

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА ИНФОРМАТИКИ И ПРИКЛАДНОЙ МАТЕМАТИКИ

## ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по дисциплине «Информатика» для студентов  
строительных специальностей очной формы обучения  
третьего семестра

*издание 2-е, дополненное и переработанное*

Строительный факультет

Группа \_\_\_\_\_

Фамилия \_\_\_\_\_

Имя \_\_\_\_\_

Отчество \_\_\_\_\_

Вариант \_\_\_\_\_

м/тел. +375 ( ) \_\_\_\_\_ - \_\_\_\_\_ - \_\_\_\_\_

УДК 004

Практикум предназначен для студентов второго курса строительного факультета, изучающих дисциплину «Информатика». В него входят задания для лабораторных работ и задания для их защиты. В теоретической части изложены методические рекомендации по работе с СКМ Maple, MathCAD и электронными таблицами MS EXCEL.

Составители: В.А. Кофанов, к.т.н., доцент  
Т.Г. Хомицкая, ст. преподаватель  
С.В. Сидак, ассистент

Рецензенты: зав. кафедрой прикладной математики и информатики БрГУ имени А.С. Пушкина, доцент, к.ф.-м.н. О.В. Матысик;  
директор Брестского филиала ИООО EPAM Systems,  
доцент, к.ф.-м.н. С.А. Тузик.

## Общие указания

Практикум предназначен для организации самостоятельной, практической и лабораторной работы студентов в первом семестре изучения дисциплины «Информатика».

Перед началом работы необходимо привести информацию на титульном листе, то есть должны быть указаны данные ее владельца (Ф.И.О., группа, номер мобильного телефона).

На текущей странице в блок «Индивидуальное задание» необходимо вклеить листок с вариантами заданий, полученный у преподавателя. Специальный блок «Условие» на странице с лабораторной работой заполняется простым переписыванием условия задачи с бланка индивидуального задания.

При выполнении лабораторных работ результаты вычислений заносятся в отчет в том же виде, в котором они отображаются на экране монитора. Ведение записей выполняется четко и разборчиво шариковой ручкой (блок-схемы – карандашом). Неправильные (ошибочные) записи на страницах практикума необходимо исправлять с использованием корректирующих средств (корректирующие ленты, штрих-корректоры и т.п.).

Каждая лабораторная работа считается выполненной только при наличии отметки преподавателя о ее защите, подтвержденной его подписью (личной печатью) на странице 4. Данные из этого листа служат основанием для допуска к итоговым испытаниям (зачету, экзамену).

### Индивидуальное задание

Место для  
индивидуального  
задания

### Отметки о защите лабораторных работ

№	Наименование работы	Защита	Примечание
1	Лабораторная работа №1(1,2)		
2	Лабораторная работа №2		
3	Лабораторная работа №3		
4	Лабораторная работа №4(1,2)		
5	Лабораторная работа №5		
6	Лабораторная работа №6		

#### Итог работы в семестре:

Все лабораторные работы выполнены в полном объеме.

---

(дата, подпись)

# Методические указания к выполнению лабораторных работ

## СИСТЕМА КОМПЬЮТЕРНОЙ МАТЕМАТИКИ MAPLE

Решение дифференциальных уравнений самых различных типов – одно из достоинств системы Maple. Для этой цели могут быть использованы функции ядра системы, инструментальные пакеты для решения обыкновенных дифференциальных уравнений и для решения дифференциальных уравнений в частных производных. Кроме этого существует множество функций, позволяющих классифицировать уравнения, производить замены в дифференциальных уравнениях, осуществлять преобразование решений, визуализировать полученные решения.

### Аналитическое решение дифференциального уравнения первого порядка

Для нахождения аналитических решений дифференциальных уравнений в Maple применяется команда **dsolve**. Синтаксис этой команды зависит от вида дифференциальных уравнений. В случае решения дифференциального уравнения первого порядка вид команды **dsolve** будет следующий:

- для нахождения общего решения

**dsolve(ODE, Fun, Options);**

Использование знака апострофа «'»	Использование команды <i>diff</i>
первый возможный вариант записи команд	
$\left[ \begin{array}{l} > \text{dsolve}(x'(t) = 2 + 0.6 \cdot t, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t + \_C1 \end{array} \right]$	$\left[ \begin{array}{l} > \text{dsolve}(\text{diff}(x(t), t) = 2 + 0.6 \cdot t, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t + \_C1 \end{array} \right]$
второй возможный вариант записи команд	
$\left[ \begin{array}{l} > \text{ODE} := x'(t) = 2 + 0.6 \cdot t \\ & \text{ODE} := D(x)(t) = 2 + 0.6 t \\ > \text{dsolve}(\text{ODE}, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t + \_C1 \end{array} \right]$	$\left[ \begin{array}{l} > \text{ODE} := \text{diff}(x(t), t) = 2 + 0.6 \cdot t \\ & \text{ODE} := \frac{d}{dt} x(t) = 2 + 0.6 t \\ > \text{dsolve}(\text{ODE}, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t + \_C1 \end{array} \right]$

- для нахождения частного решения

**dsolve({ODE, IC}, Fun, Options),**

Использование знака «'»	Использование команды <i>diff</i>
первый возможный вариант записи команд	
$\left[ \begin{array}{l} > \text{dsolve}(\{x'(t) = 2 + 0.6 \cdot t, x(0) = -4\}, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t - 4 \end{array} \right]$	$\left[ \begin{array}{l} > \text{dsolve}(\{\text{diff}(x(t), t) = 2 + 0.6 \cdot t, x(0) = -4\}, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t - 4 \end{array} \right]$
второй возможный вариант записи команд	
$\left[ \begin{array}{l} > \text{ODE} := x'(t) = 2 + 0.6 \cdot t \\ & \text{ODE} := D(x)(t) = 2 + 0.6 t \\ > \text{dsolve}(\{\text{ODE}, x(0) = -4\}, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t - 4 \end{array} \right]$	$\left[ \begin{array}{l} > \text{ODE} := \text{diff}(x(t), t) = 2 + 0.6 \cdot t \\ & \text{ODE} := \frac{d}{dt} x(t) = 2 + 0.6 t \\ > \text{dsolve}(\{\text{ODE}, x(0) = -4\}, x(t)) \\ & x(t) = \frac{3}{10} t^2 + 2t - 4 \end{array} \right]$

где **ODE** – дифференциальное уравнение;

**IC** – известное условие;

**Fun** – неизвестная функция, входящая в дифференциальное уравнение **ODE**;

**Options** – дополнительные (необязательные) параметры.

При составлении дифференциальных уравнений для обозначения производной используется команда **diff**. В случае если функция зависит от одной переменной можно использовать для сокращения записи знак апострофа «'». В последующих примерах для обозначения производной функции будем использовать, где это уместно, указанный символ.

Общее решение дифференциального уравнения зависит от произвольных постоянных, число которых равно порядку дифференциального уравнения. В Maple такие постоянные, как правило, обозначаются как **\_C1**, **\_C2**, и т.д.

### Решение дифференциального уравнения второго порядка

Для нахождения аналитических решений дифференциальных уравнений второго порядка команда **dsolve** записывается так:

- для нахождения общего решения

**dsolve(ODE, Fun, Options);**

Использование знака апострофа «'»	Использование команды <i>diff</i>
первый возможный вариант записи команд	
<pre>&gt; dsolve(y''(t) + y'(t) = 0.6*t, y'(t))       y(t) = sin(t) _C2 + cos(t) _C1 + 3/5 t</pre>	<pre>&gt; dsolve(diff(y(t), t\$2) + y'(t) = 0.6*t, y'(t))       y(t) = sin(t) _C2 + cos(t) _C1 + 3/5 t</pre>
второй возможный вариант записи команд	
<pre>&gt; ODE := y''(t) + y'(t) = 0.6*t       ODE := D^(2)(y)(t) + y'(t) = 0.6 t &gt; dsolve(ODE, y'(t))       y(t) = sin(t) _C2 + cos(t) _C1 + 3/5 t</pre>	<pre>&gt; ODE := diff(y(t), t\$2) + y'(t) = 0.6*t       ODE := d^2/dt^2 y(t) + y'(t) = 0.6 t &gt; dsolve(ODE, y'(t))       y(t) = sin(t) _C2 + cos(t) _C1 + 3/5 t</pre>

- для нахождения частного решения

**dsolve({ODE, ICs}, Fun, Options),**

Использование знака «'»	Использование команды <i>diff</i> и оператора <i>D</i>
<pre>&gt; ODE := y''(t) + y'(t) = 0.6*t       ODE := D^(2)(y)(t) + y'(t) = 0.6 t &gt; ICs := y(0) = 0.5, y'(0) = 0.7       ICs := y(0) = 0.5, D(y)(0) = 0.7 &gt; dsolve({ODE, ICs}, y'(t))       y(t) = 1/10 sin(t) + 1/2 cos(t) + 3/5 t</pre>	<pre>&gt; ODE := diff(y(t), t\$2) + y'(t) = 0.6*t       ODE := d^2/dt^2 y(t) + y'(t) = 0.6 t &gt; ICs := y(0) = 0.5, D(y)(0) = 0.7       ICs := y(0) = 0.5, D(y)(0) = 0.7 &gt; dsolve({ODE, ICs}, y'(t))       y(t) = 1/10 sin(t) + 1/2 cos(t) + 3/5 t</pre>

где **ODE** – дифференциальное уравнение;

**ICs** – известные условия;

**Fun** – неизвестная функция, входящая в дифференциальное уравнение **ODE**;

**Options** – дополнительные (необязательные) параметры.

При обозначении второй производной функции знак апострофа «'» набирается дважды (не путать со знаком парной кавычки «"»).

Если помимо дифференциального уравнения задать соответственно начальные или краевые условия для неизвестной функции, то команда **dsolve** находит решение задачи Коши или краевой задачи. Для обозначения первых производных в начальных или краевых условиях используется дифференциальный оператор **D** в следующем формате: **D(имя\_функции)(значение\_аргумента)**.

