Министерство образования Республики Беларусь

Учреждение образования

" *** "

Факультет *****************************

Кафедра *****************************

Курсовая работа

по дисциплине

"Экономико-математические методы и модели принятия решений"

Выполнил: студент 4 курса, группа 123456 — Иванов Иван Иванович

Преподаватель:

Минск 2014

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Предметом изучения курса экономико-математических методов принятия решений (ЭММ) являются количественные характеристики экономических процессов, протекающих в промышленном и сельскохозяйственном производстве, изучение их взаимосвязей на основе экономико-математических методов и моделей. В курсе ЭММ рассматриваются конкретные задачи и их экономико-математические модели. Это модели линейного и нелинейного программирования, сетевого планирования и управления, балансовые, игровые, имитационные, модели исследования операций, модели массового обслуживания. Изучаются также модели оптимального использования производственных мощностей, в частности, модели загрузки взаимозаменяемых и невзаимозаменяемых групп оборудования, модели и методы оперативно-календарного планирования.

Отдельно выделяются задачи оптимизации состава (промышленных смесей, рациона кормления животных, посевных площадей, раскроя материалов и пр.). Особое внимание уделяется методам и моделям прогнозирования конъюнктуры рынка и определения цен, моделям и методам анализа инвестиционных проектов, моделям в управлении финансами.

Большое значение имеют тренинговые системы моделирования работы предприятий на основе деловой игры "Дельта", включающие вопросы рекламы, сервиса и научные исследования в стратегии маркетинга деловой игры, проектирование производства и сбыта продукции, исследование рынка предприятий.

Немалое место отводится моделям оптимального отраслевого и регионального регулирования - экономико-математическим моделям проекта развития отдельных отраслей промышленности. Это такие важные модели, как вариантная, транспортно-производственная, модель расчета топливного баланса региона.

В курсе ЭММ освещаются модели народнохозяйственного регулирования (в частности, межотраслевого баланса, базовые, статистические и динамические модели, модели ценообразования на основе межотраслевого баланса). Представляется модель взаимосвязи конечного использования и валового продукта. Излагаются вопросы сингулярных моделей макроэкономического прогнозирования.

В отдельный раздел курса как правило выделяются экспертные методы прогнозирования в менеджменте при принятии решений в условиях неопределенности и риска.

Наряду с освещением вышеизложенных методов и моделей важное место в ЭММ занимают вопросы унификации символики и записи моделей, информационное и математическое обеспечение экономико-математических моделей, система критериев оптимальности, виды производственных функций в экономическом анализе и управлении производством.

Основным понятием курса является понятие математической модели. В общем случае слово модель - это отражение реального объекта. Такое отражение объекта может быть представлено схемой, эскизом, фотографией, моделью описательного характера в виде графиков и таблиц и т. д. *Математическая модель* - это система математических уравнений, неравенств, формул и различных математических выражений, описывающих реальный объект, составляющие его характеристики и взаимосвязи между ними. Процесс построения математической модели называют *математическим моделированием.* Естественно, моделирование и построение математической модели экономического объекта позволяют свести экономический анализ производственных процессов к математическому анализу и принятию эффективных решений.

Поскольку в курсе ЭММ изучаются экономические задачи, то и строятся экономико-математические модели, включающие:

1) выбор некоторого числа переменных величин для формализации модели объекта;

2) информационную базу данных объекта;

3) выражение взаимосвязей, характеризующих объект, в виде уравнений и неравенств;

4) выбор критерия эффективности и выражение его в виде математического соотношения — целевой функции.

Итак, для принятия эффективных решений в планировании и управлении производством необходимо экономическую сущность исследуемого экономического объекта формализовать экономико-математической моделью, т. е. экономическую задачу представить математически в виде уравнений, неравенств и целевой функции на экстремум (максимум или минимум) при выполнении всех условий на ограничения и переменные.

Применение экономико-математических методов основывается на знании содержания процессов, явлений, по которым составляется задача. Однако количественные, экономико-математические модели базируются на понимании качественной природы явлений. Они призваны конкретизировать и обогащать качественный анализ взаимосвязей в исследуемой области.

