Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Дисциплина «Математическое программирование»

Отчёт по лабораторным работам

Студент: Скалкович С.Л.

ФИТ 2 курс 1 группа

.

Минск 2023

Содержание

[Лабораторная работа 1 3](#_Toc135741366)

[Лабораторная работа 2 6](#_Toc135741367)

[Лабораторная работа 3 18](#_Toc135741368)

[Лабораторная работа 4 23](#_Toc135741369)

[Лабораторная работа 5 34](#_Toc135741370)

# Лабораторная работа 1

**Задание 1**. Разработайте три функции (start, dget и iget). Три функции будут представлены на рисунке 1.1.

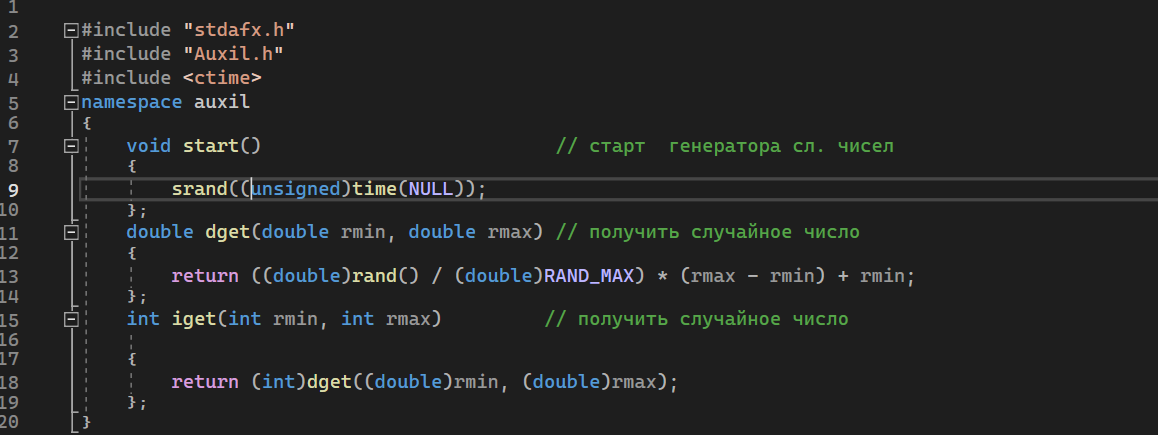


Рисунок 1.1 – Три функции

**Задание 2** будет представлена на Рисунке 1.2

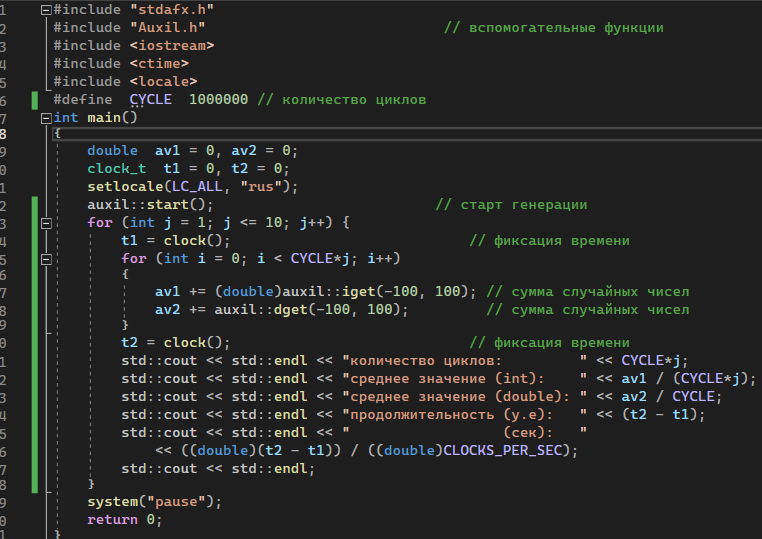


Рисунок 1.2 – Три функции

Вывод данных будет на рисунке 1.3.

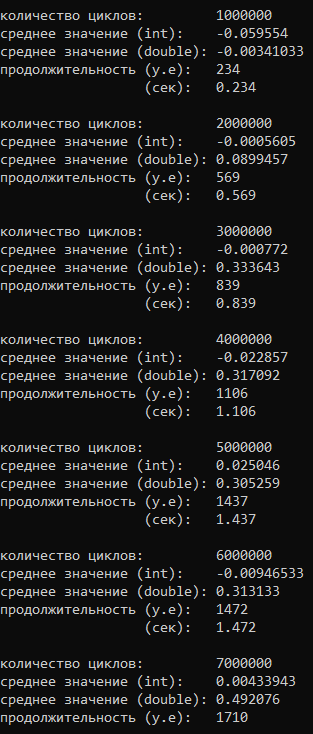


Рисунок 1.3 – Вывод данных

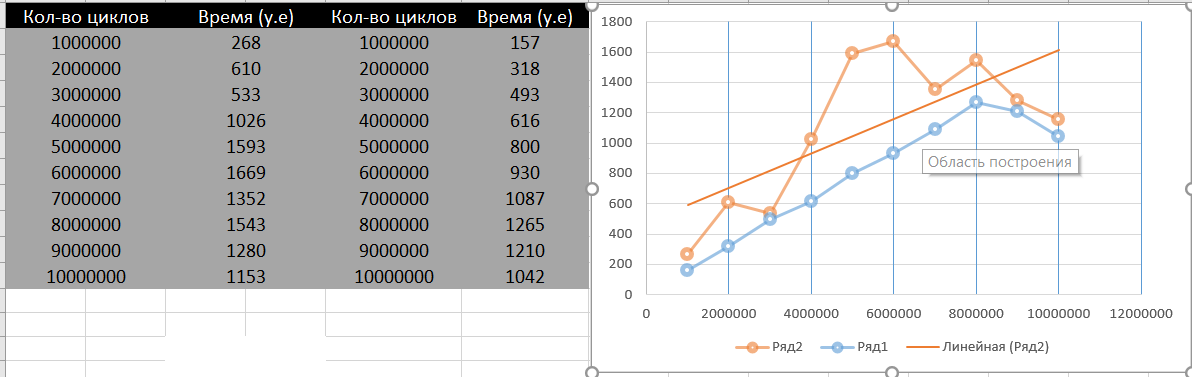
Результаты измерений и соответствующий график приведены ниже на рисунке-1.4.

Рисунок 1.4 – Результаты измерений

Критерий согласия Стьюдента был вычислен с помощью формулы =СТЬЮДЕНТ.ТЕСТ(A2:B11;C2:D11;1;1),полученный результат приведен на рисунке 1.5.



Рисунок 1.5 – Критерий согласия Стьюдента

Коэффициент корреляции был вычислен с помощью формулы = =КОРРЕЛ(A2:B11;C2:D11) полученный результат приведен на рисунке 1.6.



Рисунок 1.6 – Коэффициент корреляции

Значение коэффициента корреляции близко к единице. Это означает, что точки корреляционного поля линейно зависимы.

**Вывод:** скорость выполнения программы линейно зависит от количества итераций цикла.

# Лабораторная работа 2

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

**Задание 1:** Разработать генератор подмножеств заданного множества. Генератор подмножеств заданного множества будет на рисунке 2.1.

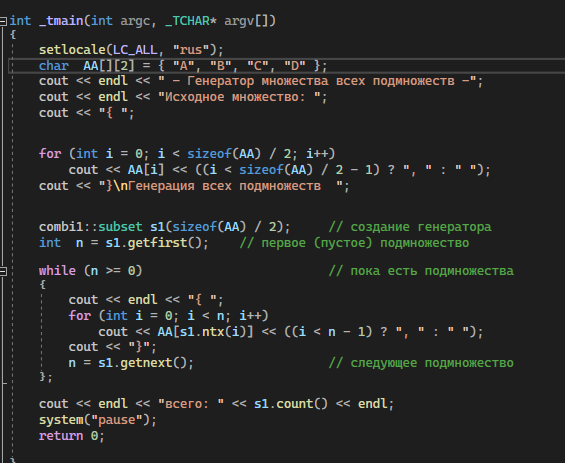


Рисунок 2.1 – Генератор подмножеств

Результат выполнения будет представлен на рисунке 2.2.