## Решение системы дифференциальных уравнений

Команда **dsolve** может найти решение системы дифференциальных уравнений (или задачи Коши), если в ней указать:

- для нахождения общего решения

**dsolve({ODEs},{Funs},Options);**

- для нахождения частного решения

**dsolve({ODEs,ICs},{Funs},Options),**

где **ODEs** – дифференциальные уравнения,

**ICs** – известные условия;

**Funs** – набор неизвестных функций, входящих в дифференциальные уравнения **ODEs**;

**Options** – дополнительные (необязательные) параметры.

Пример решения системы дифференциальных уравнений показан на рисунке 1.

```

> ODEs := v'(t) = 0.8*t - 2*u(t), u'(t) = 0.2*t + 0.5*v(t)
      ODEs := D(v)(t) = 0.8*t - 2*u(t), D(u)(t) = 0.2*t + 0.5*v(t)
Общее решение системы дифференциальных уравнений
> dsolve({ODEs}, {v(t), u(t)})
      {u(t) = sin(t) _C2 + cos(t) _C1 + 1/5 + 2/5*t, v(t) = 2*cos(t) _C2 - 2*sin(t) _C1 + 4/5 - 2/5*t}
> ICs := v(0) = 0.3, u(0) = -0.4
      ICs := v(0) = 0.3, u(0) = -0.4
Частное решение системы дифференциальных уравнений
> dsolve({ODEs, ICs}, {v(t), u(t)})
      {u(t) = -1/4*sin(t) - 3/5*cos(t) + 1/5 + 2/5*t, v(t) = -1/2*cos(t) + 6/5*sin(t) + 4/5 - 2/5*t}
    
```

Рисунок 1 – Пример решения системы дифференциальных уравнений

Результат отображается в виде вектора из двух элементов. В первом элементе содержится найденная функция  $u(t)$ , во втором элементе – функция  $v(t)$ . Порядок представления функций не всегда совпадает с порядком, указанным в команде **dsolve**.

## Численное решение дифференциального уравнения второго порядка

Для того чтобы найти численное решение дифференциального уравнения (задачи Коши или краевой задачи) в команде **dsolve** следует указать параметр **type=numeric** (или просто **numeric**). Тогда команда решения дифференциального уравнения будет иметь вид:

**dsolve({ODE,ICs},Fun,numeric,output=listprocedure),**

где **ODE** – дифференциальное уравнение;

**ICs** – известные условия;

**Fun** – неизвестная функция, входящая в дифференциальное уравнение **ODE**;

**numeric** – параметр, указывающий на численное решение дифференциального уравнения;

**output** – параметр, определяющий форму конечного результата (значение **listprocedure** указывает на то, что конечный результат в последствии будет использован командой **solve**).

Пример численного решения нелинейного дифференциального уравнения показан на рисунке 2.

```

> ODE := y''(t) - 0.8*y(t)^2 + y(t) = 0.3*sin(t)
      ODE := D(2)(y)(t) - 0.8*y(t)^2 + y(t) = 0.3*sin(t)
> ICs := y(0) = 0.2, y'(0) = -0.4
      ICs := y(0) = 0.2, D(y)(0) = -0.4
> dsolve({ODE, ICs}, y(t), numeric, output = listprocedure)
      [t = proc(t) ... end proc, y(t) = proc(t) ... end proc, d/dt y(t) = proc(t) ... end proc]
    
```

Рисунок 2 – Пример численного решения нелинейного дифференциального уравнения

## Проверка решения дифференциального уравнения

Важным этапом в решении дифференциальных уравнений является этап проверки найденного решения. Для выполнения этой проверки в Maple используется команда **odetest** в виде:

**odetest(sol,[ODE,ICs]),**

где **sol** – решение дифференциального уравнения или их системы (результат команды **dsolve**);

**ODE** – дифференциальное уравнение или их система;

**ICs** – известные условия.

Примеры использования данной команды приведены на рисунках 3-5.

```
> ODE := diff(x(t), t) = 2 + 0.6*t
      ODE := d/dt x(t) = 2 + 0.6 t
> dsolve({ODE, x(0) = -4}, x(t))
      x(t) = 3/10 t^2 + 2 t - 4
> odetest(%, [ODE, x(0) = -4])
      [0, 0]
```

Рисунок 3 – Проверка решения дифференциального уравнения первого порядка

```
> ODE := y''(t) + y'(t) = 0.6*t
      ODE := D^(2)(y)(t) + y'(t) = 0.6 t
> ICs := y(0) = 0.5, y'(0) = 0.7
      ICs := y(0) = 0.5, D(y)(0) = 0.7
> dsolve({ODE, ICs}, y(t))
      y(t) = 1/10 sin(t) + 1/2 cos(t) + 3/5 t
> odetest(%, [ODE, ICs])
      [0, 0, 0]
```

Рисунок 4 – Проверка решения дифференциального уравнения второго порядка

```
> ODEs := v'(t) = 0.8*t - 2*u(t), u'(t) = 0.2*t + 0.5*v(t)
      ODEs := D(v)(t) = 0.8 t - 2 u(t), D(u)(t) = 0.2 t + 0.5 v(t)
> ICs := v(0) = 0.3, u(0) = 0.4
      ICs := v(0) = 0.3, u(0) = 0.4
> dsolve({ODEs, ICs}, {v(t), u(t)})
      {u(t) = 1/4 sin(t) - 3/5 cos(t) + 1/5 + 2/5 t, v(t) = 1/2 cos(t) + 6/5 sin(t) + 4/5 - 2/5 t}
> odetest(%, [ODEs, ICs])
      [0, 0, 0, 0]
```

Рисунок 5 – Проверка решения системы дифференциальных уравнений первого порядка

Команда **odetest** подставляет все функции, указанные с помощью знака «%» в первой позиции, во все уравнения, указанные во второй позиции, и выводит результат невязки по каждому из них в виде вектора. Так на рисунке 3 в ответе команды **odetest** первый ноль обозначает невязку уравнения, отмеченного переменной **ODE**, а второй ноль – невязку уравнения, обозначающего начальное условие  $x(0)=-4$ .

Знак «%» в примерах на рисунках 3-5 обозначает результат выполнения предыдущей команды, т.е. в данном контексте это результат команды **odesolve**, который подставляется в параметры команды **odetest**.

Далее на рисунке 4 в ответе команды **odetest** первый ноль обозначает невязку уравнения, отмеченного переменной **ODE**, а второй и третий ноль – невязку уравнений, обозначающих начальные условия (**ICs**).

На рисунке 5 в ответе команды **odetest** первые два нуля обозначают невязку уравнений, отмеченных переменной **ODEs**, а вторые два нуля – невязку уравнений, обозначающих начальные условия (**ICs**).

Иногда из-за неверных записей выражений команда **odetest** может выводить вместо замечаний ответ, который говорит о том, что дифференциальное уравнение решено верно. Поэтому наиболее надежным способом проверки решения дифференциального уравнения является способ подстановки найденной функции в начальное дифференциальное уравнение (рисунок 6).

```
> ODE := y''(t) + y'(t) = 0.6*t
      ODE := D^(2)(y)(t) + y'(t) = 0.6 t
> ICs := y(0) = 0.5, y'(0) = 0.7
      ICs := y(0) = 0.5, D(y)(0) = 0.7
> dsolve({ODE, ICs}, y(t))
      y(t) = 1/10 sin(t) + 1/2 cos(t) + 3/5 t
> y := unapply(rhs(%), t)
      y := t -> 1/10 sin(t) + 1/2 cos(t) + 3/5 t
> y''(t) + y'(t) - 0.6*t
      0.
```

Рисунок 6 – Проверка решения дифференциального уравнения второго порядка

## Построение графика решения дифференциального уравнения

Команда `dsolve` выдает решение дифференциального уравнения в невычисляемом формате. Для того чтобы с решением можно было бы работать далее (например, построить график решения, найти значение функции на конце отрезка) следует придерживаться следующих правил.

1. Если в области вывода расположена одна функция, то для ее отображения вначале необходимо отделить правую часть полученного решения командой `rhs` и определить результат как функцию с помощью команды `unapply`. Затем построить график командой `plot` (рисунок 7).

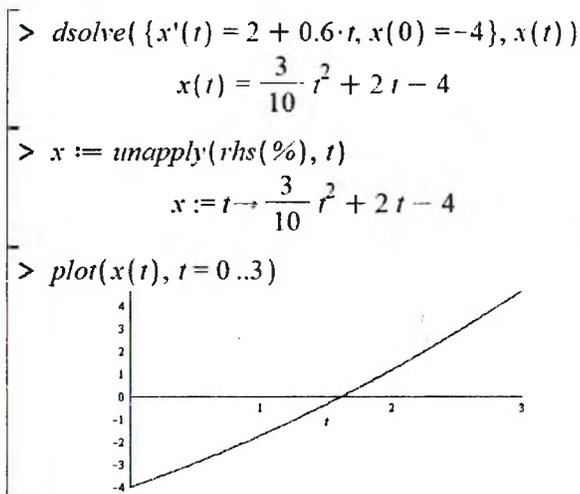


Рисунок 7 – Вывод графика решения дифференциального уравнения

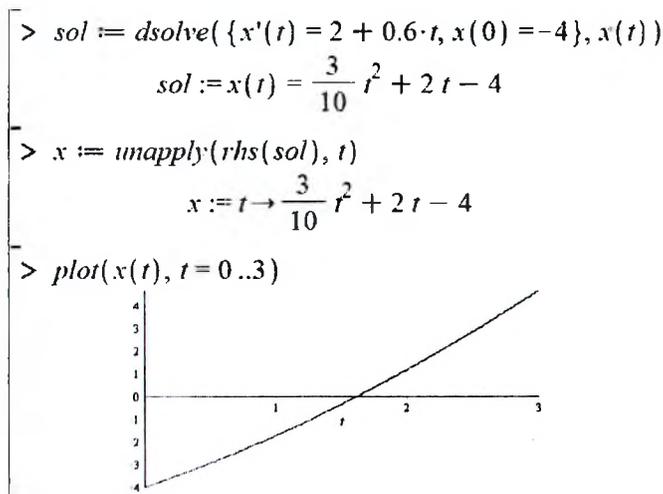


Рисунок 8 – Вывод графика решения дифференциального уравнения

Использование знака «%» в командах Maple не всегда рационально и не приспосабливается в случае изменения пользователем порядка выполнения команд. Наиболее предпочтительным способом является предварительное присвоение результата решения дифференциального уравнения некоторой переменной, которую в дальнейшем можно использовать вместо знака «%» (рисунок 8).

