Учреждение образования

«Белорусский государственный технологический университет»

**Отчет по лабораторным работам по дисциплине**

**“Математическое программирование”**

Выполнил:

Студент 2 курса 4 группы ФИТ

Корнелюк Валентин Владимирович

2023 г.

**Лабораторная работа №1**

**Вспомогательные функции**

**Цель работы:** приобретение навыков составления и отладки программ с использованием пользовательских функций для замера продолжительности процесса вычисления.

**Задание 1**

Разработайте три функции (start, dget и iget), используя следующие спецификации. Реализация представлена на рисунке 1.

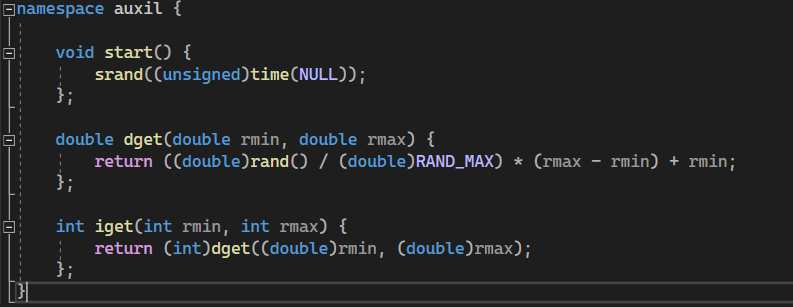


Рисунок 1 – Реализация функций

**Задание 2**

1.Реализовать пример 2.

2.Для проверки работоспособности разработанных функций и приобретения навыков замера продолжительности процесса вычисления реализуйте программу, приведенную в примере 2.

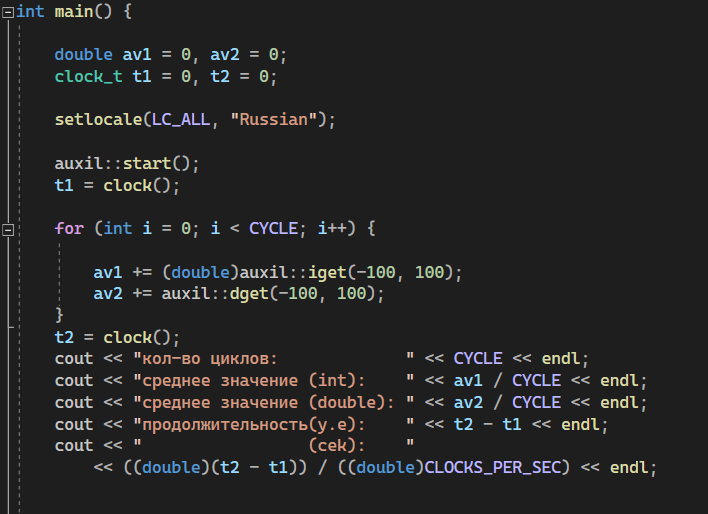


Рисунок 2 – вызов функции и измерение времени выполнения

**Задание 3**

Проведите необходимые эксперименты (разработать кодом) и постройте график зависимости (Excel) продолжительности процесса вычисления от количества циклов в примере 2.

Примечание: продолжительность вычисления измерять в условных единицах процессорного времени (функция clock). График представлен на рисунке 3

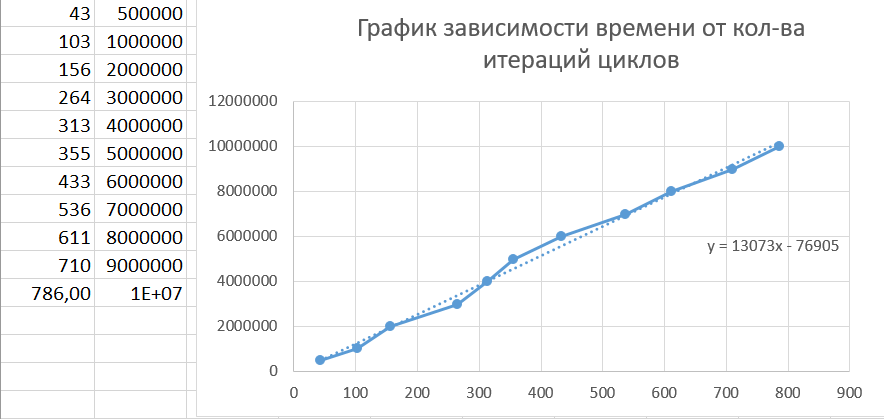


Рисунок 3 – график зависимости

Реализация Факториала в коде демонстрируется на рис. 4

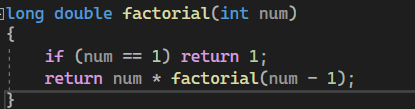


Рисунок 4 – функция для нахождения факториала

Вызов и подсчет времени для реализации факториала представлен на рисунке 5:

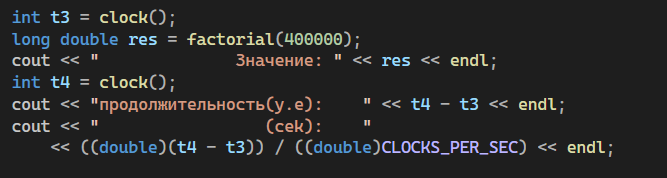


Рисунок 5 – вызов функции для подсчёта факториала

График зависимости:

