

**Обучающая программа по теме: «Гамильтоновы циклы и задача коммивояжера»**

*Чувагина Виктория Владимировна  
Ярославский государственный педагогический университет  
им. К.Д.Ушинского  
Студент*

*Корнилов Петр Анатольевич  
Ярославский государственный педагогический университет  
им. К.Д.Ушинского  
Кандидат физико-математических наук, доцент, заведующий кафедрой  
ТиМОИ*

**Аннотация**

Была разработана обучающая программа, которая поможет обучающемуся в понимании работы алгоритма Робертса-Флореса для поиска гамильтоновых циклов в графе и метода ветвей и границ для решения задачи коммивояжера. Программа включает в себя теоретическую часть, практическую, модуль игры и модуль проверки.

**Ключевые слова:** обучающая программа, гамильтоновы циклы, алгоритм Робертса-Флореса, задача коммивояжера, метод ветвей и границ

**Training program for learning about Hamiltonian cycles and Travelling salesman problem**

*Chuvagina Viktoriya Vladimirovna  
Yaroslavl State Teacher Training University named after K.D.Ushinsky  
Student*

*Kornilov Peter Anatolevich  
Yaroslavl State Teacher Training University named after K.D. Ushinsky  
Candidate of Physical and Mathematical Sciences, Associate Professor and Head  
of the Department of Theory and Methods of Teaching Informatics*

**Abstract**

A training software program had been developed to help pupils and students learn about algorithm of Roberts and Flores in finding Hamiltonian cycles and using the branch and bound method in solving the Travelling salesman problem. The software program has several units: theory, practice, game unit and checking unit.

**Keywords:** training program, Hamiltonian cycles, algorithm of Roberts and Flores, Travelling salesman problem, branches and bound method

## **1. Введение**

Знаменитая задача коммивояжера, которая была поставлена в 1934 г., даже на сегодняшний день является одной из важнейших задач в теории графов. Появляются всё новые и новые методы решения задач такого типа, но, к сожалению, ещё не достигнут результат, который по-настоящему удовлетворит те условия, которые ставятся в настоящее время, так как точные методы требуют больших объемов вычислений, а, следовательно, и времени решения, а приближенные не гарантируют нужную точность.

## **2. Обоснование цели и задач**

Актуальность данной темы заключается в том, что задача поиска оптимального пути широко используется в транспортной логистике. Такую задачу решают, используя такие математические методы, как метод ветвей и границ, метод имитации отжига, метод эластичной сети и другие.

Переборные алгоритмы не достаточно эффективны, а успех в скорости решения каждой конкретной задачи сильно зависит от того, как будет организован перебор.

До сих пор продолжают поиски эффективных алгоритмов решения данной задачи.

Поэтому целями данного проекта являются:

- 1) Познакомить обучающихся с гамильтоновыми циклами
- 2) Пошагово объяснить алгоритм Робертса-Флореса
- 3) Познакомить обучающихся с задачей коммивояжера
- 4) Пошагово объяснить метод ветвей и границ для решения задачи коммивояжера
- 5) Провести контроль усвоенного материала

## **3. Методы достижения целей**

Для реализации поставленной цели мною были изучены алгоритмы поиска гамильтоновых циклов и решения задачи коммивояжера.

С помощью анализа, синтеза и сравнения как наиболее эффективные и понятные алгоритмы были выбраны алгоритм Робертса-Флореса для поиска гамильтоновых циклов и метод ветвей и границ для решения задачи коммивояжера.

Также, чтобы наглядно объяснить обучающимся материал, была разработана обучающая программа с теоретическим описанием, пошаговым объяснением и контролем.

## **4. Теория**

### **4.1. Нахождение гамильтоновых циклов с помощью алгоритма Робертса-Флореса**

Цикл называется гамильтоновым только при условии, что он содержит все вершины графа.

Алгоритм Робертса-Флореса представляет собой перебор с возвратом (backtracking). У нас будет создан стек просмотренных вершин. Мы берём

первую произвольную вершину и заносим её в данный стек как уже просмотренную. После этого рассматриваем смежную с ней вершину, не лежащую в нашем стеке. Если находим такую смежную вершину, то также заносим её в стек, а дальше смотрим на смежную с ней. Так продолжается до тех пор пока мы не найдём гамильтонов цикл или пока не зайдём в тупик. Если у вершины нет подходящей смежной с ней вершины, то мы убираем её из стека и пытаемся найти другой вариант пути из последней, занесённой в стек.