2. Если необходимо отобразить график производной функции, то вначале нужно определить саму функцию как на рисунках 7 и 8, а затем определить функцию ее производную с помощью команды `unapply`. После этого построить график командой `plot` (рисунок 9).

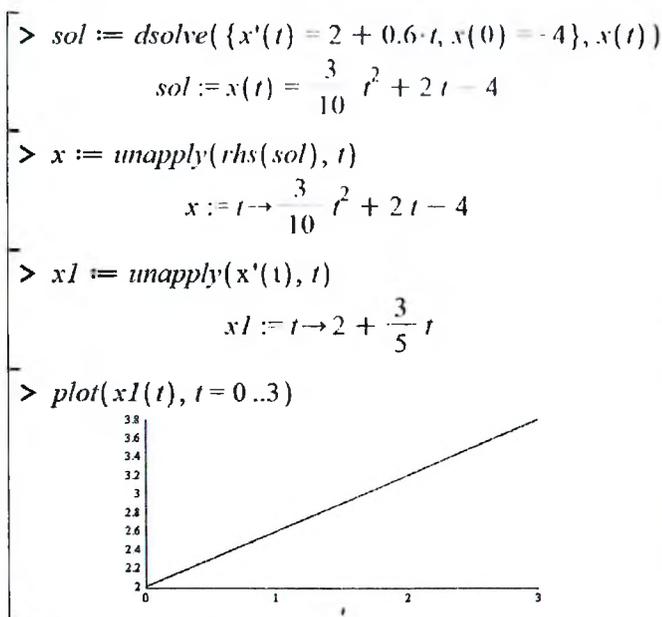


Рисунок 9 – Вывод графика производной решения дифференциального уравнения

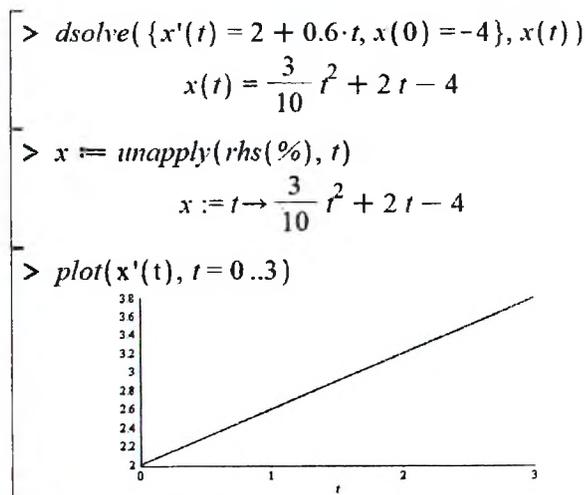


Рисунок 10 – Вывод графика производной решения дифференциального уравнения

В Maple имеется возможность построения производной функции без ее определения. Для этого достаточно в параметрах команды `plot` указать производную функции с помощью знака апострофа «'» (рисунок 10).

3. Если в области вывода расположены две функции в виде массива (рисунок 11), то для отображения одной из них вначале необходимо извлечь результат полученного решения из элемента массива, отделить пра-

вую часть командой **rhs** и определить результат как функцию с помощью команды **unapply**. Затем построить график командой **plot** (рисунок 12).

4. Для отображения двух функции с уникальными параметрами на одном графике, необходимо предварительно определить эти функции, а затем использовать команду **display** из пакета функций **plots** (рисунок 13) в формате: **display({grafs},Options)**, где **grafs** – набор графических структур, созданных, например, с помощью команды **plot**; **Options** – параметры (необязательные) графической области, общей для всех указанных в наборе структур.

```
> sol := dsolve({v'(t) = 0.8*t - 2*u(t), u'(t) = 0.2*t + 0.5*v(t), v(0) = 0.3, u(0) = -0.4}, {v(t), u(t)})
sol := {u(t) = 1/4 sin(t) - 3/5 cos(t) + 1/5 + 2/5 t, v(t) = -1/2 cos(t) + 6/5 sin(t) + 4/5 - 2/5 t}
```

Рисунок 11 – Решение системы дифференциальных уравнений

```
Продолжение. Начало на рисунке 11
> u := unapply(rhs(sol[1]), t)
u := t -> -1/4 sin(t) - 3/5 cos(t) + 1/5 + 2/5 t
> v := unapply(rhs(sol[2]), t)
v := t -> -1/2 cos(t) + 6/5 sin(t) + 4/5 - 2/5 t
> plot({u(t), v(t)}, t = 0..4)
```

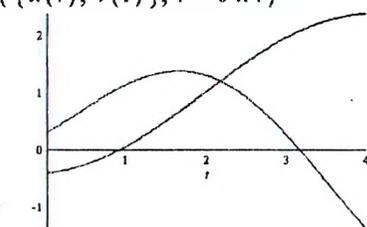


Рисунок 12 – Вывод двух функций в одной графической области

```
Продолжение. Начало на рисунке 12
> with(plots):
> plotU := plot(u(t), t = 0..2, y = -1.5..2.5)
plotU := PLOT(...)
> plotV := plot(v(t), t = 2..4, y = -1.5..2.5)
plotV := PLOT(...)
> display({plotU, plotV})
```

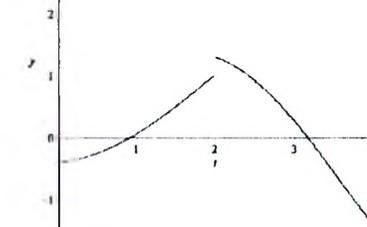


Рисунок 13 – Вывод двух функций с уникальными параметрами в одной графической области

6. В случае если результат численного решения дифференциального уравнения был получен посредством указания в команде **dsolve** значения параметра **output** как **listprocedure** (рисунок 2), то для определения функции можно использовать команду **subs** в формате: **subs(x=a,expr)** где **x=a** – набор (разделяется запятыми) заданных переменных для подстановки в выражение **expr** (несколько выражений разделяются запятыми внутри фигурных скобок). Затем построить график командой **plot** (рисунок 14).

```
> ODE := y''(t) - 0.8*y(t)^2 + y(t) = 0.3*sin(t)
ODE := D^(2)(y)(t) - 0.8*y(t)^2 + y(t) = 0.3 sin(t)
> ICs := y(0) = 0.2, y'(0) = -0.4
ICs := y(0) = 0.2, D(y)(0) = -0.4
> sol := dsolve({ODE, ICs}, y(t), numeric, output = listprocedure)
sol := | t = proc(t) ... end proc, y(t) = proc(t) ... end proc, d/dt y(t) = proc(t) ... end proc |
> y := subs(sol, y(t))
y := proc(t) ... end proc
> plot(y(t), t = 0..5)
```

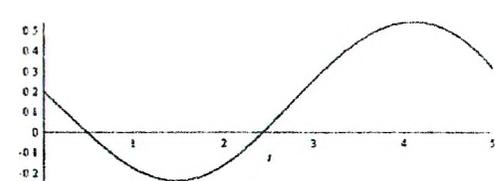


Рисунок 14 – Пример численного решения нелинейного дифференциального уравнения

7. Для отображения параметрического графика (рисунок 15), представленного двумя функциями, вначале необходимо определить эти функции, как показано на рисунках 7-10 и 12, а затем использовать команду `plot`.

8. Для отображения параметрического графика (рисунок 16), представленного тремя функциями, вначале необходимо определить эти функции, как показано на рисунках 7-10 и 12, а затем использовать команду `spacecurve` из пакета функций `plots`.

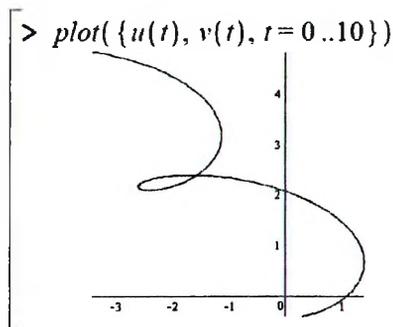


Рисунок 15 – Вывод параметрического графика

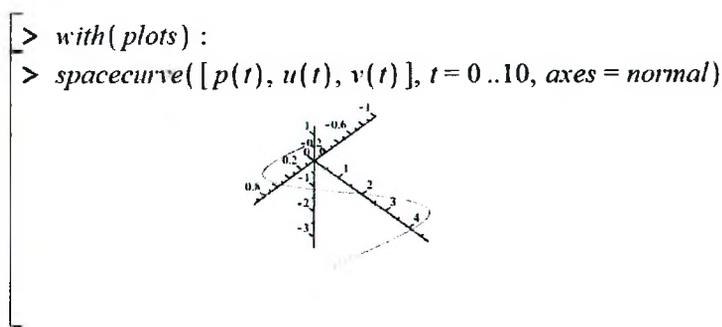


Рисунок 16 – Вывод параметрического графика в пространственной системе координат

Дополнительные параметры используемых выше команд для отображения графиков, а также примеры использования некоторых команд, часто встречающихся в ходе решения дифференциальных уравнений, рассмотрены в следующем пункте.

### Вспомогательные команды

#### 1. `unapply`.

В Maple имеется несколько способов задания пользовательской функции. Рассмотрим два из них.

Первый способ. Определение функции с помощью функционального оператора (рисунок 17), шаблон которого находится на левой боковой панели `Expression` рабочего окна Maple. Формат шаблона следующий:

имя\_функции:=(аргументы) → выражение

Аргументы перечисляются через запятую.

Второй способ. С помощью команды `unapply` (рисунок 17). Формат команды следующий:

имя\_функции:=`unapply`(выражение, аргументы)

Обращение к этой функции осуществляется наиболее привычным в математике способом, когда в скобках вместо аргументов функции указываются конкретные значения переменных.