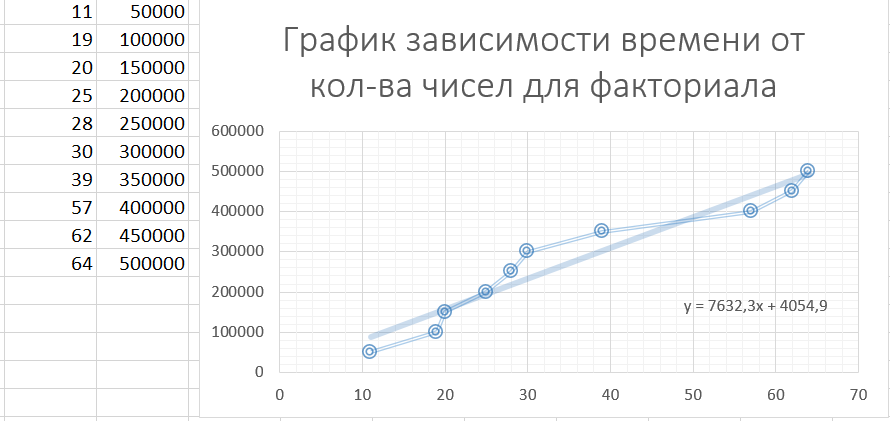


Рисунок 6 – график зависимости

**Вывод**: по полученным измерениям, можем сделать следующий вывод, что время выполнения программы линейно зависит от количества циклов или значения факториала.

**Лабораторная работы №2**

**Комбинаторные алгоритмы решения оптимизационных задач**

**ЦЕЛЬ РАБОТЫ:** приобрести навыки разработки генераторов подмножеств, перестановок, сочетаний и размещений на С++; научиться применять разработанные генераторы для решения задач о рюкзаке (упрощенную, коммивояжера, об оптимальной загрузке судна и об оптимальной загрузке судна с центровкой.

**Задание 1.** Разобрать и разработать генератор подмножеств заданного множества.

Ниже на рисунках представлены заголовочные файлы и файлы cpp с реализацией поставленной задачи.