1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1.1. Постановка оптимизационной задачи

В задаче оптимизации необходимо выбрать наилучший варианта из всех возможных. С точки зрения инженерных расчетов методы оптимизации позволяют выбрать наилучший вариант конструкции, наилучшее распределение ресурсов и т.д.

В процессе решения задачи оптимизации обычно необходимо найти оптимальные значения некоторых параметров, определяющих данную задачу. При решении инженерных задач их принято называть *проектными параметрами*, а в экономических задачах их обычно называют *параметрами плана*. В качестве проектных параметров могут быть, в частности, значения линейных размеров объекта, массы, температуры и т.п. число n проектных параметров $x_1,x_2,\ldots,x_n$ характеризует размерность (и степень сложности) задачи оптимизации.

Выбор оптимального решения задачи оптимизации проводится с помощью *целевой функции*, которую можно представить в виде

$$f(x_1,x_2,\ldots,x_n) \rightarrow \text{extr} \qquad (1)$$

Наиболее часто имеют место *условные задачи оптимизации*, или задачи с ограничениями, при формулировке которых задаются некоторые условия (ограничения) на множестве допустимых значений. Эти ограничения задаются совокупностью некоторых функций, удовлетворяющих уравнениям или неравенствам, и отражают законы природы, наличие ресурсов т. п.

При наличие ограничений оптимальное решение может соответствовать либо локальному экстремуму внутри области проектирования, либо значению целевой функции на границе области. Если ограничения отсутствуют, то ищется оптимальное решение на всей области проектирования, то есть глобальный экстремум.

1.2. Симплекс-метод

Если целевая функция (1) представляет собой линейную функцию вида

$$f = c_1x_1 + c_2x_2 + ... + c_nx_n \rightarrow extr, \qquad (2)$$

а система ограничений

$$\begin{cases} \sum_{j=1}^{n} a_{ij}x_j \le b_i \quad \left(i = \overline{1, m_1}\right), \\ \sum_{j=1}^{n} a_{ij}x_j = b_i \quad \left(i = \overline{m_1 + 1, m_2}\right), \\ \sum_{j=1}^{n} a_{ij}x_j \ge b_i \quad \left(i = \overline{m_2 + 1, m}\right), \\ x_j \ge 0 \quad \left(j = \overline{1, n}\right), \end{cases} \qquad (3)$$

имеем задачу линейного программирования.

Классическим методом решения задач линейного программирования стал *симплекс-метод* (метод последовательного улучшения плана). Идея этого метода состоит в последовательном переборе угловых точек множества допустимых планов задачи, т.е. в последовательном улучшении опорных планов задачи по определенному критерию до тех пор, пока не будет получено оптимальное решение.

Алгоритм симплекс-метода состоит из следующих этапов:

1. Приведение задачи к каноническому виду.

2. Построение начального базисного плана.

3. Проверка базисного плана на оптимальность и неограниченность целевой функции.

4. Если базисный план является оптимальным или целевая функция неограниченна – конец решения, если нет – строим следующий базисный план, на котором значение целевой функции не хуже, чем на предыдущем. Затем переходим на шаг 3.

Более подробно симплекс-метод изложен в специальной литературе [1], [3].

2. ПРАКТИЧЕСКАЯ ЧАСТЬ

2.1. Постановка задачи

Для изготовления трёх видов изделий A, B, C используется токарное, фрезерное, сварочное и шлифовальное оборудование. Затраты времени на обработку одного изделия, общий фонд рабочего времени каждого из типов используемого оборудования, а также прибыль от реализации заданы в таблице:

Тип оборудования	Затраты времени на обработку одного изделия вида:			Общий фонд рабочего времени
	A	B	C	
Фрезеровочное	2	4	5	120
Токарное	1	8	6	280
Сварочное	7	4	5	240
Шлифовальное	4	6	7	360
Прибыль	10	14	12	

Требуется определить, сколько изделий и какого вида следует изготовить предприятию, чтобы прибыль от их реализации была максимальной.

