыши .Открывает окно рабочей области графа

b) About program . При нажатии кнопки открывается подробная инструкция о работе с программой

c) Exit. Выход из программы



Отформатировано: Шрифт: (По умолчанию) Times New Roman, 14 pt

Рисунок 1 – главное окно

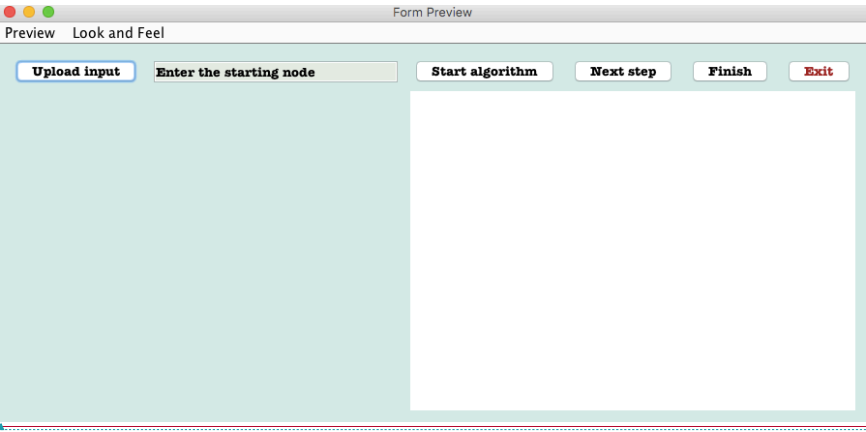


Рисунок 2 – Окно рабочей области

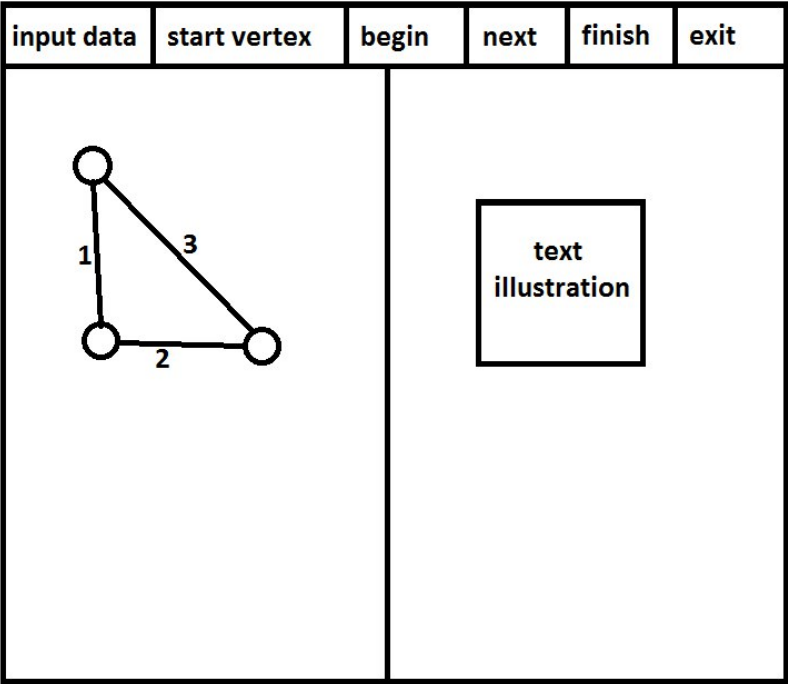


Рисунок 3 – Макет рабочей области

Отформатировано: Шрифт: (По умолчанию) Times New Roman, 14 pt

План разработки

23.06.17 Согласование спецификации.

26.06.17 Бета-версия. Предоставление частичной функциональности работы проекта. Полностью представлен интерфейс, без пошагового вывода основного дерева, без проверок на некорректные данные. Кнопка next не будет функционировать. Приемочные тесты для бета-версии.

28.06.17 Конечный проект. Демоверсия. Предоставление проекта с полной функциональностью, с пошаговым выводом. Приемочные тесты. Проект пояснительной записки

30.06.17 Конечный проект. Готовая для сдачи версия и пояснительная записка с исправлениями по замечаниям.

Текущий план представляет собой крайние сроки(deadline) для сдачи определенной функциональности проекта. Возможно данные сроки будут корректироваться и проект будет написан раньше срока, так как реализация определенной функциональности будет проходить параллельно.

Распределение обязанностей.

Горбатьюк Д.А Реализация алгоритма Прима на языке Java. Разработка пользовательского интерфейса и визуализация.

Филиппова В.А Разработка структур данных. Разработка пользовательского интерфейса и визуализация.

Максимова Е.С Оформление пояснительной записки и управление версиями программы в репозитории, разработка приемочных тестов и тестирование. Разработка пользовательского интерфейса и визуализация.

Структуры данных

Классы:

Algorithm – реализация алгоритма Прима

Graph – содержит в себе информация о графе и реализует доступ к его данным

Interface – реализует интерфейс

Main – создание окна