#### **Алгоритм нахождения гамильтоновых циклов (пошаговый):**

1.  $q$ - некоторая отправная начальная вершина.  $q$  образует первый элемент множества  $S$ , хранящего в себе уже найденные вершины строящейся цепи. К  $S$  добавляется первая вершина в столбце  $q$  матрицы  $M$  (матрица смежности).

2. Добавляем к множеству  $S$  первую возможную вершину в столбце текущей вершины. Возможная вершина не должна принадлежать  $S$ .

3. Если в столбце текущей вершины нет возможной вершины, мы возвращаемся к предыдущей рассматриваемой вершине.

4. Если найденная цепь имеет длину  $n-1$  и есть ребро между конечной и начальной вершинами, то в графе найден гамильтонов цикл  $S$ , мы возвращаемся к предыдущей вершине.

Если найденная цепь имеет длину  $n-1$  и ребра между конечной и начальной вершинами нет, то мы возвращаемся к предыдущей вершине.

5. Поиск заканчивается при условии, что множество  $S$  состоит только из начальной вершины и не существует такой возможной вершины, которую мы могли бы добавить к множеству  $S$ .

Гамильтоновы циклы, найденные к этому моменту, являются тогда всеми гамильтоновыми циклами, существующими в графе.

## **4.2. Задача коммивояжёра**

### **4.2.1. Постановка задачи**

Гамильтонову задачу о путешественнике можно преобразовать в задачу о коммивояжере. Коммивояжер – это деловой человек, который путешествует не так свободно как обычный путешественник. Он ограничен временем, денежными средствами и другими ресурсами. И, чтобы гамильтонова задача превратилась в задачу о коммивояжере, нужно каждое ребро снабдить числовой характеристикой (километраж, время на дорогу, стоимость билета, расход топлива и т.д.).

Рассмотрим задачу о коммивояжере.

Имеются  $n$  городов, расстояния (стоимость проезда, расход горючего на дорогу и т.д.) между которыми известны. Коммивояжер должен пройти все  $n$  городов по одному разу и вернуться в тот город, откуда он начинал своё путешествие. Требуется найти такой путь, при котором коммивояжёр затратит минимальное количество ресурсов.

Бесспорно, что задача коммивояжера – это задача поиска кратчайшего гамильтонова цикла в полном графе.

Если рассматривать такую схему решения данной задачи: сгенерировать все  $n!$  возможных перестановок вершин полного графа, подсчитать для каждой перестановки длину маршрута и выбрать кратчайший, то с увеличением  $n$  количество операций будет расти намного стремительней, чем  $2^n$ . Следовательно, решать задачу коммивояжера методом полного перебора практически невозможно, даже при сравнительно небольших  $n$ .

#### **4.2.2. Решение задачи методом ветвей и границ**

Метод ветвей и границ относится к методам неявного перебора, в которых делается попытка так организовать перебор, чтобы отбросить часть допустимых решений. В основе данного метода заложена идея последовательного разбиения множества допустимых решений.

На каждом шаге метода созданные подмножества рассматриваются на то, содержат ли они оптимальное решение. Проверка в основном осуществляется сравнением нижней оценки значения целевой функции с верхней оценкой функционала (рекордом).

Если оценка сверху меньше или равна оценке снизу целевой функции, то рассматриваемое подмножество не содержит решения лучше рекорда и отбрасывается. Если рекордное значение больше значения целевой функции, то происходит смена рекорда. Если просмотрены все элементы разбиения, алгоритм завершает работу, а текущий рекорд является оптимальным решением. Иначе среди непросмотренных элементов выбирается такое множество, которое является самым «перспективным». Оно разбивается по определенной схеме, а новые подмножества оцениваются по схеме, которая описана выше. Процесс продолжается до тех пор, пока не будут просмотрены все элементы разбиения [4].

### **5. Результат**

В результате работы над данной темой была разработана программа.

Описание программы.

Когда мы запускаем программу, мы видим название проекта и в меню наверху список тех действий, которые в данный момент доступны.

