

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Кафедра программных систем

**Задание №10**

**Решение симплекс таблицы**

Выполнил: Мороз

Илья Олегович

Группа № K3220

Преподаватель: Иванов Сергей Евгеньевич

Санкт-Петербург

2020

**Задание:** реализовать решение второго порядка.

**Исходный код:**

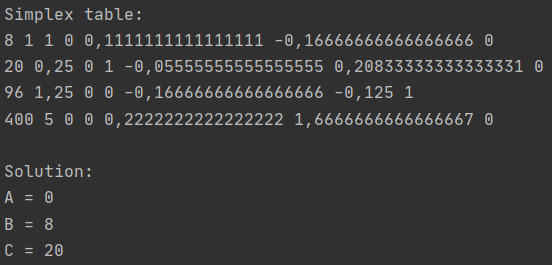
**Program.cs**

using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
  
namespace task10  
{  
 class Program  
 {  
 static void Main(string[] args)  
 {  
 double[,] table =  
 { {360, 18, 15, 12},  
 {192, 6, 4, 8},  
 {180, 5, 3, 3},  
 { 0, -9, -10, -16} };  
  
 double[] result = new double[3];  
 double[,] table\_result;  
 Simplex S = new Simplex(table);  
 table\_result = S.Calculate(result);  
  
 Console.WriteLine("Solved simplex table:");  
 for (int i = 0; i < table\_result.GetLength(0); i++)  
 {  
 for (int j = 0; j < table\_result.GetLength(1); j++)  
 Console.Write(table\_result[i, j] + " ");  
 Console.WriteLine();  
 }  
  
 Console.WriteLine();  
 Console.WriteLine("Solution of simplex table:");  
 Console.WriteLine("A = " + result[0]);  
 Console.WriteLine("B = " + result[1]);  
 Console.WriteLine("C = " + result[2]);  
 Console.ReadLine();  
 }  
 }  
}

**Simplex.cs**

using System.Collections.Generic;  
  
namespace task10  
{  
 public class Simplex  
 {  
 */\*  
 \* Simplex table  
 \*/* double[,] table;  
  
 int m, n;  
  
 List<int> basis;  
  
 public Simplex(double[,] source)  
 {  
 m = source.GetLength(0);  
 n = source.GetLength(1);  
 table = new double[m, n + m - 1];  
 basis = new List<int>();  
  
 for (int i = 0; i < m; i++)  
 {  
 for (int j = 0; j < table.GetLength(1); j++)  
 {  
 if (j < n)  
 table[i, j] = source[i, j];  
 else  
 table[i, j] = 0;  
 }  
  
 if ((n + i) < table.GetLength(1))  
 {  
 table[i, n + i] = 1;  
 basis.Add(n + i);  
 }  
 }  
  
 n = table.GetLength(1);  
 }  
  
 */\*  
 \* Write result in param result[]  
 \*/* public double[,] Calculate(double[] result)  
 {  
 int mainCol, mainRow;  
  
 while (!IsItEnd())  
 {  
 mainCol = findMainCol();  
 mainRow = findMainRow(mainCol);  
 basis[mainRow] = mainCol;  
  
 double[,] new\_table = new double[m, n];  
  
 for (int j = 0; j < n; j++)  
 new\_table[mainRow, j] = table[mainRow, j] / table[mainRow, mainCol];  
  
 for (int i = 0; i < m; i++)  
 {  
 if (i == mainRow)  
 continue;  
  
 for (int j = 0; j < n; j++)  
 new\_table[i, j] = table[i, j] - table[i, mainCol] \* new\_table[mainRow, j];  
 }  
  
 table = new\_table;  
 }  
  
 */\*  
 \* Write result  
 \*/* for (int i = 0; i < result.Length; i++)  
 {  
 int k = basis.IndexOf(i + 1);  
 if (k != -1)  
 result[i] = table[k, 0];  
 else  
 result[i] = 0;  
 }  
  
 return table;  
 }  
  
 private bool IsItEnd()  
 {  
 bool flag = true;  
  
 for (int j = 1; j < n; j++)  
 {  
 if (table[m - 1, j] < 0)  
 {  
 flag = false;  
 break;  
 }  
 }  
  
 return flag;  
 }  
  
 private int findMainCol()  
 {  
 int mainCol = 1;  
  
 for (int j = 2; j < n; j++)  
 if (table[m - 1, j] < table[m - 1, mainCol])  
 mainCol = j;  
  
 return mainCol;  
 }  
  
 private int findMainRow(int mainCol)  
 {  
 int mainRow = 0;  
  
 for (int i = 0; i < m - 1; i++)  
 if (table[i, mainCol] > 0)  
 {  
 mainRow = i;  
 break;  
 }  
  
 for (int i = mainRow + 1; i < m - 1; i++)  
 if ((table[i, mainCol] > 0) &&  
 ((table[i, 0] / table[i, mainCol]) < (table[mainRow, 0] / table[mainRow, mainCol])))  
 mainRow = i;  
  
 return mainRow;  
 }  
 }  
}

**Результат работы:**

****

**Вывод:** был реализовано нахождение решения симплекс таблицы.

Исходный код также доступен на GitHub: [ilyamore88/applied-informatics/task10](https://github.com/ilyamore88/applied-informatics/tree/master/task10).