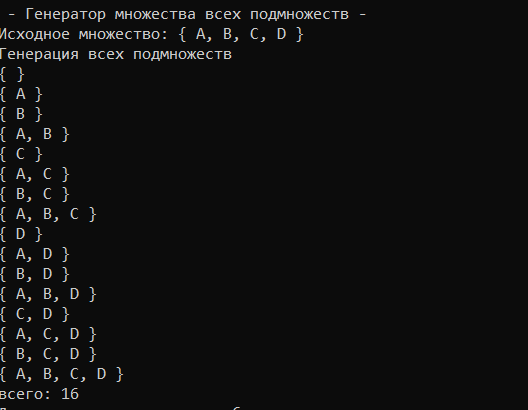


Рисунок 2.2 – Генератор подмножеств заданного множества

Алгоритм будет представлен на рисунке 2.3.



Рисунок 2.3 – Генератор подмножеств заданного множества

**Задание 2:** Разработать генератор сочетаний. Код будет представлен на рисунке 2.4.

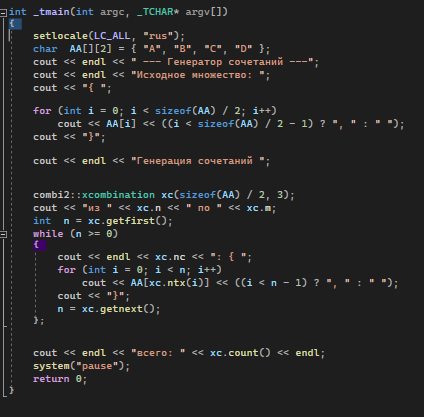


Рисунок 2.4 – Код генератора сочетаний

Результат выполнения будет представлен на рисунке 2.5.

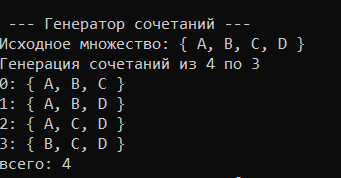


Рисунок 2.5 –Результат генератора сочетаний

Алгоритм будет представлен на рисунке 2.6

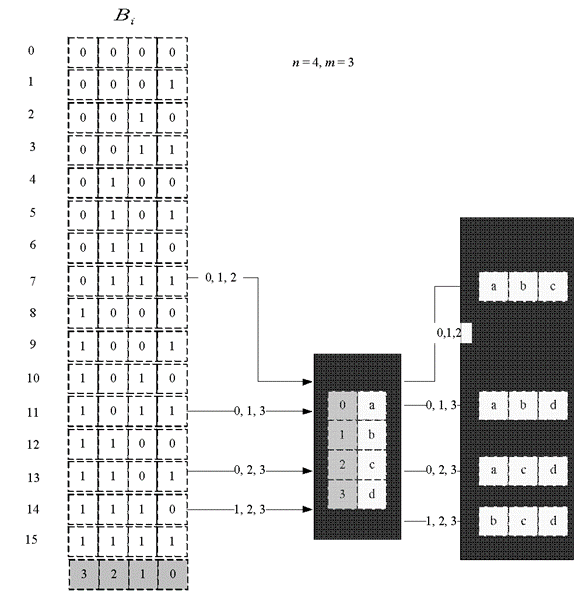


Рисунок 2.6 – Генератор сочетаний

**Задание 3:** Разработать генератор перестановок. Код будет представлен на рисунке 2.7.

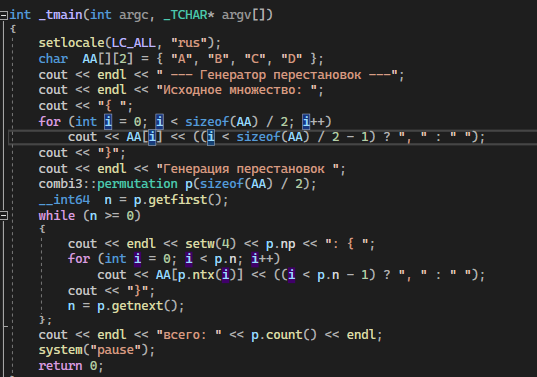


Рисунок 2.7 – Код генератора перестановок

Результат выполнения будет представлен на рисунке 2.8.

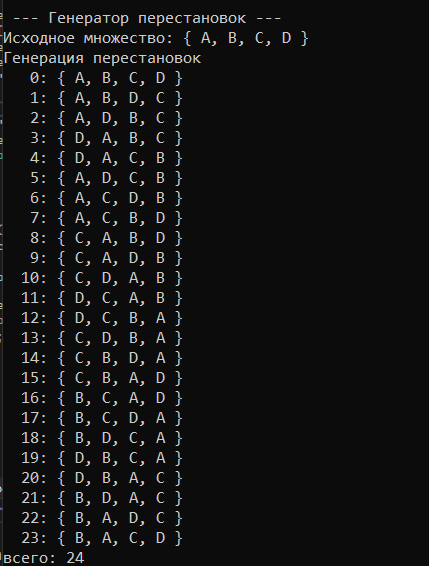


Рисунок 2.8 –Результат генератора перестановок

Алгоритм будет представлен на рисунке 2.9



Рисунок 2.9 – Генератор перестановок

**Задание 4:** Разработать генератор размещений. Код будет представлен на рисунке 2.10.

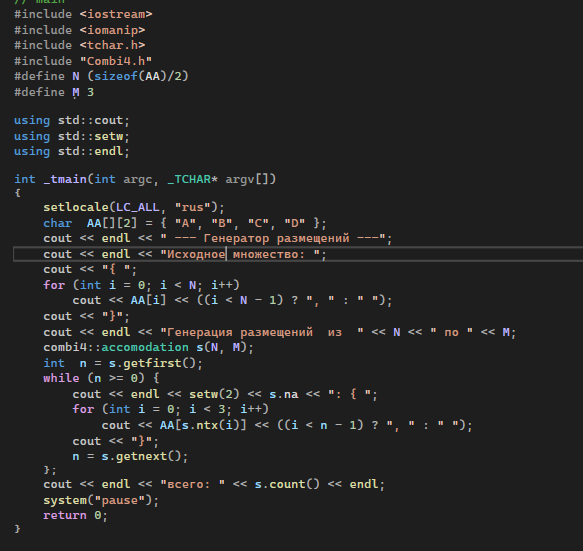


Рисунок 2.10 – Код генератора размещений

Результат выполнения будет представлен на рисунке 2.11.

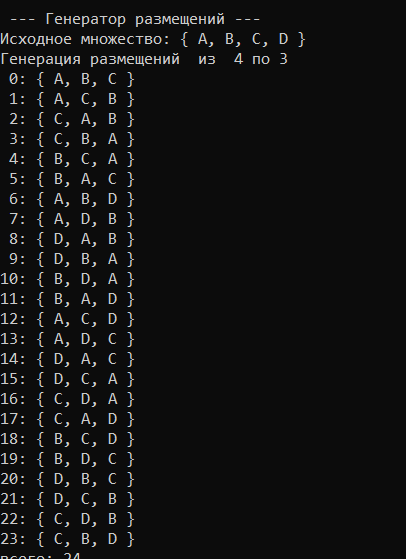


Рисунок 2.11 –Результат генератора размещений

Алгоритм будет представлен на рисунке 2.12.



Рисунок 2.12– Генератор размещений

**Задание 5:** 1, 5, 9, 13) коммивояжера (расстояния сгенерировать случайным образом: 10 городов, расстояния 10 – 300 км, 3 расстояния между городами задать бесконечными). Код этого задания будет на рисунке 2.13.

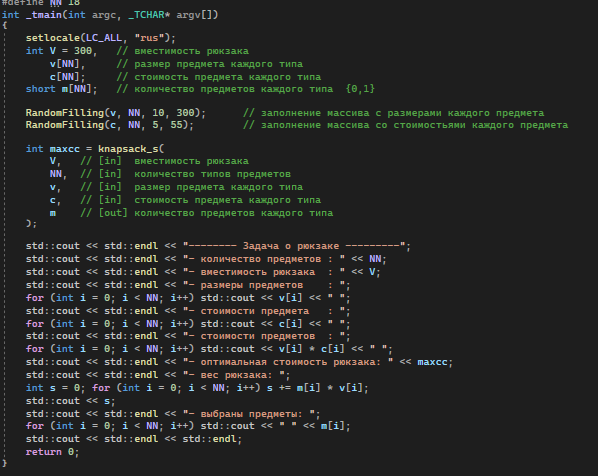


Рисунок 2.13 – Код задания 5

Результат выполнения будет представлен на рисунке 2.14.