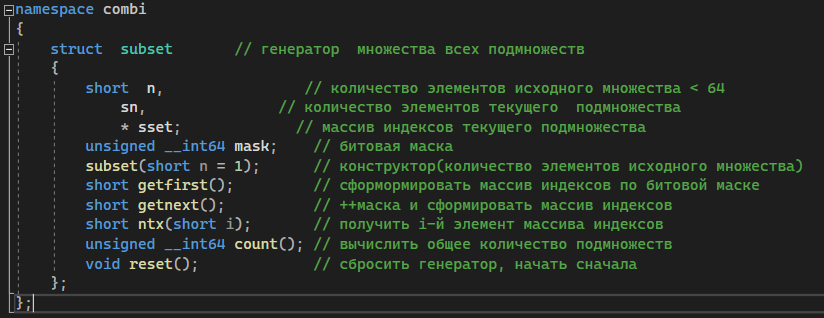


Рисунок 1 – Combi.h

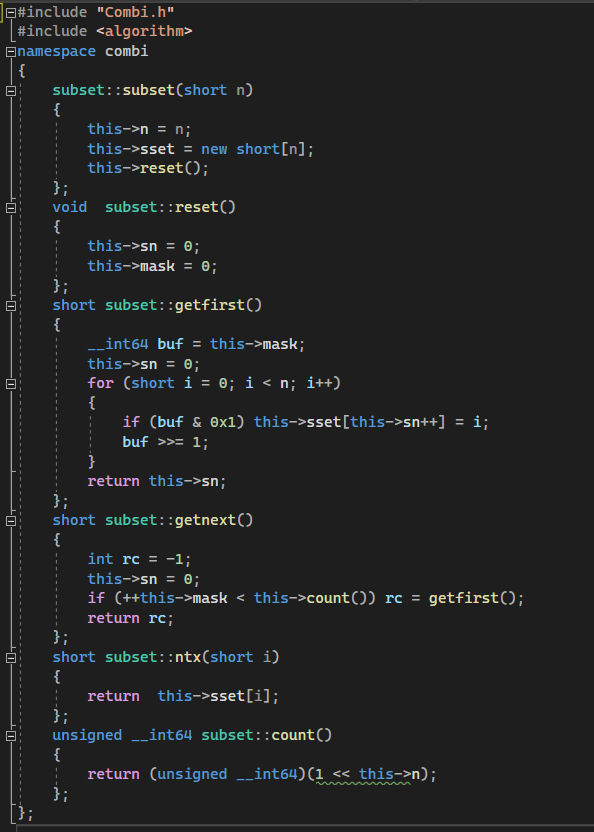


Рисунок 2 - Combi.cpp

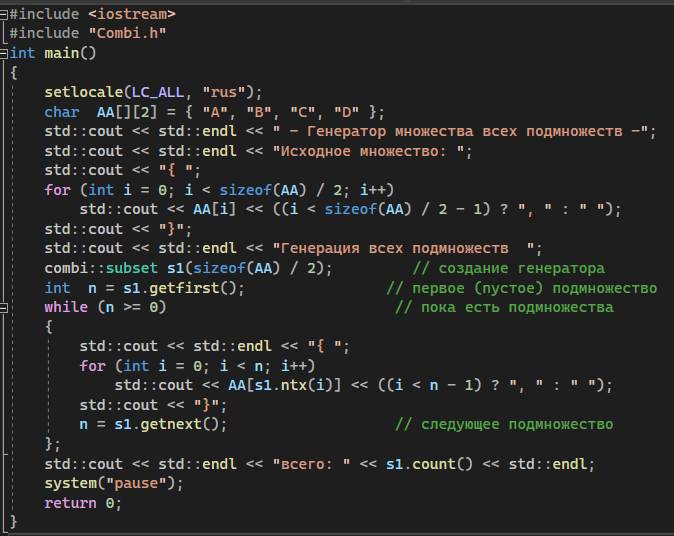


Рисунок 3 - main.cpp

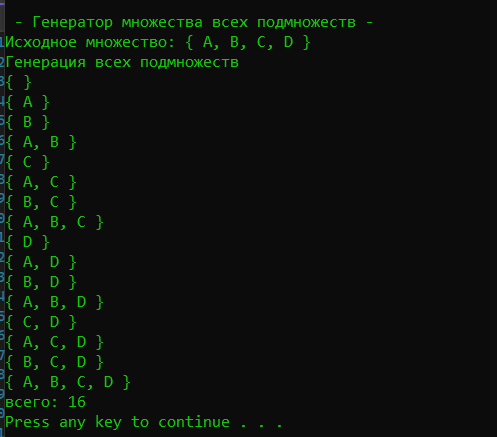


Рисунок 4 – Результат выполнения

**Задание 2.** Разобрать и разработать генератор сочетаний.

Ниже на рисунках представлены заголовочные файлы и файлы cpp с реализацией поставленной задачи.