#### 2. `evalf`.

Чтобы получить приближенное значение в виде числа с плавающей запятой, следует использовать команду `evalf(expr,[t])`, где `expr` – выражение, `t` – количество значащих цифр (рисунок 17).

#### 3. `solve`, `fsolve`.

Для символического решения уравнений в Maple существует универсальная команда `solve(eq, x)`, где `eq` – уравнение, `x` – переменная, относительно которой уравнение надо разрешить. Системы уравнений решаются с помощью такой же команды `solve({eq1,eq2,...},{x1,x2,...})`, только теперь в параметрах команды следует указывать в первых фигурных скобках через запятую уравнения, а во вторых фигурных скобках перечисляются через запятую переменные, относительно которых требуется решить систему (рисунок 18).

Для численного решения уравнений, в тех случаях, когда трансцендентные уравнения не имеют аналитических решений, используется специальная команда `fsolve(eq, x)`, параметры которой такие же, как и команды `solve` (рисунок 18).

```
> f := (x, y) → 2·x + 5·y
      f := (x, y) → 2·x + 5·y
> f := unapply(2·x + 5·y, x, y)
      f := (x, y) → 2·x + 5·y
> pf := f(1/3, 5/8)
      pf := 91/24
> evalf(pf, 5)
      3.7917
```

Рисунок 17 – Пример определения и вычисления функции

```
> solve(5·x - 23 = 0, x)
      23
      5
> solve({2·x + 3·y = 10, 5·x - 2·y = 15}, {x, y})
      {x = 65/19, y = 20/19}
> fsolve(5·x - 23 = 0, x)
      4.600000000
> fsolve({2·x + 3·y = 10, 5·x - 2·y = 15}, {x, y})
      {x = 3.421052632, y = 1.052631579}
```

Рисунок 18 – Пример решения системы уравнений

С помощью команды **fsolve** можно определить место нахождения нуля и точки экстремума функции. Формат команды следующий: **fsolve(eq, x=a..b)**, где **eq** – уравнение, **x** – переменная, относительно которой уравнение надо разрешить, **a** и **b** – диапазон поиска. Фактически поиск характерных точек сводится к решению одного уравнения с одной неизвестной (рисунок 19).

В качестве уравнения для определения нуля функции  $f(x)$  используется уравнение вида  $f(x)=0$ , для определения экстремума –  $f'(x)=0$ . Диапазон поиска обязательно должен быть таков, чтобы в него попадал только один нуль функции (в случае поиска нуля функции), или только один экстремум (если определяется экстремум функции).

```
> f := x -> sin(x) : f1 := x -> diff(f(x)) :
> fsolve(f(x) = 0, x = 3*Pi/4 .. 5*Pi/4)
3.141592654
> fsolve(f1(x) = 0, x = Pi/4 .. 3*Pi/4)
1.570796327
```

Рисунок 19 – Определение нуля и экстремума функции

#### 4. plot

Для построения графиков функции  $f(x)$  одной переменной (в интервале  $a \leq x \leq b$  по оси  $Ox$  и в интервале  $c \leq y \leq d$  по оси  $Oy$ ) используется команда

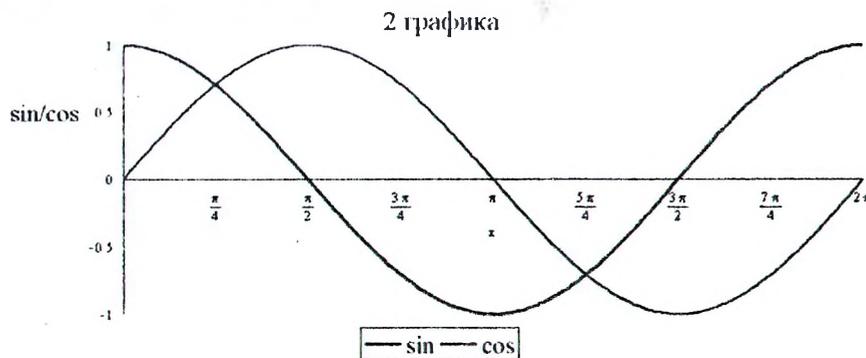
```
plot(f(x), x=a..b, y=c..d, Options),
```

где **Options** – параметры управления изображением. Если параметры не указывать, то будут использованы установки по умолчанию.

Основные параметры команды **plot**:

- **title="text"**, где **text** – заголовок рисунка;
- **coords=polar** – установка полярных координат (по умолчанию установлены декартовы);
- **color=n** – установка цвета линии: английское название цвета, например, **yellow** – желтый и т.д.;
- **thickness=n**, где **n=1,2,3...** – толщина линии (по умолчанию **n=1**);
- **labels=[tx,ty]** – надписи по осям координат: **tx** – по оси  $Ox$  и **ty** – по оси  $Oy$ ;
- **legend="s"**, где **"s"** – название функционального ряда;
- **axes** – установка типа координатных осей: **axes=normal** – обычные оси; **axes=boxed** – график в рамке со шкалой; **axes=frame** – оси с центром в левом нижнем углу рисунка.

```
> plot([sin(x), cos(x)], x=0..2*Pi, y=-1..1, title="2 графика", color=[blue, red], thickness=[2, 3], labels=[ "x", "sin/cos"], legend=[ "sin", "cos"])
```



```
> plot([sin(x), cos(x)], x=-0.5*Pi..0.5*Pi, title="параметрический график", color=red, thickness=2, labels=[ "sin", "cos"], legend=[ "окружность"])
```

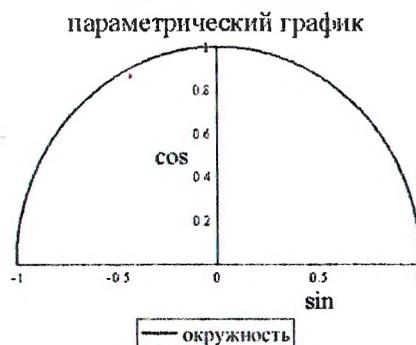


Рисунок 20 – Примеры использования дополнительных параметров в команде **plot**

## ЭЛЕКТРОННАЯ ТАБЛИЦА MICROSOFT EXCEL

Линейное программирование (ЛП) – область математики, разрабатывающая теорию и численные методы решения задач нахождения экстремума (максимума или минимума) линейной функции многих переменных при наличии линейных ограничений, т. е. линейных равенств или неравенств, связывающих эти переменные. К задачам линейного программирования сводится широкий круг вопросов планирования экономических процессов, где ставится задача поиска наилучшего (оптимального) решения.

Общая задача линейного программирования (ЗЛП) состоит в нахождении экстремального значения (максимума или минимума) линейной функции.

### Решение ЗЛП с помощью надстройки Поиск решения

Пусть дана задача линейного программирования с двумя переменными.

В общем случае для решения задачи необходимо:

1. Создать формулу для ввода условий задачи.

В ячейки B4 и C4 вносим коэффициенты при неизвестных из целевой функции. Диапазон ячеек B7:C9 заполняем коэффициентами при неизвестных из левых частей неравенств (ограничений), а диапазон F7:F9 – известными значениями из правых частей неравенств (рисунок 21).

2. Задать начальные приближения для неизвестных в ячейках, в которые будет помещен результат решения задачи (изменяемые ячейки).

На рабочем листе Excel (рисунок 21) используем две ячейки B3 и C3, которые до запуска инструмента Поиск решения будут содержать начальные приближения переменных  $x$  и  $y$  (здесь, например, мы задали их равными нулю).

$$Z(x, y) = 2x + y \rightarrow \text{extr (max или min)}$$

$$\begin{cases} 3x + 4y \geq 10; \\ 2x - y \leq 9; \\ x - 3y \geq -4; \\ x \geq 0; y \geq 0. \end{cases}$$

	A	B	C	D	E	F
1		Переменные				
2		x	y			
3	значения	0	0	ЦФ	направление	
4	коэффициенты	2	1	0	max	
5						
6		Ограничения		левая часть	знак	правая часть
7	коэф. огр. 1	3	4	0	>=	10
8	коэф. огр. 2	2	-1	0	<=	9
9	коэф. огр. 3	1	-3	0	>=	-4

Рисунок 21 – Исходные данные для решения ЗЛП

	D
1	
2	
3	ЦФ
4	=СУММПРОИЗВ(\$B\$3:\$C\$3;B4:C4)
5	
6	левая часть
7	=СУММПРОИЗВ(\$B\$3:\$C\$3;B7:C7)
8	=СУММПРОИЗВ(\$B\$3:\$C\$3;B8:C8)
9	=СУММПРОИЗВ(\$B\$3:\$C\$3;B9:C9)

Рисунок 22 – Формулы из рисунка 21

3. Ввести зависимость для целевой функции и ограничений.

В ячейку D4 вносим формулу для определения целевой функции, заменяя в ней неизвестные переменные ссылками на ячейки, в которых содержатся их начальные значения. Такую же операцию выполняем и для ячеек D7, D8 и D9, которые содержат формулы, определяющие значения левых частей неравенств (рисунок 22).

4. Указать параметры в окне Поиск решения и вывести результаты.

