

ПРОГРАММА ДЛЯ РАБОТЫ С  
ГРАФАМИ (GRIN)  
ДИСКРЕТНАЯ  
МАТИМАТИКА

Работа выполнена студентом колледжа ЯРПК. Д-КС  
20

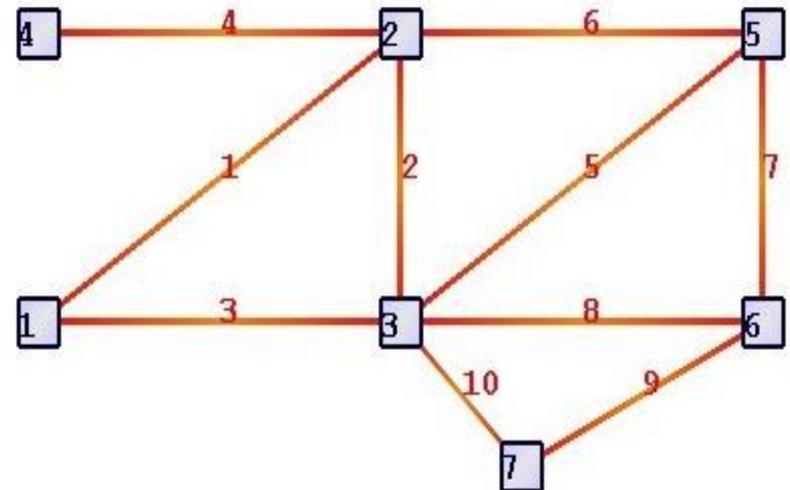
Грицукон.А.А

# Содержание:

- Описания программы Graph Calculator
- Поиск кратчайшего пути между двумя вершинами
- Дополнительные модули программы
- Примеры неориентированных графов
- Описание работы программы

# Описания программы Graph Calculator

Программа работает с помощью графического интерфейса позволяет, пользователю рисовать различные графы с помощью инструментов «Вершина» и «Ребро», расположенных на панели инструментов. Предусмотрена возможность масштабирования изображения в окне, сохранения данных в файл и чтения данных из файла.



**Поиск кратчайшего пути между двумя вершинами**, выбранными пользователем (алгоритм Дейкстры). Пользователь выбирает вершины с помощью щелчка мышью. Должна быть предусмотрена возможность выбора способа задания длины (веса) ребер: использовать длины отрезков на рисунке, считать длины всех ребер единичными, либо задать вес

The screenshot shows a window titled "Программа 'Графоанализатор' 1.3". The menu bar includes "Файл", "Граф", "Изменить", "Вид", "Режимы работы", "Алгоритмы", "Особые алгоритмы", "Справка", and language options "Rus" and "En". The toolbar contains various icons for graph manipulation. The main canvas displays a graph with five nodes: Калининград, Светлогорск, Багратионовск, Гвардейск, and Гусев. Edges are labeled with weights: 1/1 for edges from Калининград to Светлогорск, Багратионовск, and Гусев; 1/1 for edges from Светлогорск to Гвардейск; 0/1 for the edge from Багратионовск to Гвардейск; and 1/1 for the edge from Гвардейск to Гусев. Below the graph, a section titled "Результаты работы алгоритма" has "HTML" selected over "Текст". The output text reads "Максимальный поток равен 2". At the bottom, a small note says "to://unick-soft.ru - программа является абсолютно бесплатной."

каждого из ребер в специальной форме (осуществить проверку, чтобы все веса были положительны).

**Поиск остовного дерева минимального веса** (алгоритм Краскала). Как и в предыдущей задаче данный модуль предоставляет пользователю возможность выбора способа задания длины (веса) ребер (см. предыдущую задачу).

**Подсчет числа компонент связности** и сохранение матрицы инцидентности в блочном виде.

**Поиск эйлеровых и гамильтоновых циклов и цепей.** При запуске модуля проверяется возможность поиска решения. Если граф не является эйлеровым (гамильтоновым), пользователю выдается соответствующее сообщение.

**Проверка заданного орграфа на наличие циклов.** При наличии таковых вывести каждый цикл в виде последовательности вершин циклического пути.