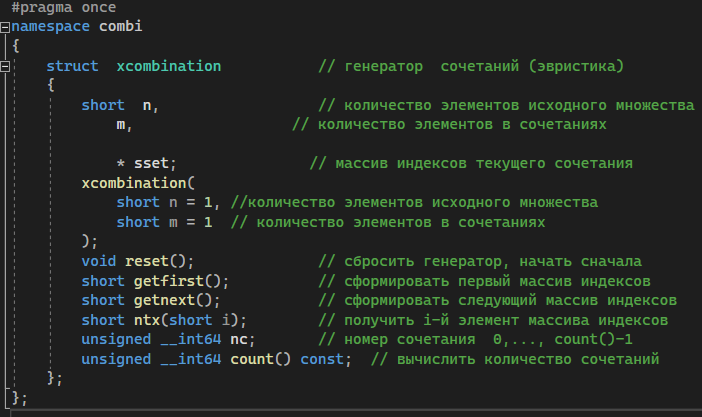


Рисунок 5 - Combi.h

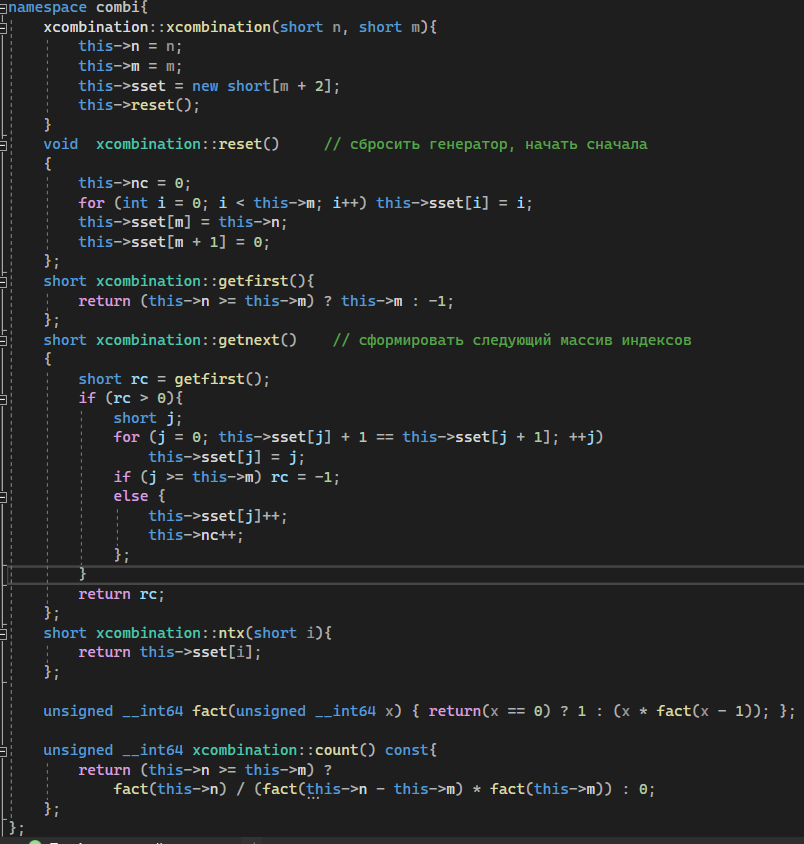


Рисунок 6 - Combi.cpp

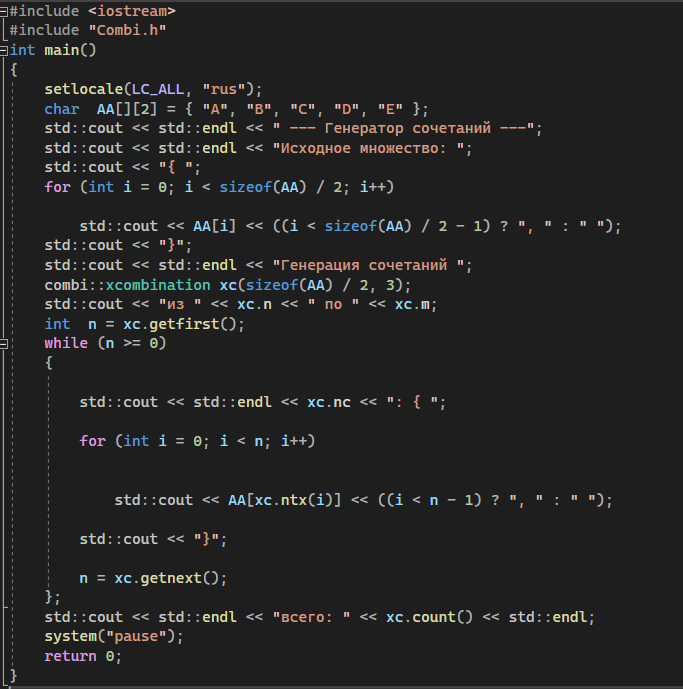


Рисунок 7 - main.cpp

Результат выполнения представлен на рисунке 8.

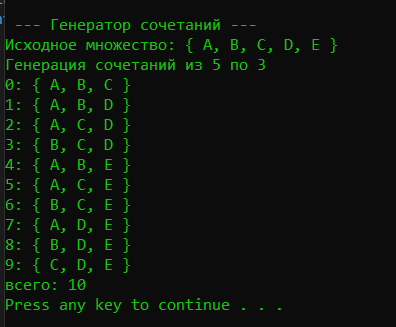


Рисунок 8 – результат выполнения программы

**Задание 3.** Разобрать и разработать генератор перестановок.

Ниже на рисунках представлены заголовочные файлы и файлы cpp с реализацией поставленной задачи.