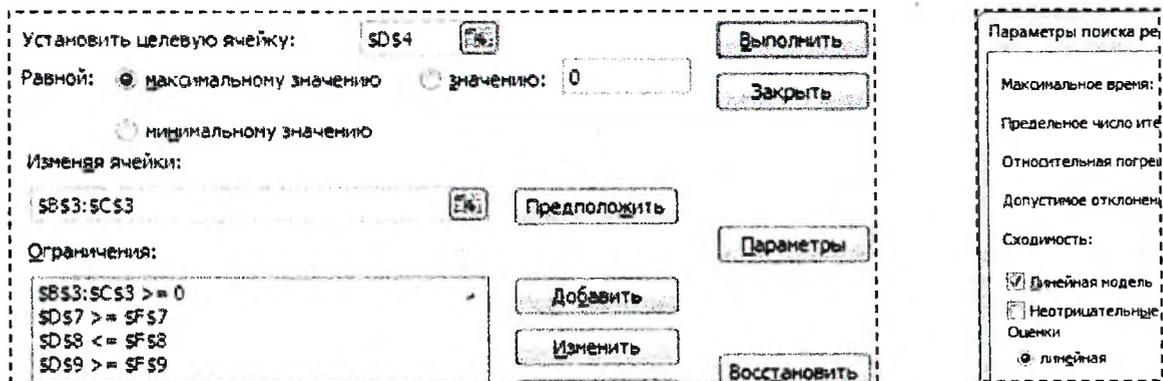


Рисунок 23 – Фрагменты диалоговых окон Поиск решения и Параметры поиска решения

Заполняем соответствующие поля диалогового окна Поиск решения: указываем целевую ячейку, направление поиска, изменяемые ячейки и ограничения (рисунок 23). Реализовать ограничения ( $x \geq 0$  и  $y \geq 0$ ) можно двумя способами: добавить его в диалоговом окне Поиск решения в группу Ограничения либо нажать на кнопку Параметры и в появившемся диалоговом окне Параметры поиска решения выбрать опцию Неотрицательные значения (рисунок 23). Второй способ Excel 2010 можно реализовать путем выбора в диалогом

окне Поиск решения опция Сделать переменные без ограничений неотрицательными. В качестве метода для поиска решения выберем Симплекс-метод (в зависимости от версии Excel: в диалоговом окне Параметры поиска решения опция Линейная модель или в диалоговом окне Поиск решения метод решения Симплекс-метод).

	A	B	C	D
1	Переменные			
2		x	y	
3	значения	6,2	3,4	ЦФ
4	коэффициенты	2	1	15,8
5	Ограничения левая часть			
7	коэф. огр. 1	3	4	32,2
8	коэф. огр. 2	2	-1	9
9	коэф. огр. 3	1	-3	-4

Рисунок 24 – Результат решения ЗЛП

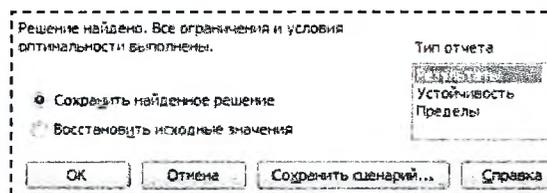


Рисунок 25 – Выбор типа отчета

По завершении процедуры Поиск решения ячейки B3 и C3 будут содержать решение ( $x=6.2$  и  $y=3.4$ ) ЗЛП (рисунок 24). Также по результатам работы процедуры Поиск решения можно сформировать отчет на отдельном листе, указав пункт Результаты в диалоговом окне Результаты поиска решения, которое отобразится после успешного завершения процедуры поиска (рисунок 25).

### Решение транспортной задачи с помощью надстройки Поиск решения

Допустим, составляется план поставок строительного песка на объекты строительной организации на очередную рабочую смену. Поставки планируется осуществлять с двух карьеров A1 и A2, способных производить по  $a1=267 \text{ м}^3$  (D3) и  $a2=133 \text{ м}^3$  (D4) в смену соответственно. Заявки поступили с двух объектов B1 и B2 в количествах  $b1=144 \text{ м}^3$  (B5) и  $b2=256 \text{ м}^3$  (C5) соответственно. Известна стоимость  $c$  ден. ед. (B3:C4) за перевозку одного  $\text{м}^3$  песка с каждого карьера на каждый объект. Например, стоимость перевозки одного  $\text{м}^3$  песка с карьера A1 на объект B2 – 5 ден. ед.

	A	B	C	D
1	Объекты			
2	Карьеры	B1	B2	предложение
3	A1	3	5	267
4	A2	7	4	133
5	спрос	144	256	

Рисунок 26 – Исходные данные для решения транспортной задачи

Требуется составить план перевозок, который полностью удовлетворяет спрос объектов в строительном песке, а суммарные издержки за его перевозку при этом минимизируются.

Порядок решения транспортной задачи совпадает, в общем, с порядком решения, описанным в предыдущем примере с некоторыми особенностями.

1. Все исходные данные представлены на рисунке 26. Т.к. суммарный объем спроса равен суммарному объему предложения, то задача является закрытой, и нет необходимости в дополнительных исходных данных. В случае если объемы спроса и предложения не равны друг другу, то задача является открытой, которую необходимо преобразовать в закрытую. Подробнее о таком преобразовании см. далее теоретический материал.

2. На рабочем листе Excel (рисунок 27) определяем диапазон ячеек B9:C10, которые до запуска инструмента Поиск решения будут содержать начальные приближения для неизвестных, обозначающих количество  $\text{м}^3$  песка, перевозимого с каждого карьера на каждый объект.

	A	B	C	D
7	Объекты			
8	Карьеры	B1	B2	предложение
9	A1	0	0	=СУММ(B9:C9)
10	A2	0	0	=СУММ(B10:C10)
11	спрос	=СУММ(B9:B10)	=СУММ(C9:C10)	
12				ЦФ
13				=СУММПРОИЗВ(B3:C4;B9:C10)

Рисунок 27 – Формулы для решения транспортной задачи

	A	B	C	D
7	Объекты			
8	Карьеры	B1	B2	предложение
9	A1	144	123	267
10	A2	0	133	133
11	спрос	144	256	
12				ЦФ
13				1579

Рисунок 28 – Результат решения задачи

3. В ячейку D13 вносим формулу для определения целевой функции, которая выражает суммарные издержки на транспортировку всего объема песка. В диапазоне ячеек D9:D10 определяем суммарный объем перевозок песка с каждого карьера, а в диапазоне B11:C11 – суммарный объем перевозок песка на каждый объект (рисунок 27).

4. Заполняем соответствующие поля диалогового окна Поиск решения: указываем целевую ячейку, направление поиска (минимальное значение), изменяемые ячейки и ограничения (рисунок Ошибка! Источник ссылки не найден.).

По завершении процедуры Поиск решения диапазон ячеек B9:C10 будет содержать решение задачи (рисунок 28). Из ответа следует, что с карьера A1 перевезут  $144 \text{ м}^3$  песка на объект B1 и  $123 \text{ м}^3$  песка – на объект B2. С карьера A2 доставят песок только на объект B2 в объеме  $133 \text{ м}^3$ .

По результатам работы процедуры Поиск решения можно сформировать отчет на отдельном листе, указав пункт Результаты в диалоговом окне Результаты поиска решения.

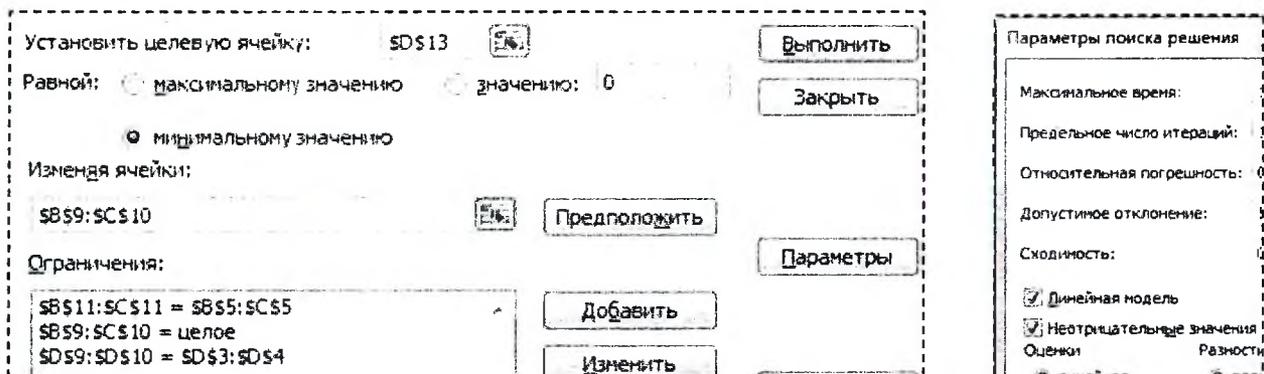


Рисунок 29 – Фрагменты диалоговых окон Поиск решения и Параметры поиска решения

### Решение задачи о раскрое с помощью надстройки Поиск решения

Предположим, что на предприятии имеются стальные бруски длиной  $L=7.9$  м (К2), из которых необходимо получить 38 (К6) заготовок  $p1$  длиной  $d1=1.4$  м (В6), 34 (К7) заготовки  $p2$  длиной  $d2=2.9$  м (В7) и 35 (К8) заготовок  $p3$  длиной  $d3=3.2$  м (В8). Требуется составить план разрезки брусков по критерию минимума отходов, обеспечивающий выполнение запроса по заготовкам.

	A	B	C	D	E	F	G	H	I	J	K
1			$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$		Длина бруска	
2	кол-во брусков	0	0	0	0	0	0	0		L= 7,9	
3		Варианты									
4											
5	Заготовки	1	2	3	4	5	6	левая часть	знак	правая часть	
6	$d1=1,4$	5	3	3	1	1	1	0	>=	38	
7	$d2=2,9$	0	1	0	2	1	0	0	>=	34	
8	$d3=3,2$	0	0	1	0	1	2	0	>=	35	
9											
10	отходы	0,9	0,8	0,5	0,7	0,4	0,1	цф	-264	--> min	

Рисунок 30 – Исходные данные для решения задачи о раскрое

1. Все исходные данные представлены на рисунке 30. Дополнительно сформировано 6 возможных вариантов  $x_1..x_6$  разрезки брусков на заготовки. Например, по четвертому варианту разрезки бруска выйдет одна заготовка  $d1$  и две заготовки  $d2$ . Подробнее о составлении возможных вариантов разрезки смотри далее в теоретическом материале.