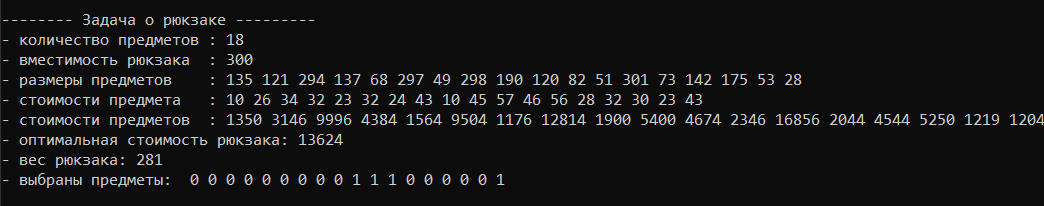


Рисунок 2.14 –Результат задания 5

**Задание 6:** Исследовать зависимость времени вычисления необходимое для решения задачи (в соответствии с вариантом) от размерности задачи и результат в виде графика с небольшим пояснением занести в отчет: (1, 5, 9, 13) коммивояжера (6 – 12 городов)



Рисунок 2.15 – Код задания 6

Результат выполнения будет представлен на рисунке 2.16.

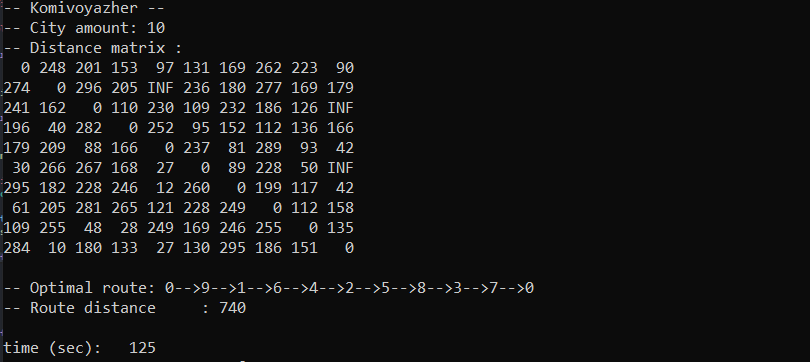


Рисунок 2.16–Результат задания 6

График будет представлен на рисунке 2.17.



Рисунок 2.17 – График

Алгоритм будет представлен на рисунке 2.18.

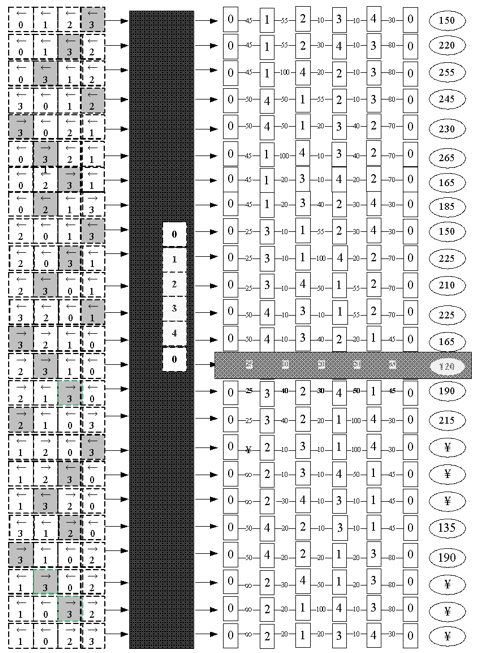


Рисунок 2.18– Схема решения задачи коммивояжера

**Дополнительное задание:** Решить задачу об оптимальной загрузке судна на основе генератора сочетаний.



Рис. 2.19 - Схема решения задачи об оптимальной нагрузке судна.



Рис. 2.20 – Код дополнительного задания.

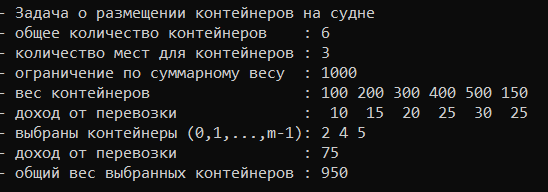


Рис 2.21 – Результат выполнения задания.

**Вывод:** приобретены навыки разработки генераторов подмножеств, перестановок, сочетаний и размещений на С++; разработано решение задачи коммивояжера и задачи о размещения судна, построена зависимость.

# Лабораторная работа 3

**Условие:**

Найти оптимальный маршрут для коммивояжера, если известно, что кол-во городов равно 5, а расстояние между городами задается следующей матрицей d:

где *n* – номер варианта;

n=9.

Задачу следует решить с использованием метода ветвей и границ. Матрица будет представлена на рисунке 3.1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 |
| 0 | inf | 18 | 30 | inf | 9 |
| 1 | 9 | inf | 24 | 59 | 75 |
| 2 | 11 | 27 | inf | 86 | 58 |
| 3 | 26 | 49 | 36 | inf | 27 |
| 4 | 84 | 75 | 52 | 22 | inf |

Рисунок 3.1– Матрица

Находим минимальное значение в каждой строке и выписываем его в отдельный столбец: Производим редукцию строк.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | min |
| 0 | inf | 18 | 30 | inf | 9 | 9 |
| 1 | 9 | inf | 24 | 59 | 75 | 9 |
| 2 | 11 | 27 | inf | 86 | 58 | 11 |
| 3 | 26 | 49 | 36 | inf | 27 | 26 |
| 4 | 84 | 75 | 52 | 22 | inf | 22 |

Находим минимальные значения в каждом столбце. Эти минимумы выписываем в отдельную строку.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 |
| 0 | inf | 9 | 21 | inf | 0 |
| 1 | 0 | inf | 15 | 50 | 66 |
| 2 | 0 | 16 | inf | 75 | 47 |
| 3 | 0 | 23 | 10 | inf | 1 |
| 4 | 62 | 53 | 30 | 0 | inf |
| min | 0 | 9 | 10 | 0 | 0 |

В дальнейшем для каждой нулевой клетки получившейся преобразованной матрицы находим «оценку». Полученную оценку записываем рядом с нулем, в скобках. Считаем нижнюю границу.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 |
| 0 | inf | 0 | 11 | inf | 0 |
| 1 | 0 | inf | 5 | 50 | 66 |
| 2 | 0 | 7 | inf | 75 | 47 |
| 3 | 0 | 14 | 0 | inf | 1 |
| 4 | 62 | 44 | 20 | 0 | inf |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | min |
| 0 |  | 0(7) |  |  | 0(1) | 9 |
| 1 | 0(5) |  |  |  |  | 9 |
| 2 | 0(7) |  |  |  |  | 11 |
| 3 | 0(0) |  | 0(5) |  |  | 26 |
| 4 |  |  |  | 0(70) |  | 22 |
| min | 0 | 9 | 10 | 0 | 0 |  |

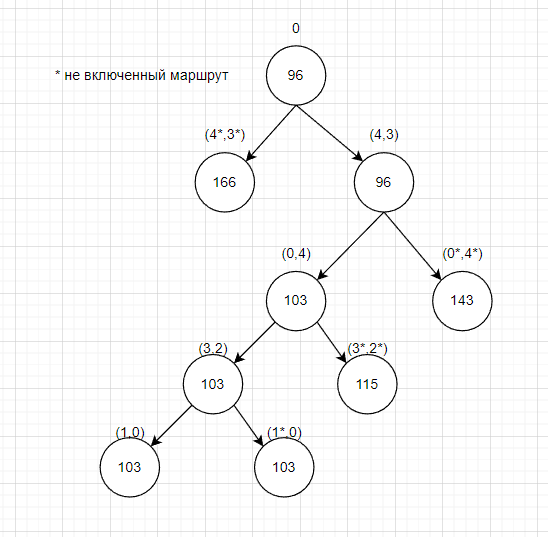
Нижняя граница H0= 0+9+10+0+0+9+9+11+26+22=96

Далее мы следуем по алгоритму:

1. Вычисление оценок нулевых клеток — считаем оценки (pij) для каждой ячейки с нулями, как сумму минимумов по строке и столбцу, в которых располагается нулевая клетка, не учитывая при этом саму нулевую клетку;
2. Выбор нулевой клетки с максимальной оценкой — ищем среди нулевых клеток обладающую наибольшей оценкой (если таких ячеек несколько, выбираем любую), и получаем пару ветвей (вариантов) решения задачи: с включением в маршрут отрезка пути относящегося к выбранной ячейке и без включения;
3. Редукция матрицы — вычеркиваем относящиеся к выбранной клетке строку и столбец, а также заменяем значение ячейки соответствующей обратному пути на M;
4. Вычисление нижней границы первой ветви (включающей отрезок пути) — вновь находим минимумы по строкам, проводим редукцию строк, находим минимумы по столбцам, проводим редукцию столбцов, после чего вычисляем локальную нижнюю границу, как сумму предыдущей локальной нижней границы и минимумов (Hk = Hk-1 + ∑di + ∑dj), и добавляем вершину в граф;
5. Вычисление нижней границы второй ветви (не включающей отрезок пути) — считаем локальную нижнюю границу, как сумму предыдущей локальной нижней границы и оценки выбранной ранее нулевой клетки (Hk\* = Hk-1 + pij), и добавляем вершину в граф;
6. Выбор ветви с минимальным значением нижней границы — среди еще не ветвившихся вершин выбираем обладающую минимальным значением локальной нижней границы (вне зависимости от того, какую ветвь рассматриваем в данный момент);
7. Если полный маршрут еще не найден, продолжаем решение, если найден — переходим к пункту 10 — если маршрут еще не найден, то ход дальнейшего решения зависит от выбранной ветви: (а) первая ветвь — переходим к пункту 1, (б) вторая ветвь — в клетку с максимальной оценкой ставим M и производим нахождения минимума по столбцам и строкам, а также редукцию матрицы, впоследствии находим корневую нижнюю границу, (в) другая ветвь — возвращаемся к соответствующим ей этапу решения и таблице данных;
8. Построение полного маршрута и определение его длины — соединяем все найденные ранее отрезки пути в полный маршрут и считаем его общую длину (данные берем из исходной таблицы).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Нижняя граница H0=96 | | | | | |  |
|  |  |  |  |  |  |  |  |
|  | Включаем маршрут 0(70) | | |  |  |  | Не включаем |
|  | 0 | 1 | 2 | 4 |  |  |  |
| 0 | inf | 0 | 11 | 0 | 0 |  | 96+70=166 |
| 1 | 0 | inf | 5 | 66 | 0 |  |  |
| 2 | 0 | 7 | inf | 47 | 0 |  |  |
| 3 | 0 | 14 | 0 | inf | 0 |  |  |
|  | 0 | 0 | 0 | 0 | 0+96 |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | 0 | 1 | 2 | 4 |  |  |  |
| 0 | inf | 0(7) | 11 | 0(47) | 0 |  | 96+47=143 |
| 1 | 0(5) | inf | 5 | 66 | 0 |  |  |
| 2 | 0(7) | 7 | inf | 47 | 0 |  |  |
| 3 | 0(0) | 14 | 0(5) | inf | 0 |  |  |
|  | 0 | 0 | 0 | 0 | 0+96 |  |  |
|  |  |  |  |  |  |  |  |
|  | 0 | 1 | 2 |  |  |  |  |
| 1 | 0(5) | inf | 5 | 0 |  |  |  |
| 2 | 0(7) | 7 | inf | 0 |  |  |  |
| 3 | inf | 14 | 0(5) | 0 |  |  |  |
|  | 0 | 7 | 0 | 96+7=143 |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | 0 | 1 |  |  |  |  |  |
| 1 | 0 | inf | 0 |  |  |  | 103+12=115 |
| 2 | 0 | 0 | 0 |  |  |  |  |
|  | 0 | 0 | 103+0 |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | 0 | 1 |  |  |  |  | 103+0=103 |
| 1 | 0(0) | inf | 0 |  |  |  |  |
| 2 | 0(0) | 0(0) | 0 |  |  |  |  |
|  | 0 | 0 | 103+0 |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | 0 |  |  |  |  |  |  |
| 1 | 0(0) | 0 |  |  |  |  |  |
|  | 0 | 103+0 |  |  |  |  |  |

Для визуального анализа построим дерево решения:



**В результате по дереву ветвлений гамильтонов цикл образуют ребра:**

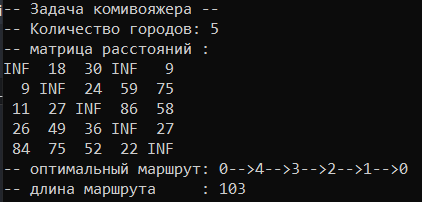
(4,3), (3,2), (2,1), (1,0), (0,4),

**Оптимальный маршрут:**

**0->4->3->2->1->0**

**Длина оптимального маршрута: φ=103**

Сравнения решения, полученного методом ветвей и границ, с решением задачи методом полного перебора. Результат работы программы представлен на рис.



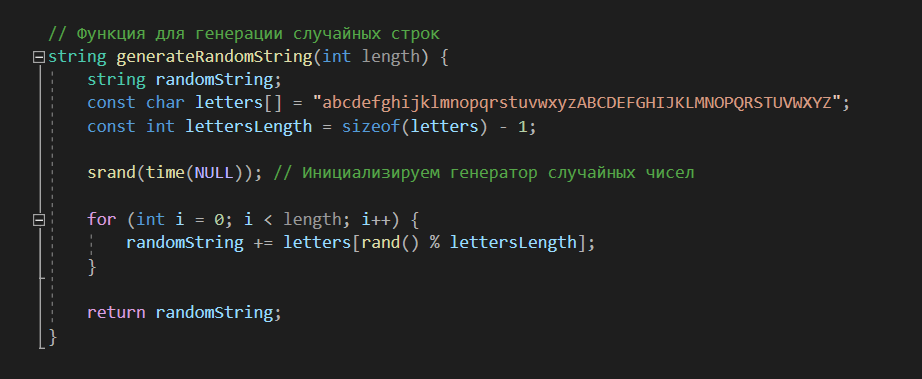
Полученные значения идентичны с результатом работы метода ветвей и границ.

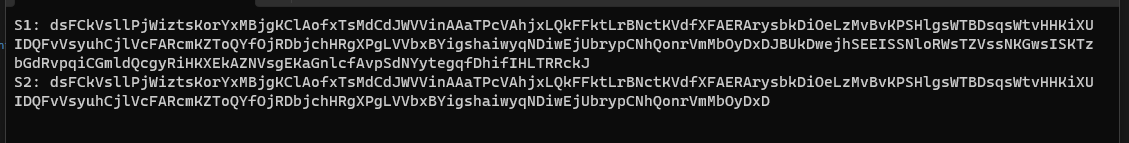
Вывод: освоил общие принципы решения задач методом ветвей и границ, решил задачу о коммивояжере данным методом, сравнил полученное решение задачи с комбинаторным методом перестановок.

# Лабораторная работа 4

***Задание 1.***

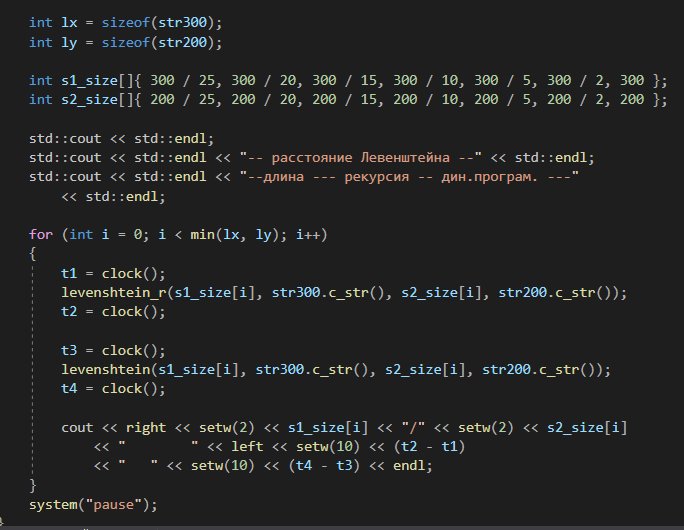
На языке С++ сгенерировать случайным образом строку букв латинского алфавита S1 длиной 300 символов и S2 длиной 200.