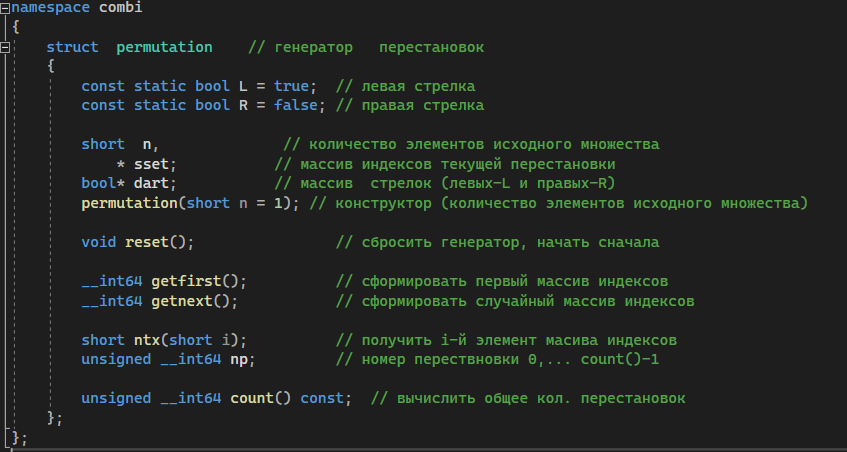


Рисунок 9 – Combi.h

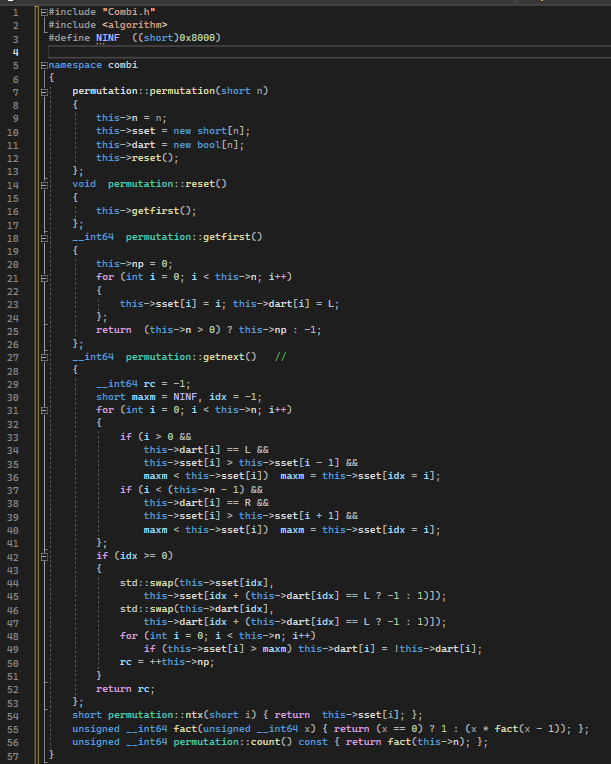


Рисунок 10 – Combi.cpp

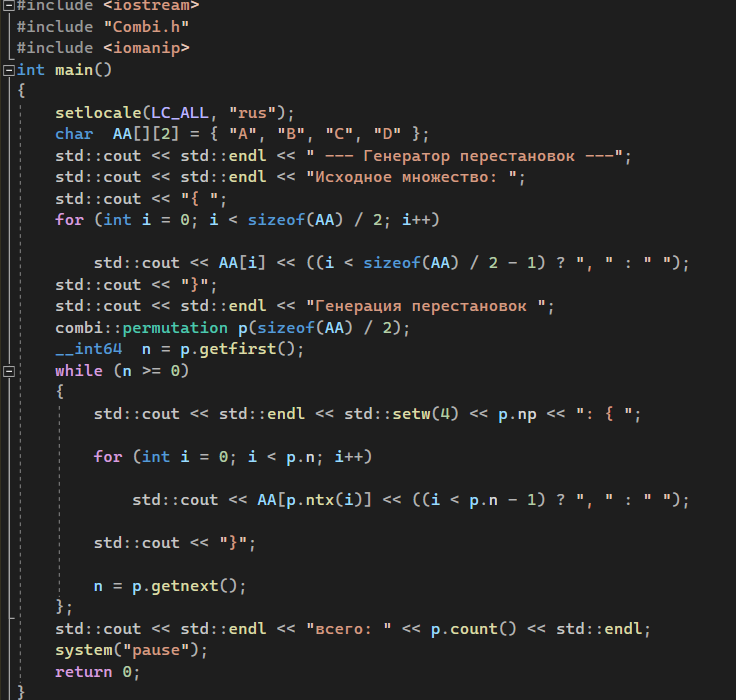


Рисунок 11 – main.cpp

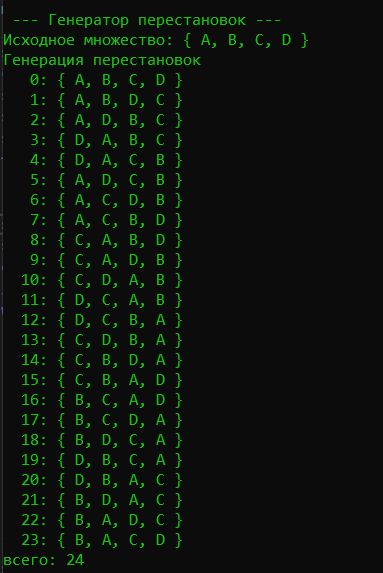


Рисунок 12 – результат выполнения программы

**Задание 4.** Разобрать и разработать генератор размещений.

Ниже на рисунках представлены заголовочные файлы и файлы cpp с реализацией поставленной задачи.

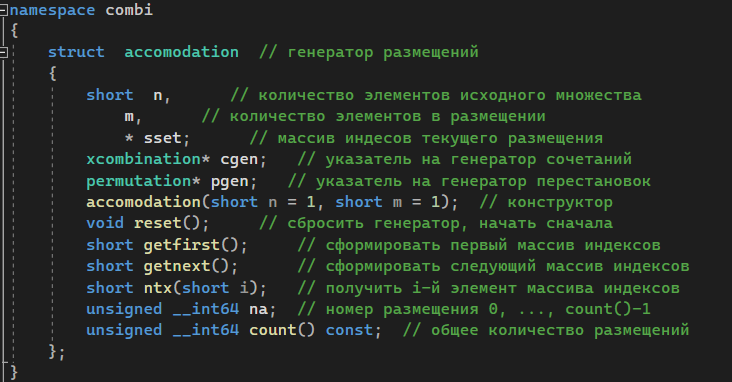


Рисунок 13 – Combi.h



Рисунок 14 – Combi.cpp

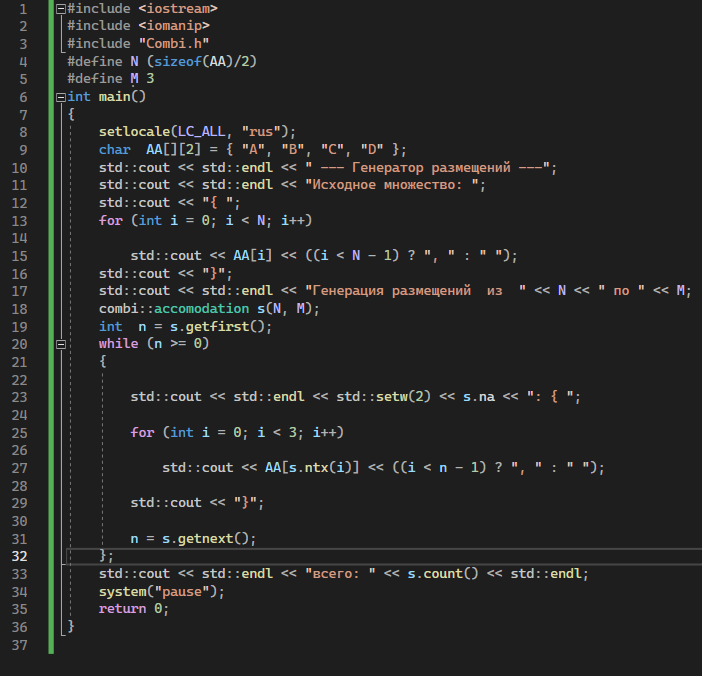


Рисунок 15 – main.cpp

Результат выполнения программы представлен на рисунке 16.