Результаты работы модулей отображаются на рисунке (если это возможно), а также приводится ответ в текстовом формате (записывается в указанный пользователем файл и выводится в соответствующее окно для отображения ответа).

На дальнейших стадиях развития проекта предполагается рассмотреть возможность трассировки решения задачи: вывода результатов конкретной итерации. Такой способ вывода ответа позволил бы полностью автоматизировать процесс решения задачи.

Кроме того, предполагается разработать несколько форматов для хранения данных (представление графов матрицами смежности, инцидентности, списками и пр.) и составить алгоритм выбора формата для конкретного графа.

## Дополнительные модули программы

Кроме решения задач теории графов, предполагается ориентировать программу и на решение прикладных задач математики и экономики. В начальный пакет Graph Calculator было решено включить классические прикладные задачи.

- **Модуль для решения задачи сетевого планирования.** Для графа, изображенного в области редактирования, в окне диалога задаются работы (для каждого созданного ребра пользователь может сопоставить с меткой ребра некоторое название работы) и время их выполнения. Пользователем выбираются источник (начальная вершина,  $t=0$ ) и сток (конечная вершина, окончание проекта), а также задаются ориентации ребер. Модуль рассчитывает минимальное время выполнения проекта, находит всевозможные критические пути и резервы времени по дугам, в этот путь не входящим.

- **Модуль для решения задачи о назначениях.**

- В качестве входных данных пользователю предлагается заполнить таблицу, соответствующую прибылям/убыткам, получаемыми организацией за занятость работника  $N$  на работе  $M$ . Число работников и работ совпадает. Пользователь самостоятельно выбирает, решается задача на минимум или на максимум. В качестве результата предоставляется список работников и назначенных им работ, а также общая сумма прибыли/убытков, полученных благодаря полученному распределению. В окне редактирования можно представить соответствующий двудольный граф, полученный в процессе решения. Отображение графа сделать опционным для пользователя

- **Модуль решения задачи о максимальном потоке.**

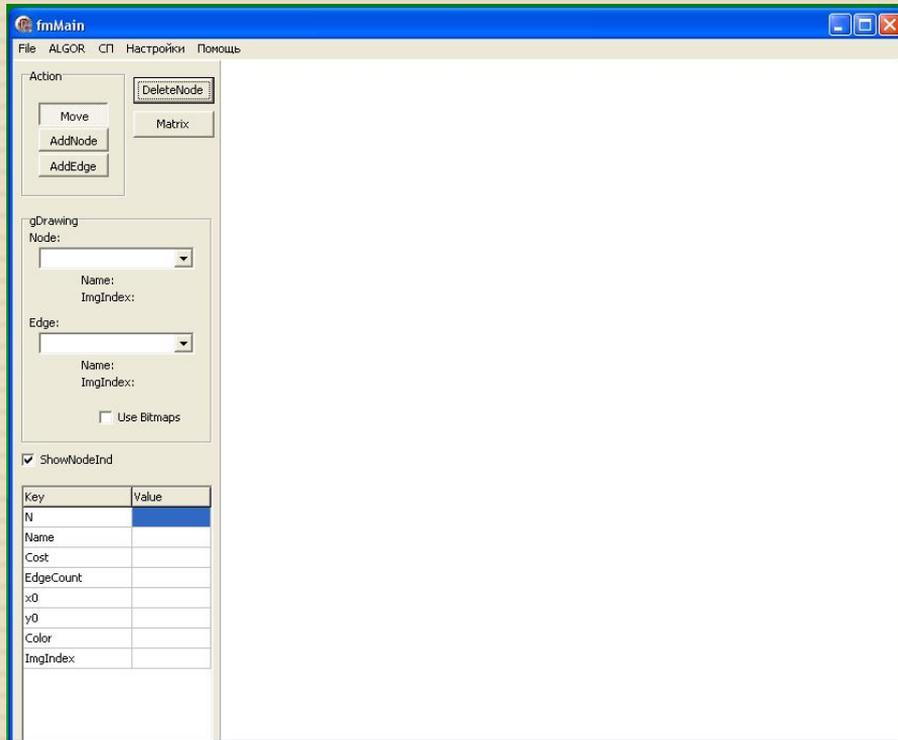
- В диалоговом окне модуля для созданного графа задаются начальная и конечная вершины (источник и сток), а также пропускные способности всех ребер. Модуль определяет максимальную пропускную способность для графа с заданными параметрами.