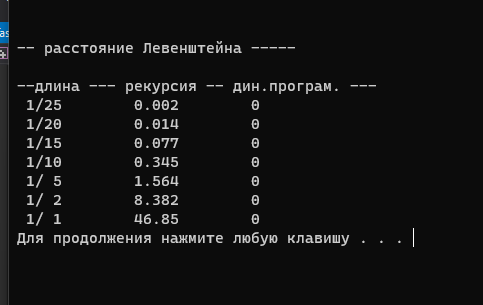




***Задание 2.***

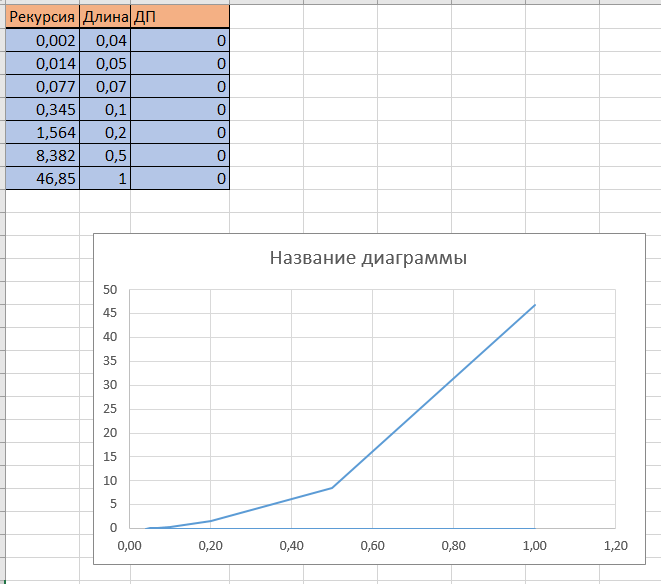
Вычислить двумя способами (рекурсивно и с помощью динамического программирования)  – дистанцию Левенштейна для , где - длина строки ,  - строка состоящая из первых  символов строки . (копии экрана и код вставить в отчет).





***Задание 3.***

Выполнить сравнительный анализ времени затраченного на вычисление дистанции Левенштейна для двух методов решения. Построить графики зависимости времени вычисления от . (копии экрана и график вставить в отчет).



Сравнительный анализ времени выполнения

***Задание 4.***

Реализовать вручную пример вычисления дистанции Левенштейна при помощи рекурсивного алгоритма (в соответствии с вариантом) (каждый шаг алгоритма по примеру из лекции вставить в отчет).

**Задание 4**

|  |  |
| --- | --- |
| Задание 4 | |
| Вол | Колун |



= 5.

= 4.



= 4.

= 3.



= 3.

= 2.



= 3.

= 2.



= 1.



= 2.

= 1.

= 1.

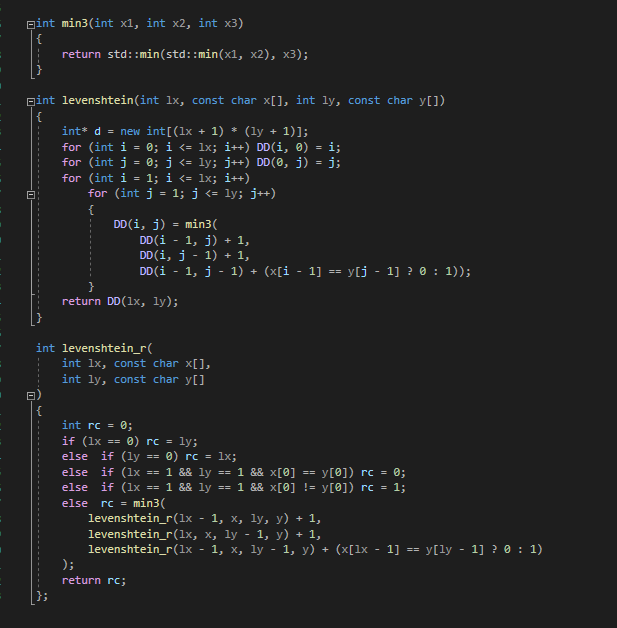


= 1.

= 2.

= 1.

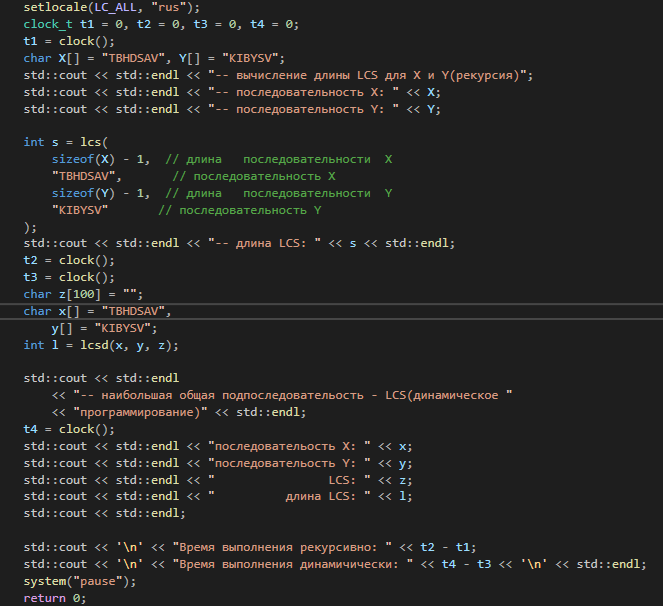


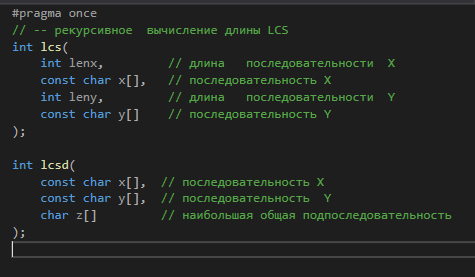


**Задание 5**

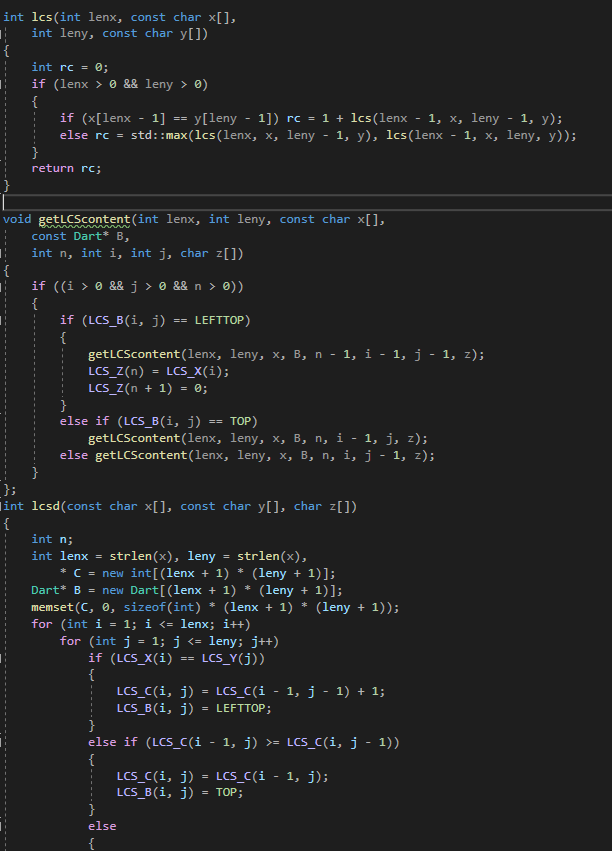
|  |  |
| --- | --- |
| Задание 5 | |
| 9 | |  |  | | --- | --- | | TBHDSAV | KIBYSV | |

**Нечетные варианты**. Выполнить сравнительный анализ времени затраченного на решение задачи о наибольшей общей подпоследовательности для двух методов решения (рекурсивное решение, динамическое программирование). Две последовательности взять в соответствии с вариантом. Построить графики зависимости времени вычисления от . **Отобразить ход решения в отчете**(по примеру из лекции) + код и копии экрана.