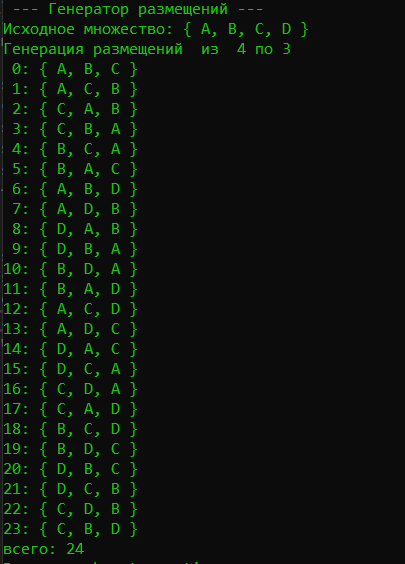


Рисунок 16 - результат выполнения программы

**Задание 5.** Решить в соответствии с вариантом задачу и результат занести в отчет. Решить задачу коммивояжера (расстояния сгенерировать случайным образом: 10 городов, расстояния 10 – 300 км, 3 расстояния между городами задать бесконечными);



Рисунок 17 – схема решения задачи

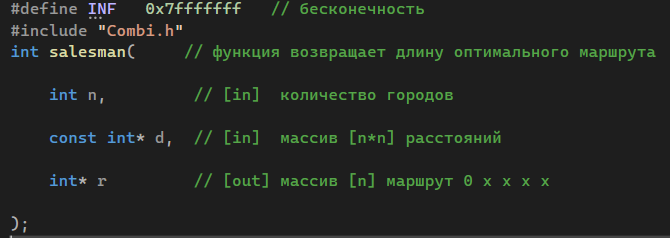


Рисунок 17 – Salesman.h

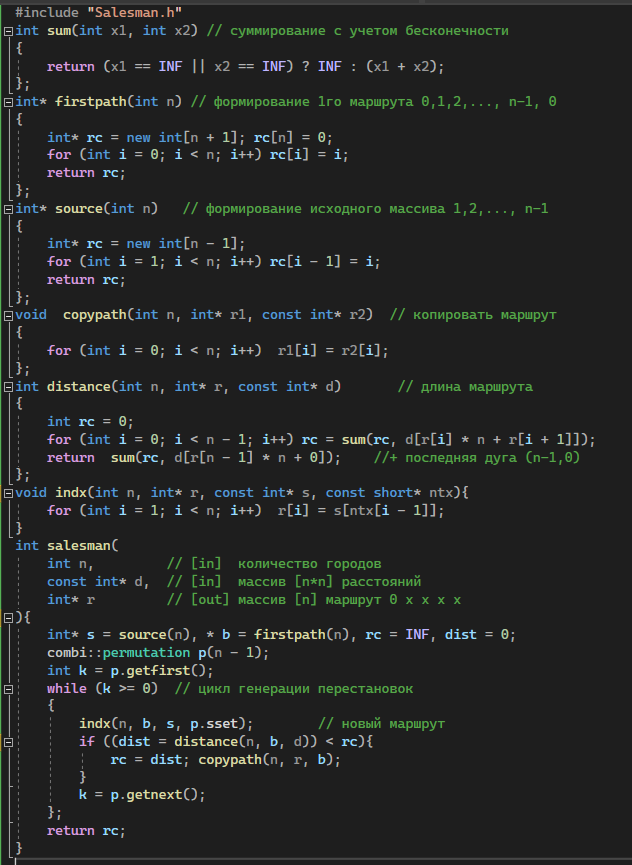


Рисунок 18 – Salesman.cpp

Результат выполнения программы представлен на рисунке 19.

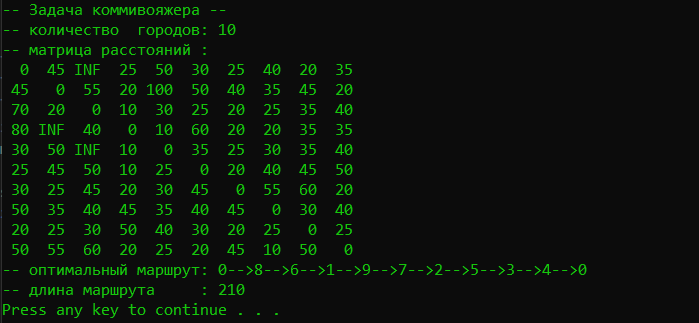


Рисунок 19 – результат работы программы

**Задание 6.** Исследовать зависимость времени вычисления необходимое для решения задачи от размерности задачи и результат в виде графика:

Результат работы программы представлен на рисунке 20

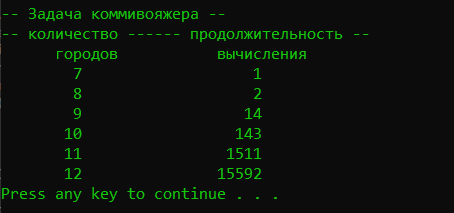


Рисунок 20 – Зависимость времени выполнения от количества городов

На рисунке 21 представлен график зависимости времени вычисления от количества городов.