2.2. Математическая модель задачи

Пусть планируется выпустить x_1 изделий A, x_2 изделий B, x_3 изделий C. Тогда суммарная прибыль от реализации

$$f = 10x_1 + 14x_2 + 12x_3 \qquad (4)$$

Необходимо найти максимальную прибыль $f = f_{max}$ и оптимальный план производства, то есть значения x_1 , x_2 , x_3, при которых функция f достигает максимума.

Выпуск продукции и величина прибыли ограничены имеющимися запасами ресурсов для производства продукции. Ресурсы в данной задачи представлены различными типами оборудования, необходимого для производства изделий.

Просуммировав затраты времени каждого типа оборудования на производство изделий всех видов, получим общий расход времени этого типа оборудования, который не должен превосходить имеющегося фонда рабочего времени. Записываем ограничения на использование каждого типа оборудования в виде неравенств

$$\begin{cases} 2x_1 + 4x_2 + 5x_3 \le 120 \\ x_1 + 8x_2 + 6x_3 \le 280 \\ 7x_1 + 4x_2 + 5x_3 \le 240 \\ 4x_1 + 6x_2 + 7x_3 \le 360 \end{cases} \qquad (5)$$

Причем по смыслу задачи выпуск продукции неотрицателен

$$x_j \ge 0, \; j = \overline{1,3} \qquad (6)$$

Соотношения (4), (5) и (6) составляют экономико - математическую модель задачи.

2.3. Решение задачи симплекс-методом

Для применения симплексного метода осуществим переход к эквивалентной канонической форме записи задачи, прибавив к левой части неравенств дополнительные неотрицательные переменные x_4, x_5, x_6 и x_7, которые имеют смысл возможных остатков неиспользуемого в производстве времени работы оборудования.

Получим

$$\begin{cases} 2x_1 + 4x_2 + 5x_3 + x_4 = 120 \\ x_1 + 8x_2 + 6x_3 + x_5 = 280 \\ 7x_1 + 4x_2 + 5x_3 + x_6 = 240 \\ 4x_1 + 6x_2 + 7x_3 + x_7 = 360 \end{cases}$$

В этой системе ограничений - равенств предпочтительными (базисными) являются переменные x_4, x_5, x_6 и x_7, поскольку они входят только в одно уравнение с коэффициентом, равным 1.

Занесем полученные условия задачи в симплексную таблицу

Базисные переменные	Свободные члены	Свободные переменные		
		x_1	x_2	x_3
x_4	120	2	$\boxed{4}$	5
x_5	280	1	8	6
x_6	240	7	4	5
x_7	360	4	6	7
f	0	-10	-14	-12

Для задачи максимизации условием оптимальности опорного плана является неотрицательность оценок функции f. В данном случае все оценки отрицательны. Наибольшая из них по абсолютной величине соответствует столбцу переменной x_2. Этот столбец и назначим разрешающим.

Для определения разрешающей строки находим минимальное симплексное отношение

$$\min\left\{\frac{120}{4}, \frac{280}{8}, \frac{240}{4}, \frac{360}{7}\right\} = \frac{120}{4}$$

Оно соответствует первой строке, которая и будет разрешающей. Следовательно, элемент a_{12} - разрешающий, в таблице он выделен рамкой. Переменную x_4 выведем из базиса, а переменную x_2 введем в базис, пересчитав элементы таблицы.

После пересчета таблица примет вид

Базисные переменные	Свободные члены	Свободные переменные		
		x_1	x_4	x_3
x_2	30	0.5	0.25	1.25
x_5	40	-3	-2	-4
x_6	120	$\boxed{5}$	-1	0
x_7	180	1	-1.5	-0.5
f	420	-3	3.5	5.5

Первый столбец – разрешающий. Для определения разрешающей строки находим минимальное симплексное отношение

$$\min\left\{\frac{30}{0.5}, \frac{120}{5}, \frac{180}{1}\right\} = \frac{120}{5}$$

Оно соответствует третьей строке, которая и будет разрешающей. Следовательно, элемент a_{31} - разрешающий, в таблице он выделен рамкой. Переменную x_6 выведем из базиса, а переменную x_1 введем в базис, пересчитав элементы таблицы.