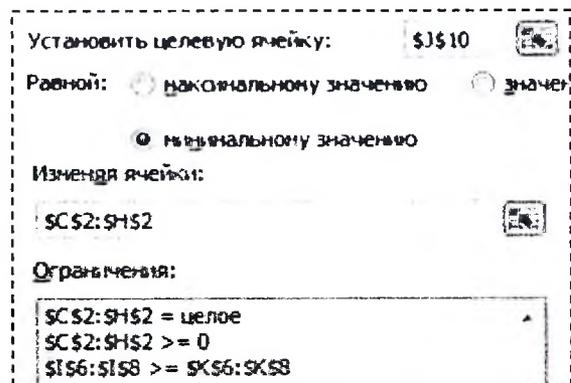


Рисунок 31 – Фрагмент диалогового окна Поиск решения для задачи о раскрое

2. На рабочем листе Excel (рисунок 30) определяем диапазон ячеек C2:H2, которые до запуска инструмента Поиск решения будут содержать начальные приближения для неизвестных, обозначающих количество использованных брусков по каждому варианту разрезки.

3. В диапазоне ячеек C10:H10 определяем отходы от каждого варианта раскроя по формуле, например,  $C10 = \$K\$2 - \text{СУММПРОИЗВ}(C6:C8; \$B\$6: \$B\$8)$

В диапазоне ячеек I6:I8 определяем суммарное количество каждого типа заготовок, полученных от всех вариантов раскроя. Например, формула в ячейке I6 = СУММПРОИЗВ( $\$C\$2: \$H\$2; C6:H6$ ).

Формулу в целевой ячейке формируем таким образом, чтобы ее результат отражал суммарное количество (длину) всех отходов, включающих в себя лишние заготовки.

$$J10 = \text{СУММПРОИЗВ}(C10:H10; \$C\$2: \$H\$2) + \text{СУММПРОИЗВ}(I6:I8 - K6:K8; B6:B8)$$

4. Заполняем соответствующие поля диалогового окна Поиск решения: указываем целевую ячейку, направление поиска, изменяемые ячейки и ограничения (рисунок 31).

По завершении процедуры Поиск решения диапазон ячеек C2:H2 будет содержать решение задачи (рисунок 32). Данный ответ отражает следующее: необходимо раскроить по третьему варианту 2 бруска, по четвертому – 17, по пятому – 1 и по шестому – 16. Суммарная длина отходов равна 20.6 метра, в которые входят 2 лишние заготовки  $d1$  и 1 заготовка  $d2$ .

	A	B	C	D	E	F	G	H	I	J	K
1			$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$		Длина бруска	
2	кол-во брусков	0	0	2	17	1	16			L= 7,9	
3		Варианты									
4											
5	Заготовки	1	2	3	4	5	6	левая часть	знак	правая часть	
6	$d1=1,4$	5	3	3	1	1	1	40	>=	38	
7	$d2=2,9$	0	1	0	2	1	0	35	>=	34	
8	$d3=3,2$	0	0	1	0	1	2	35	>=	35	
9											
10	отходы	0,9	0,8	0,5	0,7	0,4	0,1	цф	20,6	--> min	

Рисунок 32 – Фрагмент диалогового окна Поиск решения

## СИСТЕМА КОМПЬЮТЕРНОЙ МАТЕМАТИКИ MATHCAD

Математическое решение дифференциальных уравнений – очень сложная проблема. MathCAD не в состоянии решить без дополнительных упрощений многие дифференциальные уравнения и их системы.

### Численное решение дифференциального уравнения второго порядка

Одной из функций MathCAD, предназначенной для численного решения дифференциальных уравнений (рисунок 33) и их систем методом Рунге-Кутты четвертого порядка, является функция

$$\text{rkfixed}(u, a, b, N, D),$$

где  $u$  – вектор-столбец (размерности равной порядку дифференциального уравнения) начальных условий, определенных в начале интервала ( $a$ );

$a, b$  – начало и конец интервала, на котором ищется решение дифференциального уравнения;

$N$  – число разбиений интервала  $[a, b]$ ;

$D$  – вектор-столбец (таким же размером как и  $u$ ), содержащий правые части первых производных.

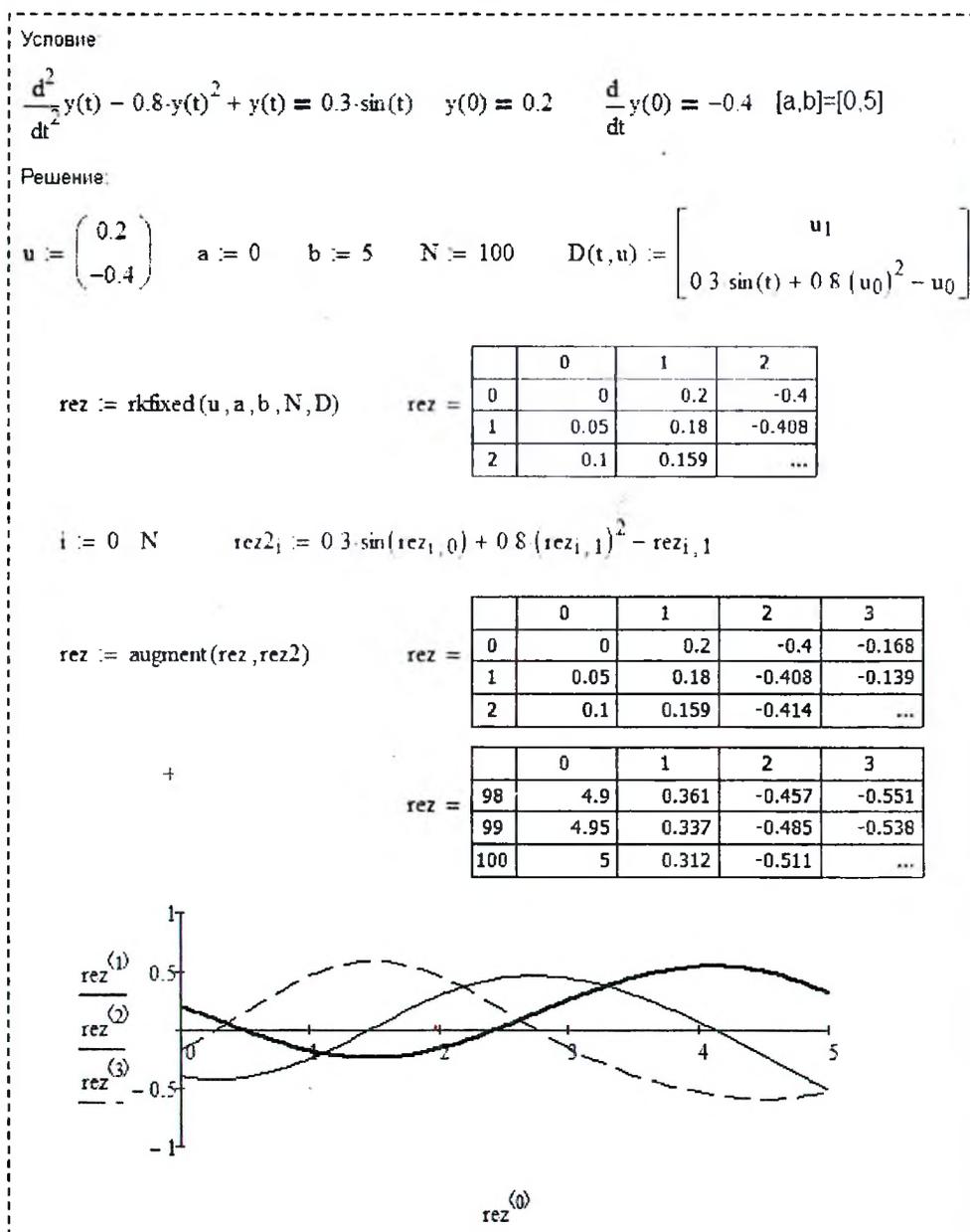


Рисунок 33 – Решение дифференциального уравнения второго порядка из рисунка 14

Перед тем как заполнять параметры функции  $\text{rkfixed}$ , решая обыкновенное дифференциальное уравнение второго порядка, необходимо это уравнение преобразовать к системе двух обыкновенных дифференциальных уравнений первого порядка. Подробнее о такого рода замене смотри далее теоретический материал.

## Решение СЛАО

Наиболее простым способом решения СЛАО в MathCAD является способ, в котором используется блок **Given..Find**.

Условие	Решение (листинг MathCAD)
$\begin{cases} 2 \cdot x + 3 \cdot y = 15 \\ -9 \cdot x + 6 \cdot y = 18 \end{cases}$	<pre> x := 1      y := 1 Given 2x + 3y = 15      -9x + 6y = 18 (x) := find(x,y)  (x) = (0.923) (y)              (y) = (4.385)                     </pre>

Необходимо помнить два важных момента. Первый – перед блоком необходимо задать начальные приближения для неизвестных параметров и второй – между левой и правой частями уравнений должен стоять знак логического равенства (панель **Булевы операторы**).

## Решение задачи линейного программирования

При решении задач линейного программирования для поиска условного экстремума функции нескольких переменных в MathCAD можно использовать блок **Given..Minimize (Given..Maximize)**. Его использование схоже с использованием блока **Given..Find**, только вместо команды **Find** используем команду **Minimize (Maximize)**, в которой указываем имя функции и необходимые ее аргументы.

Условие	Решение (листинг MathCAD)
$Z(x, y) = 2x + y \rightarrow \max$ $\begin{cases} 3x + 4y \geq 10; \\ 2x - y \leq 9; \\ x - 3y \geq -4; \\ x \geq 0; y \geq 0. \end{cases}$	<pre> Z(x, y) := 2x + y      x := 0      y := 0 given 3x + 4y &gt;= 10      2x - y &lt;= 9      x - 3y &gt;= -4      x &gt;= 0      y &gt;= 0 (x) := maximize(Z, x, y) = (6.2) (y)                                Z(x, y) = 15.8                     </pre>

Результат совпадает с решением в Excel, показанным на рисунке 24.

## Графическое решение задачи линейного программирования

Под графическим решением задачи линейного программирования будем понимать подбор неизвестного параметра вручную таким образом, чтобы на графике целевая функция касалась области допустимых ее значений.