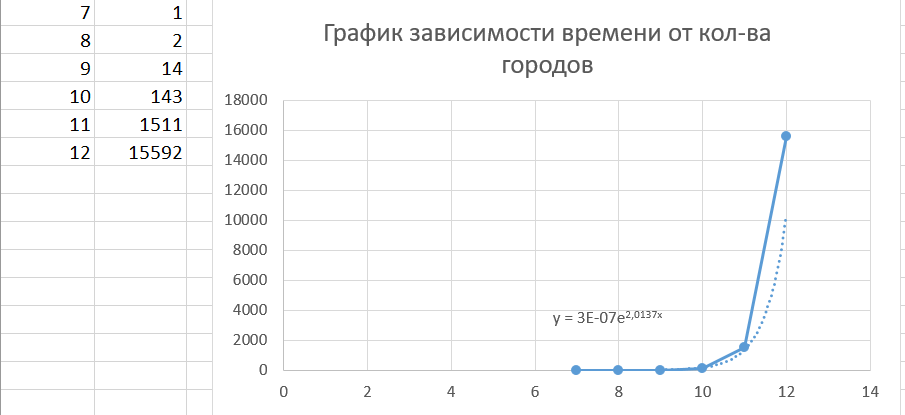


Рисунок 21 – график зависимости

**Вывод**: исходя из полученных данных и графика, можно заметить, что скорость выполнения программы экспоненциально возрастает при увеличении числа городов.

**Лабораторная работы №3**

**Метод ветвей и границ. Задача коммивояжера и методы её решения.**

**ЦЕЛЬ РАБОТЫ:** освоить общие принципы решения задач методом ветвей и границ, решить задачу о коммивояжере данным методом, сравнить полученное решение задачи с комбинаторным методом перестановок.

**Задание 1.** Сформулировать условие задачи коммивояжера с параметром.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | **1** | **2** | **3** | **4** | **5** |
| **1** | INF | 18 | 30 | INF | 9 |
| **2** | 9 | INF | 24 | 59 | 75 |
| **3** | 11 | 27 | INF | 86 | 58 |
| **4** | 26 | 49 | 36 | INF | 27 |
| **5** | 84 | 75 | 52 | 22 | INF |

Решение:

Для определения нижней границы множества воспользуемся **операцией редукции** или приведения матрицы по строкам, для чего необходимо в каждой строке матрицы D найти минимальный элемент.  
di = min(j) dij

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **4** | **5** | di |
| **1** | INF | 18 | 30 | INF | 9 | 9 |
| **2** | 9 | INF | 24 | 59 | 75 | 9 |
| **3** | 11 | 27 | INF | 86 | 58 | 11 |
| **4** | 26 | 49 | 36 | INF | 27 | 26 |
| **5** | 84 | 75 | 52 | 22 | INF | 22 |

Затем вычитаем di из элементов рассматриваемой строки.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **4** | **5** |
| **1** | INF | 9 | 21 | INF | 0 |
| **2** | 0 | INF | 15 | 50 | 66 |
| **3** | 0 | 16 | INF | 75 | 47 |
| **4** | 0 | 23 | 10 | INF | 1 |
| **5** | 62 | 53 | 30 | 0 | INF |

Такую же операцию редукции проводим по столбцам, для чего в каждом столбце находим минимальный элемент:  
dj = min(i) dij

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **4** | **5** |
| **1** | INF | 9 | 21 | INF | 0 |
| **2** | 0 | INF | 15 | 50 | 66 |
| **3** | 0 | 16 | INF | 75 | 47 |
| **4** | 0 | 23 | 10 | INF | 1 |
| **5** | 62 | 53 | 30 | 0 | INF |
| dj | 0 | 9 | 10 | 0 | 0 |

После вычитания минимальных элементов получаем полностью редуцированную матрицу, где величины di и dj называются **константами приведения**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **4** | **5** |
| **1** | INF | 0 | 11 | INF | 0 |
| **2** | 0 | INF | 5 | 50 | 66 |
| **3** | 0 | 7 | INF | 75 | 47 |
| **4** | 0 | 14 | 0 | INF | 1 |
| **5** | 62 | 44 | 20 | 0 | INF |

Сумма констант приведения определяет нижнюю границу H:  
H = 9+9+11+26+22+0+9+10+0+0 = 96

**Определяем ребро ветвления** и разобьем все множество маршрутов относительно этого ребра на два подмножества (i,j) и (i\*,j\*).  
С этой целью для всех клеток матрицы с нулевыми элементами заменяем поочередно нули на INF и определяем для них сумму образовавшихся констант приведения, они приведены в скобках.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **4** | **5** | di |
| **1** | INF | 0(7) | 11 | INF | 0(1) | 0 |
| **2** | 0(5) | INF | 5 | 50 | 66 | 5 |
| **3** | 0(7) | 7 | INF | 75 | 47 | 7 |
| **4** | 0(0) | 14 | 0(5) | INF | 1 | 0 |
| **5** | 62 | 44 | 20 | 0(70) | INF | 20 |
| dj | 0 | 7 | 5 | 50 |  |  |

Наибольшая сумма констант приведения равна (20 + 50) = 70 для ребра (5,4), следовательно, множество разбивается на два подмножества (5,4) и (5\*,4\*).