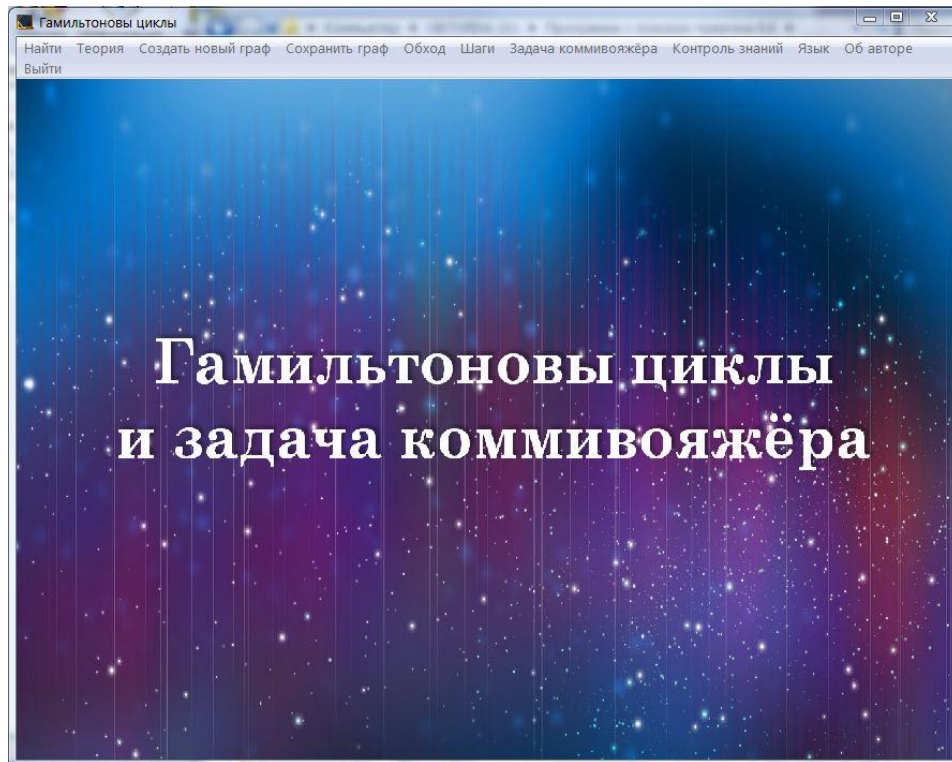


Рисунок 1 – Главный экран

Мы можем начать работу с графом сразу или для начала прочитать теорию.

В программе существует несколько способов задания графа: создать граф самому, выбрать из уже имеющихся или ввести матрицу смежности.

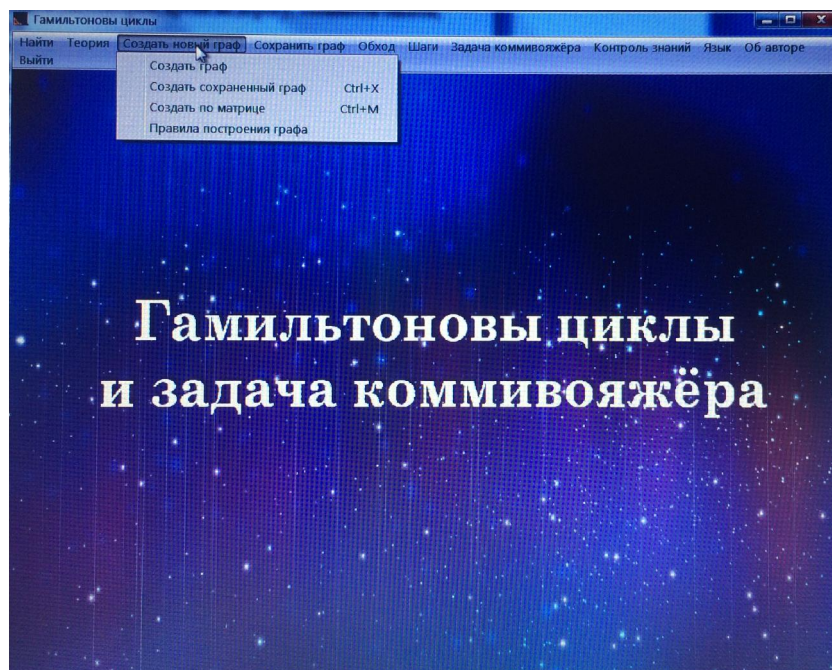


Рисунок 2 – Варианты создания графа

При построении графа первый раз вам всплывает подсказка о том, как строить граф в данной программе, а затем вы сможете в любой момент их

посмотреть нажав на «Создать новый граф»-«Правила построения графа» на русском или «Create new graph»-«Rules» на английском.