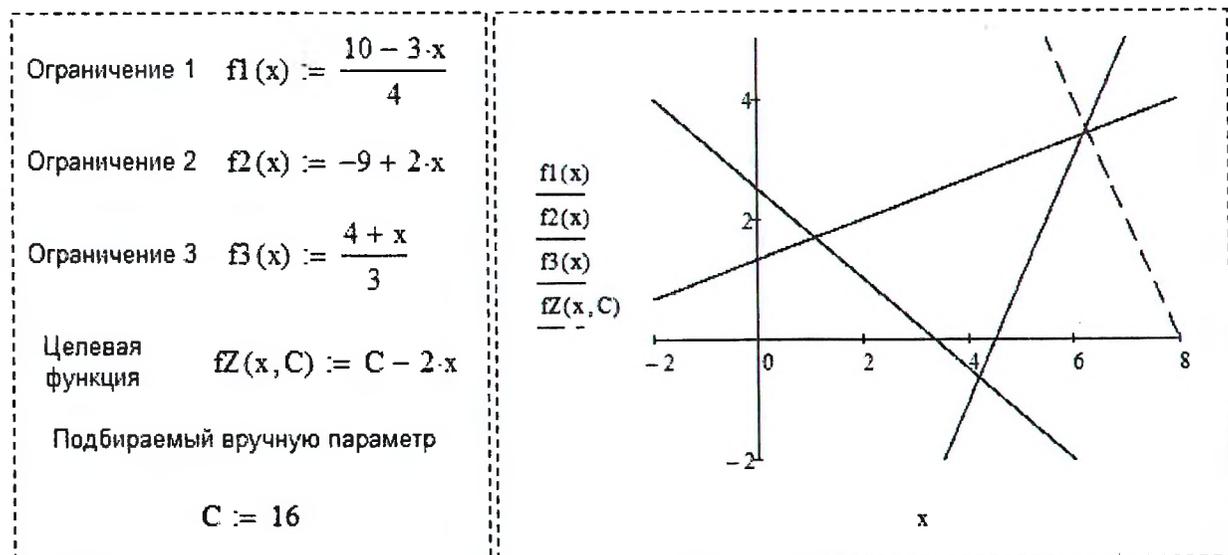


Рисунок 34 – Графическое решение задачи линейного программирования, показанной на рисунке 24.

## ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

### Дифференциальные уравнения

Дифференциальные уравнения – это уравнения, в которых неизвестная функция входит под знаком производной или дифференциала.

Обыкновенным дифференциальным уравнением (ОДУ) называются уравнения, которые содержат одну или несколько производных от искомой функции.

$$F(x, y, y', \dots, y^{(n)}) = 0.$$

Линейным дифференциальным уравнением называется уравнение, линейное относительно искомой функции и её производной.

Решением дифференциального уравнения называется функция  $y = \varphi(x)$ , которая после подстановки в дифференциальное уравнение превращает его в тождество.

Общее решение ДУ  $n$ -го порядка имеет вид

$$y(x) = \varphi(x, c_1, c_2, \dots, c_n).$$

Частное решение ДУ получается из общего, если произвольным постоянным  $c_1, c_2, \dots, c_n$  придать определенные значения.

В зависимости от способа задания дополнительных (значения искомой функции и её производных в некоторых точках) условий различают два типа задач:

- *задача Коши* дополнительные условия заданы в одной точке;
- *краевая задача* дополнительные условия заданы в более чем в одной точке.

Различают следующие методы решения ОДУ:

- аналитические методы;
- приближенные методы;
- графические методы;
- численные методы.

Обыкновенное дифференциальное уравнение или систему таких уравнений можно решить аналитически или численно в СКМ Maple с помощью команды `dsolve`. СКМ MathCAD позволяет получить решение таких уравнений с помощью функции `rkfixed`. Однако для удобства пользования этой функцией необходимо дифференциальное уравнение  $n$  порядка привести к системе из  $n$  уравнений эквивалентной дифференциальному уравнению  $n$  порядка. Например,

<p>Дано:</p> $y''(t) - 0.8 \cdot y(t)^2 + y(t) = 0.3 \cdot \sin(t)$ $y(0) = 0.2, y'(0) = -0.4$ <p>Выражаем <math>y''(t)</math></p> $y''(t) = 0.3 \cdot \sin(t) + 0.8 \cdot y(t)^2 - y(t)$	<p>Выполняем замену</p> $y'(t) = z(t)$	<p>Получаем систему двух ДУ</p> $\begin{cases} y'(t) = z(t) \\ z'(t) = 0.3 \cdot \sin(t) + 0.8 \cdot y(t)^2 - y(t) \end{cases}$ $y(0) = 0.2, z(0) = -0.4$
---	--	---

Результат передаем в параметры функции `rkfixed` следующим образом. Вектор-столбец  $u$  заполняем правыми частями дополнительных условий, а вектор-столбец  $D$  – правыми частями дифференциальных уравнений из системы (пример на рисунке 33).

### Графический метод решения задачи линейного программирования (ЗЛП)

Дана задача линейного программирования с двумя переменными:

Необходимо определить экстремальное значение функции. Для достижения этой цели выполним следующие действия.

1. Построим область допустимых значений (ОДЗ).

Каждое из пяти ограничений графически представляет собой прямую (рисунок 35), которая делит все координатное пространство на две части: одна часть содержит ОДЗ функции, вторая – нет. Координатное пространство, находящееся на пересечении пяти таких областей, будет являться ОДЗ функции. ОДЗ может быть в виде: неограниченной области, ограниченной области, отрезка или точки.

В нашем случае областью допустимых значений является четырехугольник ABCD.

$$Z(x, y) = 2x + y \rightarrow \text{extr (max или min)}$$

$$\begin{cases} 3x + 4y \geq 10; \\ 2x - y \leq 9; \\ x - 3y \geq -4; \\ x \geq 0; y \geq 0. \end{cases}$$

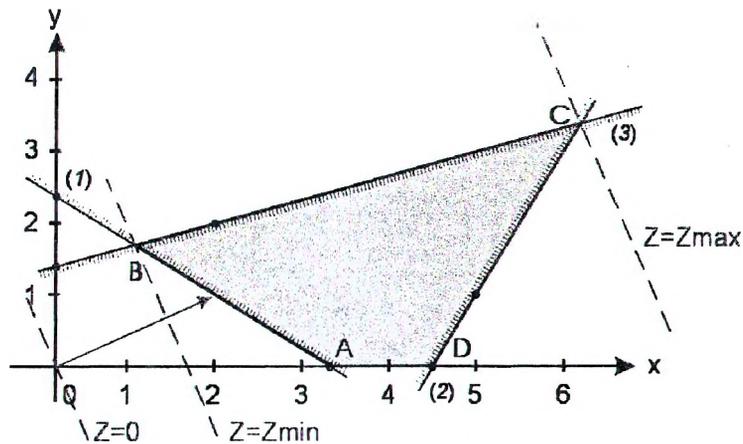


Рисунок 35 – Графическое решение задачи линейного программирования, показанной на рисунке 24

2. Построим вектор наискорейшего возрастания целевой функции (ЦФ).

Вектор указывает направление наискорейшего возрастания функции, а противоположный ему – направление наискорейшего убывания. Величина вектора равна производной функции в этом направлении:

$$\text{grad}F = \begin{pmatrix} \frac{\partial F}{\partial x} \\ \frac{\partial F}{\partial y} \end{pmatrix}, \text{ в нашем случае } \text{grad}F = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

3. Проводим произвольную линию уровня, проходящую через ОДЗ.

Прямая, перпендикулярная градиенту, является линией уровня целевой функции и поэтому во всех своих точках принимает одно и то же значение. Приравнивая целевую функцию к постоянной  $C$ , а затем меняя ее, получим семейство прямых, каждая из которых является линией уровня, которые обладают свойством: при смещении в одну сторону уровень только возрастает, а в другую – только убывает.

4. Перемещаем линию уровня.

При решении задачи на максимум линию уровня перемещаем в направлении градиента. При решении задачи на минимум линию уровня перемещаем в антиградиентном направлении. Линию уровня в обоих случаях перемещают до тех пор, пока она не коснется ОДЗ в крайнем (последнем) положении. Возможны следующие случаи. (1) Крайнее положение линии уровня представляет собой вершину многоугольника, координаты которой и есть *единственное решение ЗЛП*. Значение  $C$ , при котором получено крайнее положение линии уровня, является экстремальным значением ЦФ ЗЛП. (2) ЗЛП имеет *бесконечное множество решений*, если линия уровня в своем крайнем положении лежит на стороне многоугольника. Тогда координаты любой точки, лежащей на стороне многоугольника, и есть решение ЗЛП. (3) Если при перемещении линии уровня невозможно получить крайнее положение, то *ЗЛП не имеет решения*, т.к. ЦФ неограниченно возрастает (или убывает). В нашем случае максимум функции достигается в точке  $C$ , а минимум – в точке  $B$ .

5. Определяем экстремальное значение функции.

Определить экстремальное значение функции можно в MathCAD с использованием блока **Given..Minimize (Given..Maximize)** или в Excel с помощью надстройки **Поиск решения**.

### Задача о раскрое материалов

Сущность задачи об оптимальном раскрое состоит в разработке таких технологически допустимых планов раскроя, при которых получается необходимый комплект заготовок, а отходы по площади, весу или стоимости сводятся к минимуму.

Предположим, что на предприятии имеются стальные бруски длиной  $L=7.9$  м, из которых необходимо получить 38 заготовок  $p_1$  длиной  $d_1=1.4$  м, 34 заготовки  $p_2$  длиной  $d_2=2.9$  м и 35 заготовок  $p_3$  длиной  $d_3=3.2$  м. Требуется составить возможные варианты разреза бруска и определить величину отходов для каждого из вариантов, а также привести возможные математическую модели для определения количества брусков по каждому варианту разреза, чтобы удовлетворить спрос заготовок при минимальных отходах.

1. Определяем рациональных способов раскроя материала.

В задачах оптимального раскроя рассматриваются так называемые рациональные (оптимальные по Парето) способы раскроя. Предполагается, что из единицы материала можно изготовить заготовки нескольких видов. Способ раскроя единицы материала называется *рациональным (оптимальным по Парето)*, если увеличение числа заготовок одного вида возможно только за счет сокращения числа заготовок другого вида.