Базисные переменные	Свободные члены	Свободные переменные		
		x_6	x_4	x_3
x_2	18	-0.1	0.35	1.25
x_5	112	0.6	-2.6	-4
x_1	24	0.2	-0.2	0
x_7	156	-0.2	-1.3	-0.5
f	492	0.6	2.9	5.5

Все оценки целевой функции положительные, следовательно, получено оптимальное решение, которое находим, приравнивая значения свободных переменных к нулю

$$X^* = (24; 18; 0)$$

То есть, при производстве 24 изделий вида А и 18 изделий вида В предприятие получит максимально возможную прибыль

$$f_{max} = 10x_1^* + 14x_2^* + 12x_3^* = 10 \cdot 24 + 14 \cdot 18 = 492 \text{ ден. ед.}$$

2.4. Экономическая интерпретация решения

Поскольку $x_3^*=0$, изделия вида С при оптимальном плане выпуска продукции не производятся.

Дополнительные переменные $x_4^* = x_6^* = 0$, следовательно общий фонд рабочего времени фрезеровочного и сварочного оборудования расходуется полностью. Фрезеровочное и сварочное оборудование не простаивает.

Ресурсы токарного и шлифовального оборудования недефицитны, при оптимальном плане выпуска продукции их остается соответственно $x_5^* = 112$ и $x_7^* = 156$ врем. ед.

3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ РЕШЕНИЯ ЗАДАЧИ

3.1. Подготовка исходных данных

Поскольку исходных данных в данной задаче достаточно много, а при больших матрицах может доходить до сотен чисел, вводить их с клавиатуры при каждом запуске программы неудобно и трудоемко. При этом при случайных ошибках необходимо повторять весь ввод заново.

Поэтому запишем исходные данные оптимизационой задачи в текстовый файл, а программа, осуществляющая решение задачи, будет считывать их из этого файла. В различных файлах можно хранить исходные данные разных задач.

Определим следующий формат текстового файла.

В первой строке помещаем число изделий и число ресурсов задачи. Целые числа разделяем одним или несколькими пробелами.

В следующих строках помещаем матрицу коэффициентов по условию задачи. Далее в виде столбца располагаем объемы ресурсов. В последней строке находятся коэффициенты при целевой функции.

Вид текстового редактора с введенными в файле date_1.txt исходными данными задачи показан на рисунке 1.

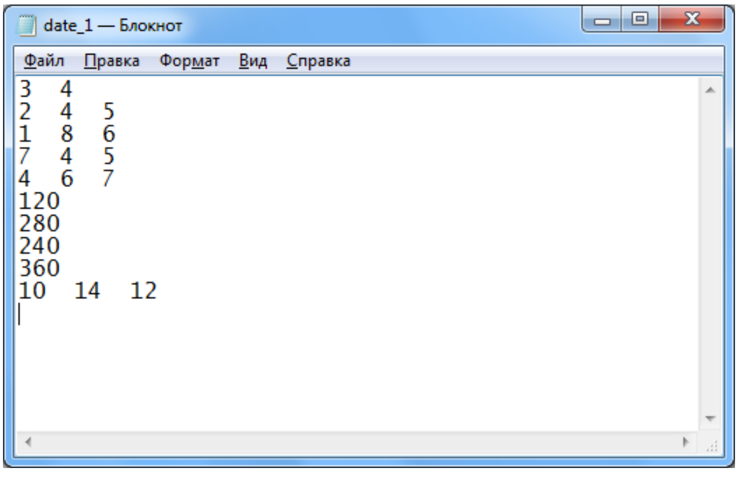

Рис. 1. Подготовка исходных данных в текстовом редакторе.

3.2. Описание интерфейса программы

Поместим на форму программы следующие элементы.

Вверху формы надпись с именем текстового файла с исходными данными задачи. В средней части формы расположим два поля списка ListBox. Первое поле предназначено для контрольного вывода исходных данных, помещенных в текстовый файл. Во второе поле программа будет выводить результаты расчета.

Внизу формы находятся управляющие кнопки "Выбор файла", "Расчет" и "Выход". Вид окна программы показан на рисунке 2.