**Исключение ребра** (5,4) проводим путем замены элемента d54 = 0 на INF, после чего осуществляем очередное приведение матрицы расстояний для образовавшегося подмножества (5\*,4\*), в результате получим редуцированную матрицу.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **4** | **5** | di |
| **1** | INF | 0 | 11 | INF | 0 | 0 |
| **2** | 0 | INF | 5 | 50 | 66 | 0 |
| **3** | 0 | 7 | INF | 75 | 47 | 0 |
| **4** | 0 | 14 | 0 | INF | 1 | 0 |
| **5** | 62 | 44 | 20 | INF | INF | 20 |
| dj | 0 | 0 | 0 | 50 | 0 | 70 |

Нижняя граница гамильтоновых циклов этого подмножества:  
H(5\*,4\*) = 96 + 70 = 166

**Включение ребра** (5,4) проводится путем исключения всех элементов 5-ой строки и 4-го столбца, в которой элемент d45 заменяем на М, для исключения образования негамильтонова цикла.  
В результате получим другую сокращенную матрицу (4 x 4), которая подлежит операции приведения.  
После операции приведения сокращенная матрица будет иметь вид:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **5** | di |
| **1** | INF | 0 | 11 | 0 | 0 |
| **2** | 0 | INF | 5 | 66 | 0 |
| **3** | 0 | 7 | INF | 47 | 0 |
| **4** | 0 | 14 | 0 | INF | 0 |
| dj | 0 | 0 | 0 | 0 | 0 |

Нижняя граница подмножества (5,4) равна:  
H(5,4) = 96 + 0 = 96 ≤ 166

Поскольку нижняя граница этого подмножества (5,4) меньше, чем подмножества (5\*,4\*), то ребро (5,4) включаем в маршрут с новой границей H = 96.

**Определяем ребро ветвления**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **5** | di |
| **1** | INF | 0(7) | 11 | 0(47) | 0 |
| **2** | 0(5) | INF | 5 | 66 | 5 |
| **3** | 0(7) | 7 | INF | 47 | 7 |
| **4** | 0(0) | 14 | 0(5) | INF | 0 |
| dj | 0 | 7 | 5 | 47 |  |

max: d(1,5)=47

**Исключение ребра** (1,5): d15=INF.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | **5** | di |
| **1** | INF | 0 | 11 | INF | 0 |
| **2** | 0 | INF | 5 | 66 | 0 |
| **3** | 0 | 7 | INF | 47 | 0 |
| **4** | 0 | 14 | 0 | INF | 0 |
| dj | 0 | 0 | 0 | 47 | 47 |

H(1\*,5\*) = 96 + 47 = 143

**Включение ребра** (1,5): d51=М.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | di |
| **2** | 0 | INF | 5 | 0 |
| **3** | 0 | 7 | INF | 0 |
| **4** | 0 | 14 | 0 | 0 |
| dj | 0 | 7 | 0 | 7 |

H(1,5) = 96 + 7 = 103 ≤ 143

Ребро (1,5) включаем в маршрут с новой границей H=103.

**Определяем ребро ветвления**.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | di |
| **2** | 0(5) | INF | 5 | 5 |
| **3** | 0(0) | 0(7) | INF | 0 |
| **4** | INF | 7 | 0(12) | 7 |
| dj | 0 | 7 | 5 |  |

max: d(4,3)=12.

**Исключение ребра** (4,3): d43=INF.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i j | **1** | **2** | **3** | di |
| **2** | 0 | INF | 5 | 0 |
| **3** | 0 | 0 | INF | 0 |
| **4** | INF | 7 | INF | 7 |
| dj | 0 | 0 | 5 | 12 |

(4\*,3\*) = 103 + 12 = 115

**Включение ребра** (4,3): d34=INF.

|  |  |  |  |
| --- | --- | --- | --- |
| i j | **1** | **2** | di |
| **2** | 0 | INF | 0 |
| **3** | 0 | 0 | 0 |
| dj | 0 | 0 | 0 |

H(4,3) = 103 + 0 = 103 ≤ 115

Ребро (4,3) включаем в маршрут с новой границей H=103.  
В соответствии с этой матрицей включаем в гамильтонов маршрут ребра (2,1) и (3,2).  
В результате по дереву ветвлений гамильтонов цикл образуют ребра:  
(5,4), (4,3), (3,2), (2,1), (1,5),  
Длина маршрута = 103

Граф решения представлен на рисунке 3.1

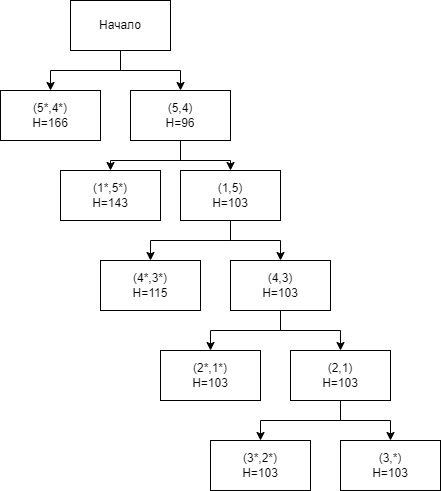


Рисунок 3.1 – граф решения

Результат решения задачи с помощью генератора перестановок представлен на рисунке 3.2

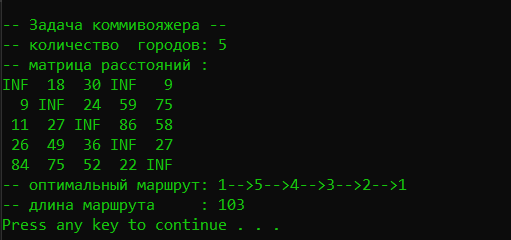


Рисунок 3.1 – результат решения с помощью генератора перестановок

**ВЫВОД**: были освоены общие принципы решения задач методом ветвей и границ, решена задача о коммивояжере данным методом, полученное решение было сравнено с комбинаторным методом перестановок.