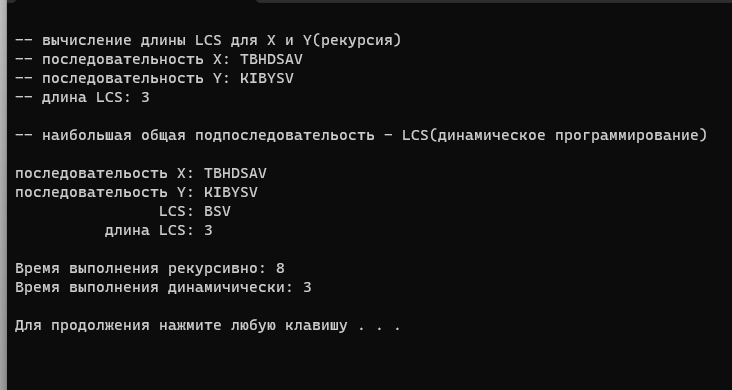
Main.cpp  
  
 s3

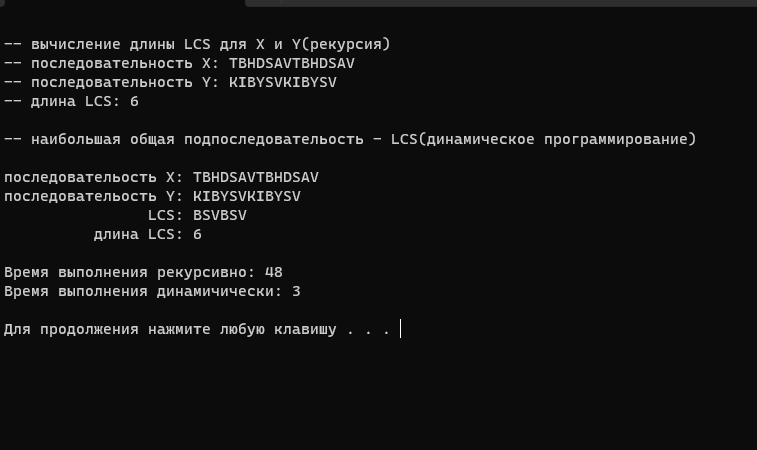
LCS.h

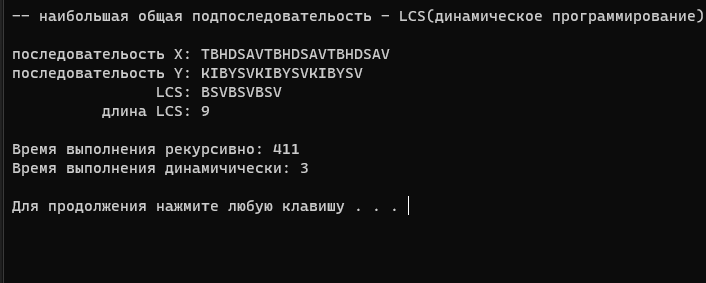


LCS.cpp

При выполнении сравнительного анализа времени, затраченного на решение задачи о наибольшей общей подпоследовательности для рекурсивного метода решения и для динамического, можно сделать вывод, что динамических работает быстрее, но при небольших последовательностях разница не очень большая. Результат выполнения программы представлен :









**Алгоритм прохождения:**

1. Создаем двумерный массив размером (len(TBHDSAV) + 1) на (len(KIBYSV) + 1), где каждый элемент равен 0.

2. Проходим по каждой строке и столбцу массива, начиная с первого и до последнего.

3. Если значение строки или столбца равно 0, то элемент массива на пересечении этой строки и столбца остается равным 0.

4. Если символ строки и столбца равны, то элемент массива на пересечении этой строки и столбца равен значению элемента на предыдущей диагонали плюс 1, т.е. с\_(𝑛,𝑝) = c\_(𝑛−1,𝑝−1) + 1.

5. Если символ строки и столбца не равны, то элемент массива на пересечении этой строки и столбца равен максимуму между значением элемента на предыдущей строке и значением элемента на предыдущем столбце, т.е. c\_(𝑛,𝑝) = max⁡( с\_(𝑛,𝑝"−1" ), c\_(𝑛−1,𝑝)).

6. После прохода по всем элементам массива, наибольшая общая подпоследовательность будет равна значению элемента на пересечении последней строки и последнего столбца.

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | TBHDSAV | KIBYSV | |

Алгоритм прохода будет выглядеть следующим образом:

|  |  |
| --- | --- |
| C(7,6)=C(6,5)+1  C(6,5)=max(C(6,4),C(5,5))  C(6,4)=max(C(6,3),C(5,4))  C(5,5)=C(4,4)+1  C(6,3)=max(C(6,2),C(5,3))  C(5,4)=max(C(5,3),C(4,4))  C(4,4)=max(C(4,3),C(3,4))  C(6,2)=max(C(6,1),C(5,2))  C(5,3)=max(C(5,2),C(4,3))  C(4,3)=max(C(4,2),C(3,3))  C(3,4)=max(C(3,3),C(2,4))  C(6,1)=max(C(6,0),C(5,1))  C(5,2)=max(C(5,1),C(4,2))  C(4,2)=max(C(4,1),C(3,2))  C(3,3)=max(C(3,2),C(2,3))  C(2,4)=max(C(2,3),C(1,3))  C(6,0)=0  C(5,1)=max(C(5,0),C(4,1))  C(4,1)=max(C(4,0),C(3,1))  C(3,2)=max(C(3,1),C(2,2))  C(2,3)=C(1,2)+1  С(1,3)=max(C(1,2),C(0,2))  C(5,0)=0  С(4,0)=0  C(3,1)=max(C(3,0),C(2,1))  C(2,2)=max(C(2,1),C(1,2))  C(1,2)=max(C(1,1),C(0,2))  C(0,2)=0  C(3,0)=0  C(2,1)=max(C(2,0),C(1,1)  C(2,0)=0  C(1,1)=max(C(1,0),C(0,1))  C(0,1)=0  C(1,0)=0 | C(1,1)=max(0,0)=0  C(2,1)=max(0,0)=0  C(1,2)=max(0,0)=0  C(2,2)=max(0,0)=0  C(3,1)=max(0,0)=0  C(1,3)=max(0,0)=0  C(2,3)=0+1=1  C(3,2)=max(0,0)=0  C(4,1)=max(0,0)=0  C(5,1)=max(0,0)=0  C(2,4)=max(1,0)=1  C(3,3)=max(0,1)=1  C(4,2)=max(0,0)=0  C(5,2)=max(0,0)=0  C(6,1)=max(0,0)=0  C(3,4)=max(1,1)=1  C(4,3)=max(0,1)=1  C(5,3)=max(0,1)=1  C(6,2)=max(0,0)=0  C(4,4)=max(1,1)=1  C(5,4)=max(1,1)=1  C(6,3)=max(0,1)=1  C(5,5)=1+1=2  C(6,4)=max(1,1)=1  C(6,5)=max(1,2)=2  C(7,6)=2+1=3 |

Вывод: в результате изучения метода динамического программирования можно сделать вывод, что данный метод основан на разбиении сложной задачи на меньшие подзадачи, решение которых сохраняется для дальнейшего использования. Это позволяет снизить время выполнения задачи и улучшить ее эффективность.

Сравнив решения задач, полученные методом динамического программирования и рекурсивным методом, можно отметить, что первый метод обладает меньшей сложностью и более быстрой скоростью выполнения. Рекурсивный метод может привести к переполнению стека и повышенному потреблению памяти.

Таким образом, освоение принципов динамического программирования позволяет эффективнее решать сложные задачи, сокращая время и затраты на их выполнение.

Расстояние Левенштейна является эффективным инструментом для сравнения двух строк. Он может быть использован во многих задачах, таких как поиск ошибок в написании слов, определение сходства текстов и т.д. Более того, расстояние Левенштейна позволяет определить наименьшее количество операций, которые необходимо выполнить, чтобы преобразовать одну строку в другую. Таким образом, алгоритм Левенштейна очень полезен в обработке текстовой информации и может быть использован во многих приложениях, где необходима сравнительная аналитика текстовых данных.

# Лабораторная работа 5

**Цель работы:** Приобретение навыков решения открытой транспортной задачи

**Задание для выполнения:**

**Задание.** Решить транспортную задачу. Имеется 5 поставщиков продукции и 6 потребителей. Величина запасов, потребностей и стоимость затрат на перевозку продукции взять в соответствии с вариантом (*N*). Оформить отчет.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12** | **20** | **10** | **177** |
| 2 | **19** | **9** | **17** | **14** | **16** | **22** | **122** |
| 3 | **10** | **14** | **20** | **17** | **11** | **20** | **159** |
| 4 | **13** | **19** | **19** | **12** | **22** | **11** | **168** |
| 5 | **12** | **20** | **18** | **9** | **19** | **13** | **109** |
| ПОТРЕБНОСТИ | **152** | **116** | **140** | **202** | **104** | **172** |  |

**Проверка задачи на закрытость**

Обозначим суммарный запас груза у всех поставщиков символом A, а суммарную потребность в грузе у всех потребителей – символом B. Тогда: Транспортная задача называется закрытой, если A = B. Если же A ≠ B, то транспортная задача называется открытой. В случае закрытой задачи от поставщиков будут вывезены все запасы груза, и все заявки потребителей будут удовлетворены. В случае открытой задачи для ее решения придется вводить фиктивных поставщиков или потребителей.