# Примеры неориентированных графов

<b>Граф</b>	<b>Вершины</b>	<b>Ребра</b>
Семья	Люди	Родственные связи
Город	Перекрестки	Улицы
Сеть	Компьютеры	Кабели
Домино	Костяшки	Возможность
Дом	Квартиры	Соседство
Лабиринт	Развилки и тупики	Переходы
Метро	Станции	Пересадки
Листок в клеточку	Клеточки	Наличие общей границы

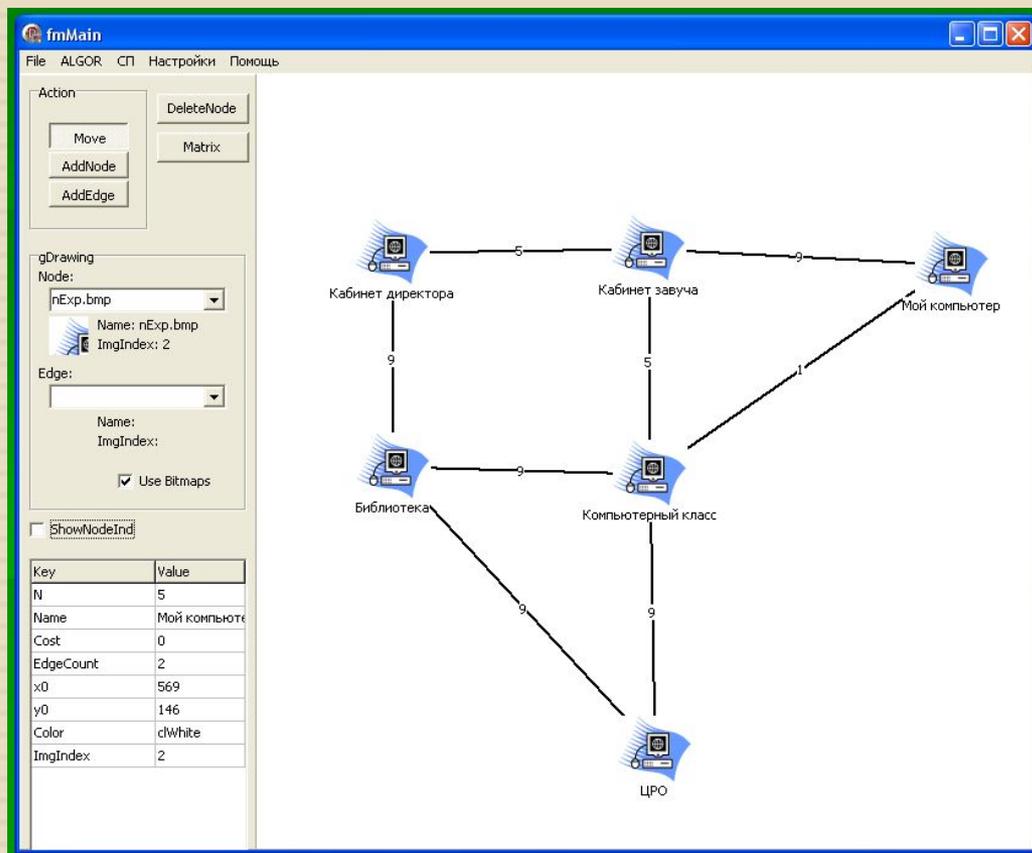
# Описание работы программы

1. Создание графа в Редакторе.
2. Применение алгоритма Дейкстры к получившемуся графу и просмотр результата.



- Вы увидите это окно.
- В данном окне вы должны ввести параметры:
- Количество вершин графа ('AddNode')
- Ребра и их вес ('AddNode', 'Matrix' – веса ребер)
- Имена вершин (ПКМ на вершине, поле 'NodeName')
- Здесь вы можете дополнительно
- выбрать графическое изображение
- вершин.

# Создание графа в Редакторе



Мы видим пример сети, оформленной в виде графа. Расстояние между вершинами показаны на линиях.