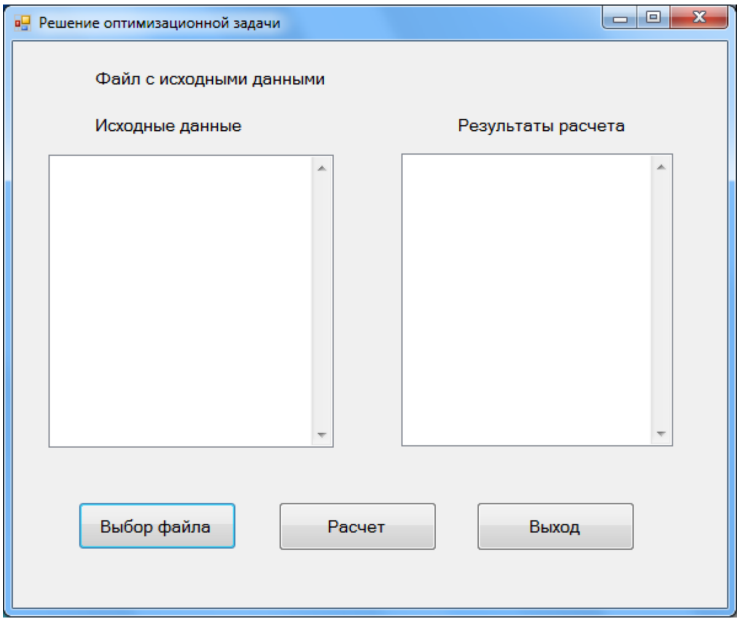

Рис. 2. Вид окна программы решения оптимизационной задачи.

При нажатии на кнопку "Выбор файла" появляется стандартное диалоговое окно открытия файла, в котором можно выбрать необходимый файл на любом доступном носителе персонального компьютера. После выбора файла его имя выводится вверху формы, а содержание – в поле "Исходные данные".

При нажатии на кнопку "Расчет" осуществляется решение задачи с выводом результатов расчета в поле "Результаты расчета".

3.3. Расчет тестовой задачи

Для проверки работоспособности программы выполним расчет аналогичной задачи линейного программирования об оптимальном распределении материалов, решение которой известно [2, стр. 50].

Имеющийся фонд материалов b_i (i=1, 3) нужно так распределить между изготовителями, чтобы получить максимальную прибыль от реализации всей продукции, произведенной из данных материалов. Нормы a_{ij} (i=1, 3; j=1,5) расхода на единицу продукции и прибыль c_j, получаемая от реализованной единицы продукции, представлены в таблице.

Материалы	Продукция					Объем материала
	$П_1$	$П_2$	$П_3$	$П_4$	$П_5$	
b_1	0.7	0.9	1.5	2.3	1.8	50000
b_2	1.4	0.3	0.7	2.5	2	28000
b_3	0.5	2.1	1.8	0.7	2	40000
Прибыль c_j	5	7	6	9	8	
План x	x_1	x_2	x_3	x_4	x_5	

Исходные данные поместим в файл date_2.txt. Окно программы с результатами расчетов показано на рисунке 3.