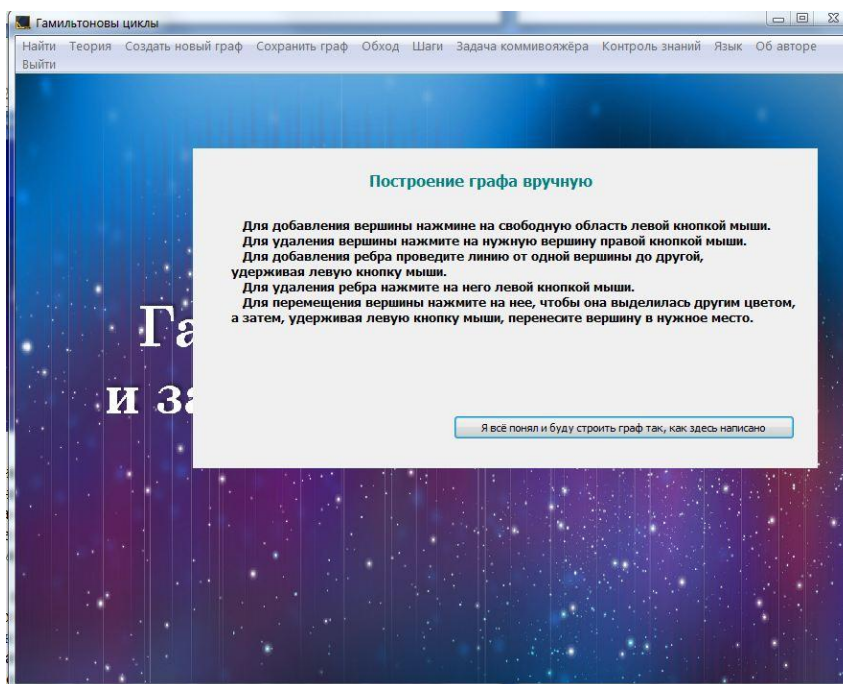


Рисунок 3 – Правила построения графа

Если нужно, можно загрузить уже готовые графы и выбрать способ задания веса рёбер.



Рисунок 4 – Сохраненные графы

Так же можно воспользоваться способом создания графа по матрице смежности, которую нужно вводить в нужное поле.