Проверим необходимое и достаточное условие разрешимости задачи.

∑А = 177 + 122 + 159 + 168 + 109 = 735  
∑В = 152 + 116 + 140 + 202 + 104 + 172 = 886

Как видно, суммарная потребность груза в пунктах назначения превышает запасы груза на базах. Следовательно, модель исходной транспортной задачи является открытой. Чтобы получить закрытую модель, введем дополнительную (фиктивную) базу с запасом груза, равным 151 (886-735). Тарифы перевозки единицы груза из базы во все магазины полагаем равны нулю.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12** | **20** | **10** | **177** |
| 2 | **19** | **9** | **17** | **14** | **16** | **22** | **122** |
| 3 | **10** | **14** | **20** | **17** | **11** | **20** | **159** |
| 4 | **13** | **19** | **19** | **12** | **22** | **11** | **168** |
| 5 | **12** | **20** | **18** | **9** | **19** | **13** | **109** |
| 6 | **0** | **0** | **0** | **0** | **0** | **0** | **151** |
| ПОТРЕБНОСТИ | **152** | **116** | **140** | **202** | **104** | **172** |  |

Используя метод наименьшей стоимости, построим первый опорный план транспортной задачи. Суть метода заключается в том, что из всей таблицы стоимостей выбирают наименьшую, и в клетку, которая ей соответствует, помещают меньшее из чисел ai, или bj. Затем, из рассмотрения исключают либо строку, соответствующую поставщику, запасы которого полностью израсходованы, либо столбец, соответствующий потребителю, потребности которого полностью удовлетворены, либо и строку и столбец, если израсходованы запасы поставщика и удовлетворены потребности потребителя. Из оставшейся части таблицы стоимостей снова выбирают наименьшую стоимость, и процесс распределения запасов продолжают, пока все запасы не будут распределены, а потребности удовлетворены.  
Искомый элемент равен (ai=2, bi=2)=9. Для этого элемента запасы равны 122, потребности 116. Поскольку минимальным является 116, то вычитаем его.  
x22 = min(122,116) = 116.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **-** | **15** | **12** | **20** | **10** | **177** |
| 2 | **19** | **9** | **17** | **14** | **16** | **22** | **6** |
| 3 | **10** | **-** | **20** | **17** | **11** | **20** | **159** |
| 4 | **13** | **-** | **19** | **12** | **22** | **11** | **168** |
| 5 | **12** | **-** | **18** | **9** | **19** | **13** | **109** |
| 6 | **0** | **-** | **0** | **0** | **0** | **0** | **151** |
| ПОТРЕБНОСТИ | **152** | **0** | **140** | **202** | **104** | **172** |  |

Искомый элемент равен (ai=5, bi=4) =9. Для этого элемента запасы равны 109, потребности 202. Поскольку минимальным является 109, то вычитаем его.  
min(109,202) = 109.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **-** | **15** | **12** | **20** | **10** | **177** |
| 2 | **19** | **9** | **17** | **14** | **16** | **22** | **6** |
| 3 | **10** | **-** | **20** | **17** | **11** | **20** | **159** |
| 4 | **13** | **-** | **19** | **12** | **22** | **11** | **168** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **0** | **151** |
| ПОТРЕБНОСТИ | **152** | **0** | **140** | **93** | **104** | **172** |  |

Искомый элемент равен (ai=1, bi=6) =10. Для этого элемента запасы равны 177, потребности 172. Поскольку минимальным является 172, то вычитаем его.  
min(177,172) = 172.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **-** | **15** | **12** | **20** | **10** | **5** |
| 2 | **19** | **9** | **17** | **14** | **16** | **-** | **6** |
| 3 | **10** | **-** | **20** | **17** | **11** | **-** | **159** |
| 4 | **13** | **-** | **19** | **12** | **22** | **-** | **168** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **152** | **0** | **140** | **93** | **104** | **0** |  |

Искомый элемент равен (ai=3, bi=1) =10. Для этого элемента запасы равны 159, потребности 152. Поскольку минимальным является 152, то вычитаем его.  
min(159,152) = 152.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **15** | **12** | **20** | **10** | **5** |
| 2 | **-** | **9** | **17** | **14** | **16** | **-** | **6** |
| 3 | **10** | **-** | **20** | **17** | **11** | **-** | **7** |
| 4 | **-** | **-** | **19** | **12** | **22** | **-** | **168** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **0** | **0** | **140** | **93** | **104** | **0** |  |

Искомый элемент равен (ai=3, bi=5) =11. Для этого элемента запасы равны 7, потребности 104. Поскольку минимальным является 7, то вычитаем его.  
 min(7,104) = 7.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **15** | **12** | **20** | **10** | **5** |
| 2 | **-** | **9** | **17** | **14** | **16** | **-** | **6** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **22** | **-** | **168** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **0** | **0** | **140** | **93** | **97** | **0** |  |

Искомый элемент равен (ai=1, bi=4) =12. Для этого элемента запасы равны 5, потребности 93 . Поскольку минимальным является 5, то вычитаем его.  
 min(5,93) = 5.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **-** | **12** | **-** | **10** | **0** |
| 2 | **-** | **9** | **17** | **14** | **16** | **-** | **6** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **22** | **-** | **168** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **0** | **0** | **140** | **88** | **97** | **0** |  |

Искомый элемент равен (ai=4, bi=4) =12. Для этого элемента запасы равны 168, потребности 88. Поскольку минимальным является 88, то вычитаем его.  
min(168,88) = 88.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **-** | **12** | **-** | **10** | **0** |
| 2 | **-** | **9** | **17** | **-** | **16** | **-** | **6** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **22** | **-** | **80** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **0** | **0** | **140** | **0** | **97** | **0** |  |

Искомый элемент равен (ai=2, bi=5) =16. Для этого элемента запасы равны 6, потребности 97. Поскольку минимальным является 6, то вычитаем его.  
min(6,97) = 6.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **-** | **12** | **-** | **10** | **0** |
| 2 | **-** | **9** | **-** | **-** | **16** | **-** | **0** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **22** | **-** | **80** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **0** | **0** | **140** | **0** | **91** | **0** |  |

Искомый элемент равен (ai=4, bi=3) =19. Для этого элемента запасы равны 80, потребности 140. Поскольку минимальным является 80, то вычитаем его.  
min(80,140) = 80.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **-** | **12** | **-** | **10** | **0** |
| 2 | **-** | **9** | **-** | **-** | **16** | **-** | **0** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **-** | **-** | **0** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **0** | **-** | **0** | **0** | **0** | **-** | **151** |
| ПОТРЕБНОСТИ | **0** | **0** | **60** | **0** | **91** | **0** |  |

Искомый элемент равен (ai=6, bi=3) =0. Для этого элемента запасы равны 151, потребности 60. Поскольку минимальным является 60, то вычитаем его.  
min(151,60) = 60.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **-** | **12** | **-** | **10** | **0** |
| 2 | **-** | **9** | **-** | **-** | **16** | **-** | **0** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **-** | **-** | **0** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **-** | **-** | **0** | **-** | **0** | **-** | **91** |
| ПОТРЕБНОСТИ | **0** | **0** | **0** | **0** | **91** | **0** |  |

Искомый элемент равен (ai=6, bi=5) =0. Для этого элемента запасы равны 91, потребности 91. Поскольку минимальным является 91, то вычитаем его.  
min(91,91) = 91.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **-** | **-** | **-** | **12** | **-** | **10** | **0** |
| 2 | **-** | **9** | **-** | **-** | **16** | **-** | **0** |
| 3 | **10** | **-** | **-** | **-** | **11** | **-** | **0** |
| 4 | **-** | **-** | **19** | **12** | **-** | **-** | **0** |
| 5 | **-** | **-** | **-** | **9** | **-** | **-** | **0** |
| 6 | **-** | **-** | **0** | **-** | **0** | **-** | **0** |
| ПОТРЕБНОСТИ | **0** | **0** | **0** | **0** | **0** | **0** |  |

В результате получаем:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12(5)** | **20** | **10(172)** | **177** |
| 2 | **19** | **9(116)** | **17** | **14** | **16(6)** | **22** | **122** |
| 3 | **10(152)** | **14** | **20** | **17** | **11(7)** | **20** | **159** |
| 4 | **13** | **19** | **19(80)** | **12(88)** | **22** | **11** | **168** |
| 5 | **12** | **20** | **18** | **9(109)** | **19** | **13** | **109** |
| 6 | **0** | **0** | **0(60)** | **0** | **0(91)** | **0** | **151** |
| ПОТРЕБНОСТИ | **152** | **116** | **140** | **202** | **104** | **172** |  |

В результате получен первый опорный план, который является допустимым, так как все грузы из баз вывезены, потребность магазинов удовлетворена, а план соответствует системе ограничений транспортной задачи.