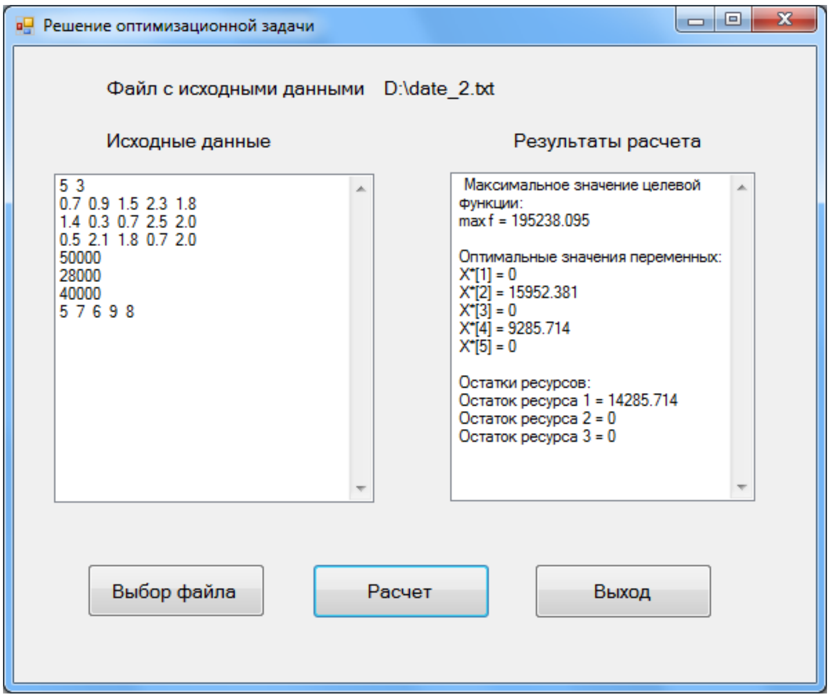

Рис. 3. Вид окна программы решения тестовой задачи.

Результаты расчетов полностью совпадают с результатами решения задачи, приведенными в [2, стр. 52]. Это подтверждает правильность метода решения задачи и написанной в соответствии с ним программы на языке высокого уровня C#.

3.4. Расчет поставленной задачи

Исходные данные для поставленной задачи находятся в файле date_1.txt. Окно программы с результатами расчетов показано на рисунке 4.

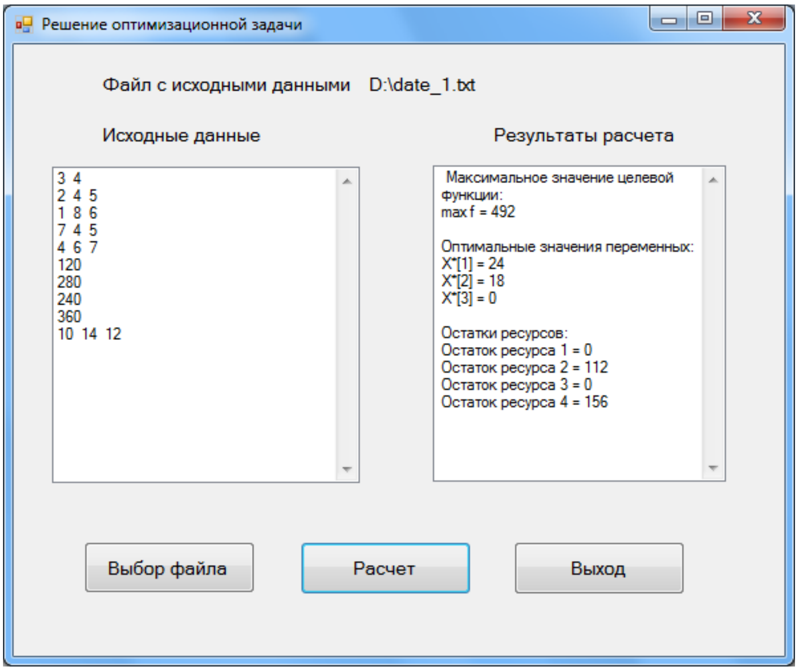

Рис. 4. Вид окна программы решения поставленной задачи.

Полученные результаты полностью совпадают с решением этой задачи симплекс-методом, полученной в пункте 2.3 курсовой работы.

ЗАКЛЮЧЕНИЕ

Изучение и практическое применение методов решения задач оптимизации связано с насущными потребностями планирования и организации производства, способствует принятию управленческих решений. Современная экономическая наука широко использует математические методы как для решения прикладных, практических задач, так и для теоретического моделирования социально-экономических явлений и процессов.

В данной курсовой работе наряду с теоретическим материалом разбирается конкретная оптимизационная задача с целью закрепления основных понятий и математических методов. Решение задачи теоретическим методом и с помощью информационных технологий расчета на персональном компьютере на современном языке высокого уровня C# дает возможность практического освоения процесса и практических навыков.

СПИСОК ЛИТЕРАТУРЫ

1. Кузнецов А. В., Сакович В. А. Высшая математика. Математическое программирование. Минск, "Вышэйшая школа", 2001 г.