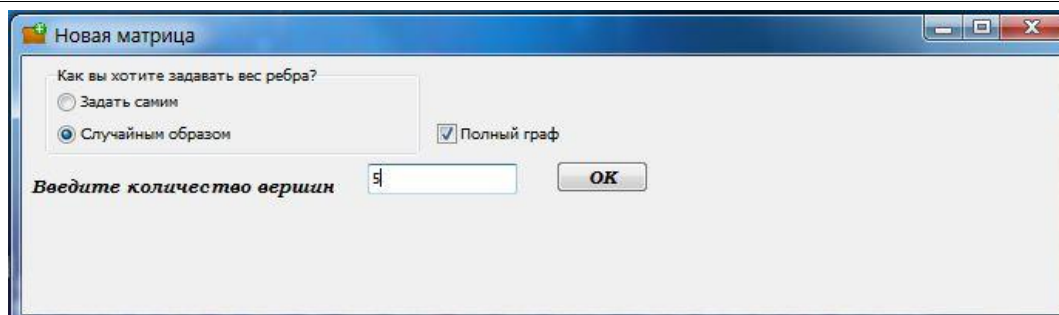


Рисунок 5 – Выбор количества вершин графа

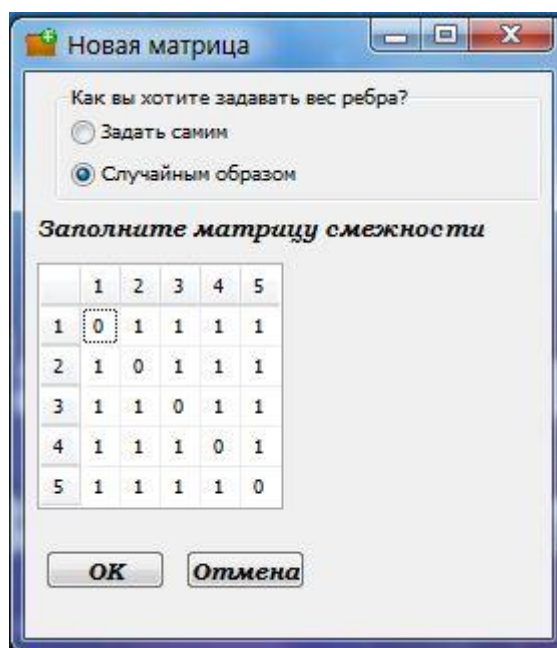


Рисунок 6 – Заполнение матрицы смежности

После того, как мы создали граф, мы можем приступить к алгоритмам. Для того, чтобы начать алгоритм Робертса-Флореса, нужно нажать на «Обход»/ «Start algorithm» в меню. После нажатия появляются элементы, необходимые для объяснения алгоритма: рабочий стек (слева от графа); таблица, в которой представлен список вершин и под каждой столбец вершин, смежных с ней, в порядке возрастания (внизу формы); кнопка следующего шага (*кнопка предыдущего шага появится после выполнения хотя бы одного шага*); кнопка для показа результата («Результат»/ «Result»); список уже найденных циклов и кнопка, с помощью которой мы можем исправить граф.

При выполнении алгоритма пошагово под графом пишутся пояснения о проделанном шаге, на самом графе красным выделяются ребра, которые входят в путь на данном шаге, в таблице выделяются желтым те вершины, которые на данный момент используются в продолжении. Если вернуться назад, то все элементы изменятся под этот шаг.

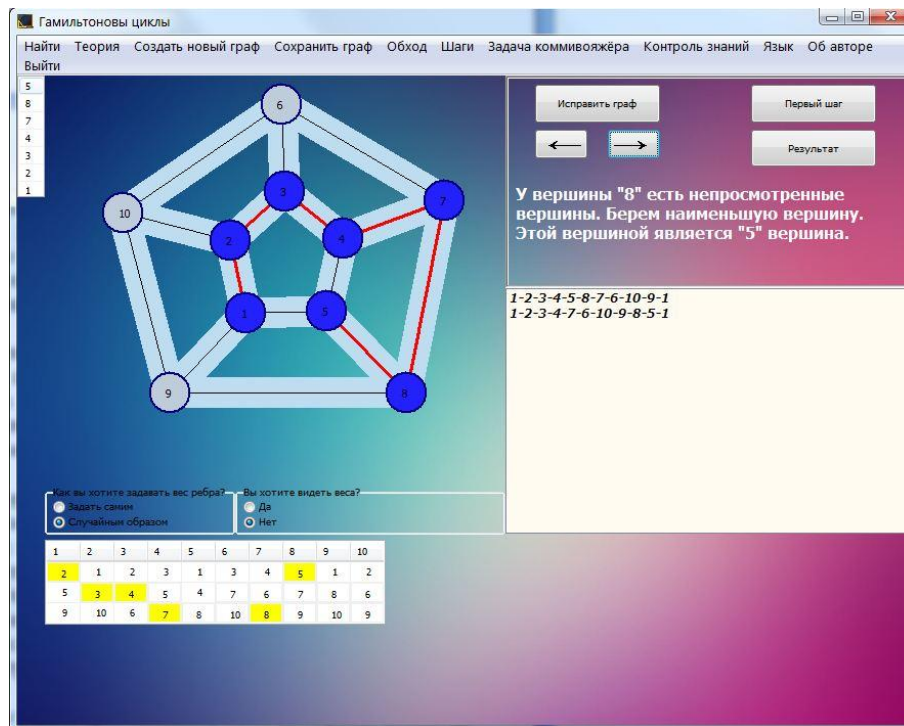


Рисунок 7 – Пример работы пошагового алгоритма Робертса-Флореса

При нажатии на кнопку «Исправить граф»/ «Change graph» во время выполнения обхода, появится предупреждение о том, что можно поправить граф, но обход при этом начнётся с начала. Можно согласиться исправить граф или отказаться.