В оформлении вершин используется пиктограмма компьютера.

Для сохранения полученного графа выбираем из меню File -> Save as и сохраняем под любым именем.

# Просмотр результата

Вы увидите результат работы:  
В окне задания параметров появится строка с длиной кратчайшего пути и сам путь.

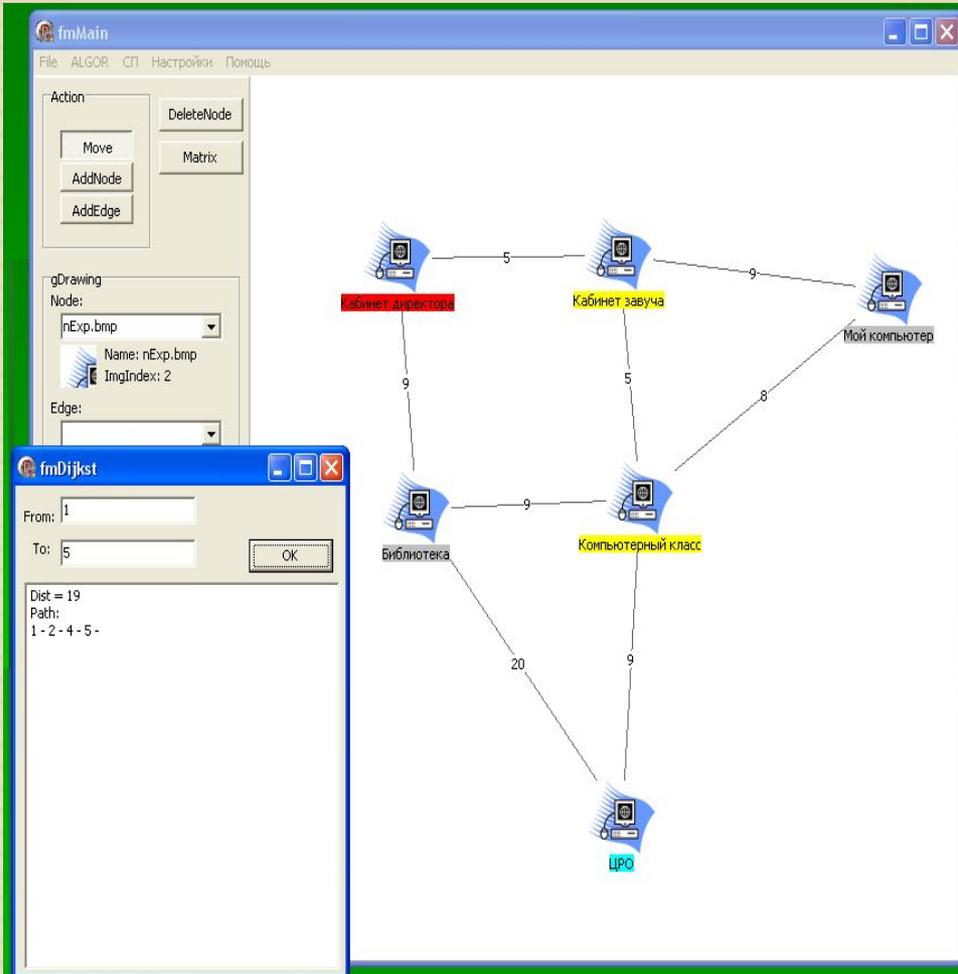
В окне редактора отобразится пройденный путь и вершины окрасятся в следующие цвета:

**Красный** – начальная вершина.

**Синий** – конечная вершина.

**Желтый** – вершины искомого пути.

**Серый** – вершины, посещенные при работе алгоритма, но не включённые в конечный путь.



# Достоинства программы

- С помощью этой программы вы можете создать любой граф с помощью удобного редактора графов: схема метро, карта городов, компьютерные сети, карту лабиринта и многое другое.
- Представить его в графическом виде, добавляя названия вершин, пиктограммы, расстояния.
- Определить кратчайший путь между двумя заданными вершинами и увидеть результат работы алгоритма в графическом и текстовом виде
- Программа была создана на языке “Delphi” с использованием объектно-ориентированного программирования.