2. Кузнецов А. В., Сакович В. А., Холод Н. И. Сборник задач и упражнений по высшей математике. Математическое программирование. Минск, "Вышэйшая школа", 1995 г.

3. Кузнецов Ю. Н. Математическое программирование. Москва "Наука", 1986 г.

4. Минюк С. А., Ровба Е. А., Кузьмич К. К. Математические методы и модели в экономике. Минск, "ТетраСистемс", 2002 г.

5. Павловская Т. А. C# Программирование на языке высокого уровня. – СПб.: Питер, 2010 г.

ПРИЛОЖЕНИЕ

Текст программы на языке C#

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO;

namespace Simplex_metod
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        //************* Кнопка Выбор файла *********************
        private void button1_Click(object sender, EventArgs e)
        {
            var t = openFileDialog1.ShowDialog();
            if (t == System.Windows.Forms.DialogResult.OK)
            {
                label2.Text = openFileDialog1.FileName;
                listBox1.Items.Clear();                         // очистка списка
                // переносим строки текстового файла в listBox1
                try
                {
                    using (var stream = new StreamReader(openFileDialog1.FileName))
                    {   // после выхода поток закрывается
                        string buf;
                        while ((buf = stream.ReadLine()) != null)
                            listBox1.Items.Add(buf);

                    }
                }
                catch (FileNotFoundException exc)
                {
                    MessageBox.Show(exc.Message);
                    return;
                }
                listBox2.Items.Clear();
            }

        }
        //************** Кнопка Выход *******************
        private void button2_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        //**************** Кнопка Расчет ****************

        private void button3_Click(object sender, EventArgs e)
        {
            if (listBox1.Items.Count==0) return;    // если список пуст

            int i, j;
            int nb, nc;                         // число базисных и свободных переменных
            string buf;                         // вспомогательная строковая переменная
            double[,] a = new double[11,11];    // матрица коэффициентов
            double[,] G = new double[11,11];
            double[] b = new double[11];        // объем ресурсов
            double[] T = new double[11];
            double[] f = new double[11];        // коэффициенты целевой функции
            double[] Y = new double[11];
            int[] mc = new int[11];             // номера свободных переменных
            int[] mb = new int[11];             // номера БП
            int flagOpt = 0;                    // план не оптимален
            int cikl = 0;                       // счетчик циклов для защиты от зацикливания
            double f0 = 0;                      // свободный член в целевой функции
            double min;
            int ColCalc, RowCalc=1;             // столбец и строка Разр Элем
```

```
                              // RowCalc определяется в if и компилятор не пропускает
double eps = 1e-10;

// из нулевой строки списка listBox1 вводим целые nc, nb
buf = listBox1.Items[0].ToString();
var mass = buf.Split(new[] { ' ' }, StringSplitOptions.RemoveEmptyEntries)
    .Select(s => Convert.ToInt32(s))
    .ToArray();
nc = mass[0];
nb = mass[1];
                        // контрольный вывод listBox2.Items.Add("nc="+nc.ToString());
// из 1 - nb строк вводим действительную матрицу a
for (i=1; i<=nb; i++)
{
    buf = listBox1.Items[i].ToString();
     var mas = buf.Split(new[] { ' ' }, StringSplitOptions.RemoveEmptyEntries)
        .Select(s => Convert.ToDouble(s))
        .ToArray();
     for (j=1; j<=nc; j++) a[i,j] = mas[j-1];

}
     // контрольный вывод listBox2.Items.Add("a[nb,nc]=" + a[nb, nc].ToString());

// из nb+1 - nb+nb строк вводим действительный вектор b
for (j=nb+1; j<=nb+nb; j++)
     b[j-nb] = Convert.ToDouble( listBox1.Items[j].ToString());


// из nb+nb+1 строки вводим действительный вектор f с заменой знаков
buf = listBox1.Items[nb + nb + 1].ToString();
var mass2 = buf.Split(new[] { ' ' }, StringSplitOptions.RemoveEmptyEntries)
        .Select(s => Convert.ToDouble(s))
        .ToArray();
for (j=1; j<=nc; j++) f[j] = -mass2[j-1];