На любом шаге алгоритма можно начать его с начала, нажав на кнопку «Первый шаг»/ «The first step». После нажатия на кнопку «Результат»/ «Result» появляется список всех гамильтоновых циклов. При нажатии на них в графе выделяется этот путь и в нижнем левом углу графа появляется сам путь и его вес.

Если мы хотим узнать кратчайший путь, то нам нужно нажать на «Задача коммивояжёра»/ «TSP» на панели меню.

После этого нужно построить граф или воспользоваться тем, с которым мы уже работали.



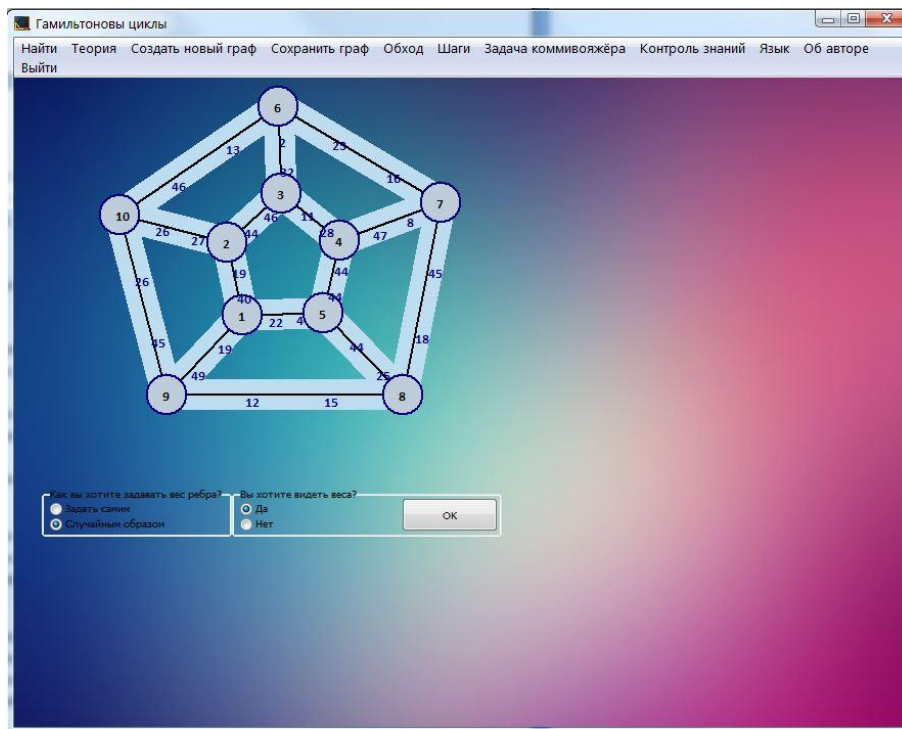


Рисунок 8 – Построение для задачи коммивояжера

После нажатия на кнопку «ОК»/ «ОК» откроется новая форма, в которую перенесется матрица.