Подсчитаем число занятых клеток таблицы, их 11, а должно быть m + n - 1 = 11. Следовательно, опорный план является невырожденным.  
Значение целевой функции для этого опорного плана равно:  
F(x) = 12\*5 + 10\*172 + 9\*116 + 16\*6 + 10\*152 + 11\*7 + 19\*80 + 12\*88 + 9\*109 + 0\*60 + 0\*91 = 8074

Этап II. Улучшение опорного плана.  
Проверим оптимальность опорного плана. Найдем предварительные потенциалы ui, vj. по занятым клеткам таблицы, в которых ui + vj = cij, полагая, что u1 = 0.  
u1 + v4 = 12; 0 + v4 = 12; v4 = 12  
u4 + v4 = 12; 12 + u4 = 12; u4 = 0  
u4 + v3 = 19; 0 + v3 = 19; v3 = 19  
u6 + v3 = 0; 19 + u6 = 0; u6 = -19  
u6 + v5 = 0; -19 + v5 = 0; v5 = 19  
u2 + v5 = 16; 19 + u2 = 16; u2 = -3  
u2 + v2 = 9; -3 + v2 = 9; v2 = 12  
u3 + v5 = 11; 19 + u3 = 11; u3 = -8  
u3 + v1 = 10; -8 + v1 = 10; v1 = 18  
u5 + v4 = 9; 12 + u5 = 9; u5 = -3  
u1 + v6 = 10; 0 + v6 = 10; v6 = 10

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12(5)** | **20** | **10(172)** | **u1=0** |
| 2 | **19** | **9(116)** | **17** | **14** | **16(6)** | **22** | **u2=-3** |
| 3 | **10(152)** | **14** | **20** | **17** | **11(7)** | **20** | **u3=-8** |
| 4 | **13** | **19** | **19(80)** | **12(88)** | **22** | **11** | **u4=0** |
| 5 | **12** | **20** | **18** | **9(109)** | **19** | **13** | **u5=-3** |
| 6 | **0** | **0** | **0(60)** | **0** | **0(91)** | **0** | **u6=-19** |
| ПОТРЕБНОСТИ | **v1=18** | **v2=12** | **v3=19** | **v4=12** | **v5=19** | **v6=10** |  |

Опорный план не является оптимальным, так как существуют оценки свободных клеток, для которых ui + vj > cij:  
(1;2): 0 + 12 >11; ∆12 = 0 + 12 - 11 = 1 > 0  
(1;3): 0 + 19 > 15; ∆13 = 0 + 19 - 15 = 4 > 0  
(4;1): 0 + 18 > 13; ∆41 = 0 + 18 - 13 = 5 > 0  
(5;1): -3 + 18 > 12; ∆51 = -3 + 18 - 12 = 3 > 0  
max(1,4,5,3) = 5

Выбираем максимальную оценку свободной клетки (4;1): 13

Для этого в перспективную клетку (4;1) поставим знак «+», а в остальных вершинах многоугольника чередующиеся знаки «-», «+», «-».

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12[5]** | **20** | **10[172]** | **u1=0** |
| 2 | **19** | **9[116]** | **17** | **14** | **16[6]** | **22** | **u2=-3** |
| 3 | **10(152-)** | **14** | **20** | **17** | **11(7+)** | **20** | **u3=-8** |
| 4 | **13(+)** | **19** | **19(80-)** | **12(88)** | **22** | **11** | **u4=0** |
| 5 | **12** | **20** | **18** | **9(109)** | **19** | **13** | **u5=-3** |
| 6 | **0** | **0** | **0(60+)** | **0** | **0(91-)** | **0** | **u6=-19** |
| ПОТРЕБНОСТИ | **v1=18** | **v2=12** | **v3=19** | **v4=12** | **v5=19** | **v6=10** |  |

Цикл приведен в таблице (4,1 → 4,3 → 6,3 → 6,5 → 3,5 → 3,1).  
Из грузов хij стоящих в минусовых клетках, выбираем наименьшее, т.е. у = min (4, 3) = 80. Прибавляем 80 к объемам грузов, стоящих в плюсовых клетках и вычитаем 80 из Хij, стоящих в минусовых клетках. В результате получим новый опорный план.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12[5]** | **20** | **10[172]** | **177** |
| 2 | **19** | **9[116]** | **17** | **14** | **16[6]** | **22** | **122** |
| 3 | **10[72]** | **14** | **20** | **17** | **11[87]** | **20** | **159** |
| 4 | **13[80]** | **19** | **19** | **12[88]** | **22** | **11** | **168** |
| 5 | **12** | **20** | **18** | **9[109]** | **19** | **13** | **109** |
| 6 | **0** | **0** | **0[140]** | **0** | **0(11)** | **0** | **151** |
| ПОТРЕБНОСТИ | **152** | **116** | **140** | **202** | **104** | **172** |  |

Проверим оптимальность опорного плана. Найдем предварительные потенциалы ui, vj. по занятым клеткам таблицы, в которых ui + vj = cij, полагая, что u1 = 0.  
u1 + v4 = 12; 0 + v4 = 12; v4 = 12  
u4 + v4 = 12; 12 + u4 = 12; u4 = 0  
u4 + v1 = 13; 0 + v1 = 13; v1 = 13  
u3 + v1 = 10; 13 + u3 = 10; u3 = -3  
u3 + v5 = 11; -3 + v5 = 11; v5 = 14  
u2 + v5 = 16; 14 + u2 = 16; u2 = 2  
u2 + v2 = 9; 2 + v2 = 9; v2 = 7  
u6 + v5 = 0; 14 + u6 = 0; u6 = -14  
u6 + v3 = 0; -14 + v3 = 0; v3 = 14  
u5 + v4 = 9; 12 + u5 = 9; u5 = -3  
u1 + v6 = 10; 0 + v6 = 10; v6 = 10

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ПОТРЕБИТЕЛИ  ПОСТАВЩИКИ | 1 | 2 | 3 | 4 | 5 | 6 | ЗАПАСЫ |
| 1 | **21** | **11** | **15** | **12[5]** | **20** | **10[172]** | **u1=0** |
| 2 | **19** | **9[116]** | **17** | **14** | **16[6]** | **22** | **u2=2** |
| 3 | **10[72]** | **14** | **20** | **17** | **11[87]** | **20** | **u3=-3** |
| 4 | **13[80]** | **19** | **19** | **12[88]** | **22** | **11** | **u4=0** |
| 5 | **12** | **20** | **18** | **9[109]** | **19** | **13** | **u5=-3** |
| 6 | **0** | **0** | **0[140]** | **0** | **0(11)** | **0** | **u6=-14** |
| ПОТРЕБНОСТИ | **v1=13** | **v2=7** | **v3=14** | **v4=12** | **v5=14** | **v6=10** |  |

Опорный план является оптимальным, так все оценки свободных клеток удовлетворяют условию ui + vj ≤ cij.  
Минимальные затраты составят:  F(x) = 12\*5 + 10\*172 + 9\*116 + 16\*6 + 10\*72 + 11\*87 + 13\*80 + 12\*88 + 9\*109 + 0\*140 + 0\*11 = 7674

Анализ оптимального плана.  
Из 1-го склада необходимо груз направить в 4-й магазин (5 ед.), в 6-й магазин (172ед.)  
Из 2-го склада необходимо груз направить в 2-й магазин (116 ед.), в 5-й магазин (6ед.)  
Из 3-го склада необходимо груз направить в 1-й магазин (72 ед.), в 5-й магазин (87ед.)  
Из 4-го склада необходимо груз направить в 1-й магазин (80 ед.), в 4-й магазин (88ед.)  
Из 5-го склада необходимо весь груз направить в 4-й магазин(109ед.).  
Потребность 3-го магазина остается неудовлетворенной на 140 ед.  
Оптимальный план является вырожденным, так как базисная переменная x63=0.  
Потребность 5-го магазина остается неудовлетворенной на 11 ед.  
Оптимальный план является вырожденным, так как базисная переменная x65=0.

Вывод: В ходе решения лабораторной работе мной были приобретены навыки решения транспортной задачи.