// Подготовка номеров БП и СП
for (j = 1; j <= nc; j++) mc[j] = j;
for (i = 1; i <= nb; i++) mb[i] = i+nc;

// цикл пересчета симплекс-таблицы
while ((flagOpt == 0) && (cikl<40))
{
    // поиск разрешающего столбца
    min = f[1];
    ColCalc = 1;
    for (j = 2; j <= nc; j++)
        if (f[j] < min)
        {
            min = f[j];
            ColCalc = j;
        }

    // поиск разрешающей строки по минимальному симпл. отношению
    for (i = 1; i <= nb; i++)
        if (a[i, ColCalc] > eps) min = b[i] / a[i, ColCalc];
    for (i = 1; i <= nb; i++)
        if ((a[i, ColCalc] > eps) && (b[i]/a[i,ColCalc]<=min))
        {
            min = b[i] / a[i, ColCalc];
            RowCalc = i;
        }

    // Пересчет симплекс-таблицы

    double E=(1.0)/a[RowCalc,ColCalc];
    i=mb[RowCalc];                      // обмен БП и СП
    mb[RowCalc]=mc[ColCalc];
    mc[ColCalc]=i;

    for (i = 1; i <= nb; i++) T[i]=b[i]-b[RowCalc]*a[i,ColCalc]*E;
    T[RowCalc]=b[RowCalc]*E;

     for (i = 1; i <= nb; i++)
          for (j = 1; j <= nc; j++)
              if ((i!=RowCalc) || (j!=ColCalc))
                  G[i,j]=a[i,j]-a[RowCalc,j]*a[i,ColCalc]*E;


     for (i = 1; i <= nb; i++) G[i,ColCalc]=-a[i,ColCalc]*E;    // разреш. столбец
```

```
                for (j = 1; j <= nc; j++)
                {
                    Y[j]=f[j]-f[ColCalc]*a[RowCalc,j]*E;       // инд строка
                    G[RowCalc,j]=a[RowCalc,j]*E;               // разреш. строка
                }

                f0=f0-f[ColCalc]*b[RowCalc]*E;
                G[RowCalc,ColCalc]=E;
                Y[ColCalc]=-f[ColCalc]*E;

                // Обновляем матрицы
                for (i = 1; i <= nb; i++) b[i]=T[i];
                for (i = 1; i <= nb; i++)
                    for (j = 1; j <= nc; j++) a[i,j] = G[i,j];

                for (j = 1; j <= nc; j++) f[j]=Y[j];

                // анализ матрицы на оптимальность с установкой флага
                cikl++;
                min = f[1];
                for (j = 2; j <= nc; j++)
                    if (f[j] < min)  min = f[j];

                if (min>-eps) flagOpt=1;

            }

            // вывод результата поиска
            listBox2.Items.Add("  Максимальное значение целевой");
            listBox2.Items.Add(" функции:");
            listBox2.Items.Add(" max f = " + f0.ToString("0.###"));
            listBox2.Items.Add(" ");
            listBox2.Items.Add(" Оптимальные значения переменных:");
            for (j = 1; j <= nc; j++)
            {
                for (i = 1; i <= nb; i++)
                    if (mb[i] == j) listBox2.Items.Add(" X*[" + j.ToString() + "] = " +
b[i].ToString("0.###"));
                for (i = 1; i <= nc; i++)
                    if (mc[i] == j) listBox2.Items.Add(" X*[" + j.ToString() + "] = 0");
            }
            listBox2.Items.Add(" ");
            listBox2.Items.Add(" Остатки ресурсов:");
            for (j = 1; j <= nb; j++)
            {
                for (i = 1; i <= nb; i++)
                    if (mb[i] == j + nc) listBox2.Items.Add(" Остаток ресурса " + (mb[i] -
nc).ToString() + " = " + b[i].ToString("0.###"));
                for (i = 1; i <= nc; i++)
                    if (mc[i] == j + nc) listBox2.Items.Add(" Остаток ресурса " + (mc[i] -
nc).ToString() + " = 0");
            }
        }
    }
}
```