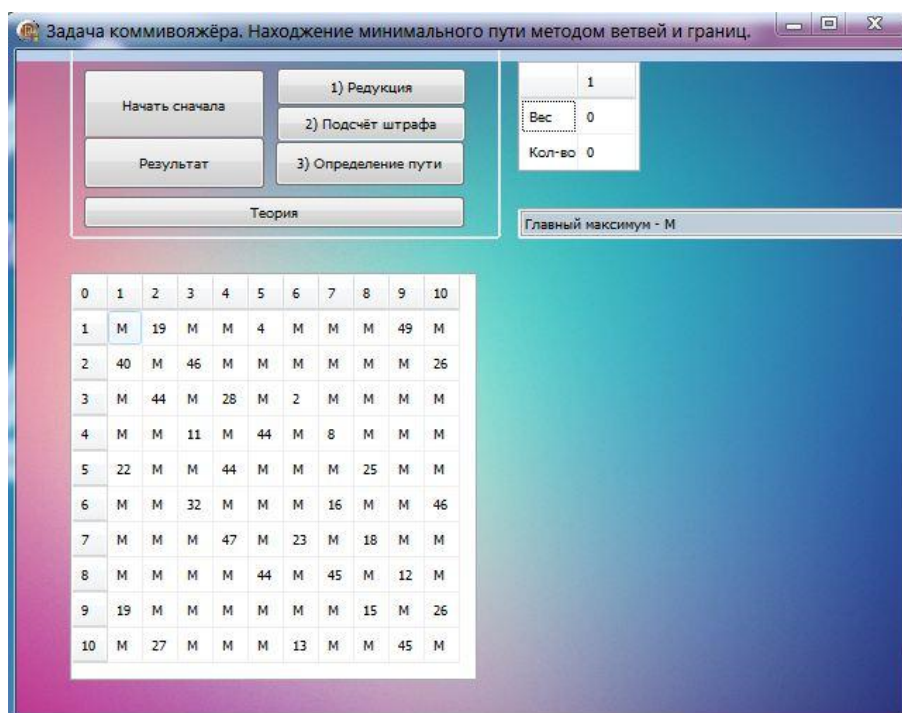


Рисунок 9 – Модуль задачи коммивояжера

Можно воспользоваться просмотром по шагам, для этого нужно поочередно нажимать на шаги, в программе установлено, что если ты не выполнил 1-ый шаг, то на 2-ой ты нажать не сможешь и т.д. Если нажать на первый шаг происходит редукция матрицы.

При нажатии на второй шаг происходит подсчёт штрафа.

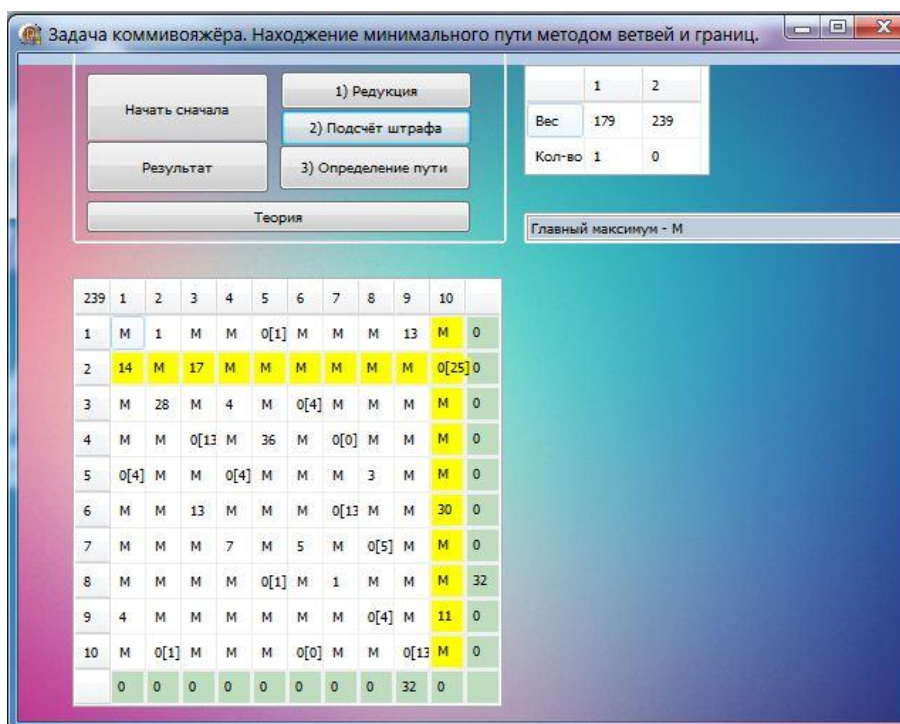


Рисунок 10 – Демонстрация работы первых двух шагов в модуле коммивояжера

При нажатии на третий шаг происходит разделение матрицы на две (в одной мы используем данное ребро, в другой- нет), так же наверху формы идет подсчёт матриц и их весов.

Также в любой момент можно начать алгоритм сначала, нажав на кнопку «Начать сначала»/ «Start».

После завершения алгоритма выводится ответ.

После подсчёта минимального пути, при закрытии формы, этот путь будет выделяться в списке путей и на графе тоже можно будет его увидеть.

Можно проверить правильность решения, пройдясь по всем путям, и тем самым убедиться, что мы нашли минимальный путь.

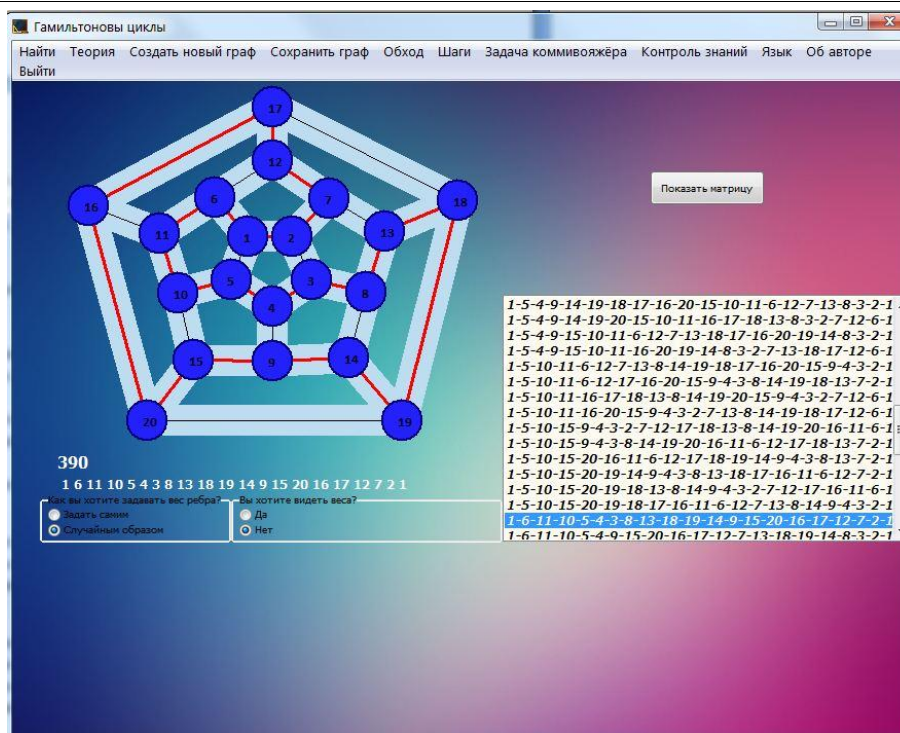


Рисунок 11 – Вывод результата в задаче коммивояжера

В модуле «Контроль»/ «Test» пользователю предлагается ответить на 5 вопросов, которые каждый раз генерируются со случайными данными (т.е. пользователь не сможет записать все правильные ответы, а потом просто вводить их в поле). По окончании тестирования ученику показывают его результат в процентах, а на почту преподаватель приходит полный отчёт, в котором содержатся фамилия и имя, введённые пользователем, правильные ответы на вопросы, вариант ученика и время, за которое был дан ответ на каждый вопрос.

При нажатии на «Об авторе»/ «About author» Вы увидите информацию о создателе программы.

Также имеется кнопка для прохождения контроля по данной теме. В отдельном окне открываются задания, на которые пользователь должен дать ответ, используя полученные в данной программе знания. После прохождения всех заданий обучающемуся выводят его отметку, а для преподавателя в отдельном документе сохраняется подробный отчёт.

## 6. Заключение

Данная программа сделана для того, чтобы помочь разобраться в алгоритме Робертса-Флореса о нахождении гамильтоновых циклов и в решения задачи коммивояжера, то есть задачи о нахождении кратчайшего цикла, с помощью метода ветвей и границ. Так как в программе можно использовать собственные графы, смотреть действия на каждом шаге, а в алгоритме Робертса-Флореса ещё вернуться на несколько шагов назад, программа сможет дать очень подробную информацию о данных алгоритмах.

**Библиографический список**

1. Кристофидес Н. Теория графов. Алгоритмический подход. 1978. 249 с.
2. Roberts S.M., Flores B. An engineering approach to the traveling salesman problem // Man. Sci.. 1967. №13. С. 269.
3. Roberts S.M., Flores B. Systematic generation of Hamiltonian circuits // Comm. of ACM. 1966. №9. С. 690.
4. Гончаров Е. Н., Ерзин А. И., Залюбовский В. В. Исследование операций. Примеры и задачи. Новосибирск: Изд-во НГУ, 2005.