Пусть  $i$  – индекс вида заготовки,  $i = 1..m$ ;  $v$  – индекс способа раскроя единицы материала,  $v = 1..s$ ;  $a_{iv}$  – количество (целое число) заготовок вида  $i$ , полученных при раскрое единицы материала  $v$ -м способом.

Приведенное определение рационального способа раскроя может быть формализовано следующим образом. Способ раскроя  $k$  называется *рациональным (оптимальным по Парето)*, если для любого другого способа раскроя  $v$  из соотношений  $a_{iv} > a_{ik}$ ,  $i = 1..m$ , следуют соотношения  $a_{iv} = a_{ik}$ ,  $i = 1..m$ .

вариант	1	2	3	4	5	6	7	8	9	10	11	12
d1= 1,4	5	4	3	3	2	2	1	1	1	0	0	0
d2= 2,9	0	0	1	0	1	0	2	1	0	2	1	0
d3= 3,2	0	0	0	1	0	1	0	1	2	0	1	2
отходы	0,9	2,3	0,8	0,5	2,2	1,9	0,7	0,4	0,1	2,1	1,8	1,5
		(1)		(3)	(4)				(7)	(8)	(9)	

Тогда количество отходов при  $v$  способе раскроя материала:

$$c_k = L - \sum_{i=1}^m (a_{ik} \cdot d_i),$$

где  $k = 1..n$ , где  $n$  – общее число рациональных способов раскроя.

2. Определение интенсивности использования рациональных способов раскроя.

Для определения интенсивности составим математическое описание исследуемого процесса.

а) переменные:  $x_k$  – количество брусков ( $x_k \geq 0$ ,  $x_k$  – целые), полученных по варианту  $k$ .

б) целевая функция:  $c_1 x_1 + c_2 x_2 + \dots + c_n x_n$  – общая сумма остатков, которая по условию задачи должна быть минимальна

$$\sum_{k=1}^n (c_k \cdot x_k) \rightarrow \min,$$

где  $c_k \cdot x_k$  – сумма остатков от разрезки брусков по варианту  $k$ .

в) ограничения:  $a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n$  – общее количество заготовок длины  $d_i$ , после разреза всех брусков должно быть не менее заданного по условию задачи количества  $p_i$

$$\sum_{k=1}^n (a_{i,k} \cdot x_k) \geq p_i,$$

где  $a_{i,k} x_k$  – количество заготовок длины  $d_i$  от разрезки брусков по варианту  $k$ .

Если количество полученных заготовок превысит требуемое, то лишние заготовки будут считаться отходами. Тогда целевая функция примет следующий вид

$$\sum_{k=1}^n (c_k \cdot x_k) + \sum_{i=1}^m \left( \left( \sum_{k=1}^n (a_{i,k} \cdot x_k) - p_i \right) \cdot d_i \right) \rightarrow \min$$

В итоге, математическая модель (в скалярной форме) для выбранного варианта будет следующей

$$0.9 \cdot x_1 + 0.8 \cdot x_2 + 0.5 \cdot x_3 + 0.7 \cdot x_4 + 0.4 \cdot x_5 + 0.1 \cdot x_6 + (5 \cdot x_1 + 3 \cdot x_2 + 3 \cdot x_3 + x_4 + x_5 + x_6 - 38) \cdot 1.4 + (x_2 + 2 \cdot x_4 + 5 \cdot x_5 - 34) \cdot 2.9 + (x_3 + x_5 + 2 \cdot x_6 - 35) \cdot 3.2 \rightarrow \min$$

$$\begin{cases} 5 \cdot x_1 + 3 \cdot x_2 + 3 \cdot x_3 + x_4 + x_5 + x_6 \geq 38; \\ x_2 + 2 \cdot x_4 + 5 \cdot x_5 \geq 34; \\ x_3 + x_5 + 2 \cdot x_6 \geq 35; \\ x_1 \geq 0; x_2 \geq 0; x_3 \geq 0; x_4 \geq 0; x_5 \geq 0; x_6 \geq 0; \\ x_1, x_2, x_3, x_4, x_5, x_6 - \text{целые.} \end{cases}$$

3. Определяем экстремальное значение функции.

Определить экстремальное значение функции можно в Excel с помощью надстройки Поиск решения.

### Транспортная задача

Допустим, составляется план поставок строительного песка на объекты строительной организации на очередную рабочую смену. Поставки планируется осуществлять с  $m=2$  карьеров  $A1$  и  $A2$ , способных производить по  $a_1=267 \text{ м}^3$  и  $a_2=133 \text{ м}^3$  в смену соответственно. Заявки поступили с  $n=2$  объектов  $B1$  и  $B2$  в количествах  $b_1=144 \text{ м}^3$  и  $b_2=256 \text{ м}^3$  соответственно. Известна стоимость  $c$  ден. ед. за перевозку одного  $\text{м}^3$  песка с каждого карьера на каждый объект ( $c_{1,1}=3$ ,  $c_{1,2}=5$ ,  $c_{2,1}=7$ ,  $c_{2,2}=4$ ).

Требуется составить план перевозок, который полностью удовлетворяет спрос объектов в строительном песке, а суммарные издержки за его перевозку при этом минимизируются.

В суммарные издержки как правило входят издержки от перевозки песка с карьеров на объекты, а также издержки, связанные с затратами на выработку, на недопоставку или избыток объемов песка.

Различают две модели транспортной задачи: **открытая** и **закрытая**. Модель транспортной задачи называют **закрытой**, если суммарный объем груза, имеющегося у поставщиков, равен суммарному спросу потребителей, т.е. выполняется равенство

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j.$$

Построим математическую модель задачи.

а) переменные:  $x_{ij}$  ( $i = 1..m, j = 1..n$ ) – объем песка, перевозимого с карьера  $A_i$  на объект  $B_j$ . Объем перевозимого песка есть натуральное число.

б) целевая функция: суммарные затраты на транспортировку продукции должны быть минимальны.

$$Z = Z_T = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min$$

где  $(c_{ij} \cdot x_{ij})$  – стоимость перевозки песка с карьера  $A_i$  на объект  $B_j$ .

в) ограничения: объем перевезенного песка с каждого карьера равен объему его производства и одновременно объем перевезенного песка на каждый объект равен объему его потребления, т.е.

$$\sum_{j=1}^n x_{ij} = a_i, \quad \sum_{i=1}^m x_{ij} = b_j.$$

В нашем случае математическая модель в скалярной форме примет вид:

$$Z = Z_T = 3 \cdot x_{1,1} + 5 \cdot x_{1,2} + 7 \cdot x_{2,1} + 4 \cdot x_{2,2} \rightarrow \min$$

$$\begin{cases} x_{1,1} + x_{1,2} = 267; & x_{1,1} + x_{2,1} = 144; & x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2} \geq 0; \\ x_{2,1} + x_{2,2} = 133; & x_{1,2} + x_{2,2} = 256; & x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2} - \text{целые.} \end{cases}$$

Модель транспортной задачи называют **открытой**, если выполняется одно из условий:

$$\sum_{i=1}^m a_i < \sum_{j=1}^n b_j \quad \text{или} \quad \sum_{i=1}^m a_i > \sum_{j=1}^n b_j.$$

Для разрешимости транспортной задачи с открытой моделью необходимо преобразовывать ее в закрытую, путем ввода фиктивного поставщика  $A_{m+1}$ , способного производить

$$a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i; \quad c_{m+1,j} = 0, \quad j = \overline{1, n};$$

либо фиктивного потребителя  $B_{n+1}$ , способного принять

$$b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j; \quad c_{i,n+1} = 0, \quad i = \overline{1, m}.$$

Рассмотрим два варианта открытой задачи путем умышленного изменения объема производства на карьере А1. В первом случае объем производства будет равен  $207 \text{ м}^3$  (добавляем фиктивный карьер), а во втором –  $307 \text{ м}^3$  (добавляем фиктивный объект).

	A	B	C	D
1	Объекты			
2	Карьеры	B1	B2	предложение
3	A1	3	5	207
4	A2	7	4	133
5	ф А3	0	0	60
6	спрос	144	256	

Рисунок 36 – Транспортная задача с фиктивным поставщиком

	A	B	C	D	E
1	Объекты				
2	Карьеры	B1	B2	ф B3	предложение
3	A1	3	5	0	307
4	A2	7	4	0	133
5	спрос	144	256	40	

Рисунок 37 – Транспортная задача с фиктивным потребителем

На рисунке 36 для трансформации открытой транспортной задачи в закрытую добавлен фиктивный карьер А3 с объемом производства  $60 \text{ м}^3$ . На рисунке 37 добавлен фиктивный объект В3 с объемом потребления  $40 \text{ м}^3$ . В обоих вариантах стоимость транспортировки единицы объема песка от фиктивного поставщика или к фиктивному потребителю равна нулю.

В связи с добавлением фиктивного (несуществующего) поставщика (карьера) изменится математическая модель задачи путем добавления фиктивных (недопоставленных) объемов  $x_{3,1}$  и  $x_{3,2}$ . Таким образом:

$$Z = Z_T = 3 \cdot x_{1,1} + 5 \cdot x_{1,2} + 7 \cdot x_{2,1} + 4 \cdot x_{2,2} + 0 \cdot x_{3,1} + 0 \cdot x_{3,2} \rightarrow \min$$

$$\begin{cases} x_{1,1} + x_{1,2} = 207; & x_{3,1} + x_{3,2} = 60; & x_{1,1} + x_{2,1} + x_{3,1} = 144; & x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2}, x_{3,1}, x_{3,2} \geq 0; \\ x_{2,1} + x_{2,2} = 133; & & x_{1,2} + x_{2,2} + x_{3,2} = 256; & x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2}, x_{3,1}, x_{3,2} - \text{целые.} \end{cases}$$

С добавлением фиктивного (несуществующего) потребителя (объекта) математическая модель задачи изменится путем добавления фиктивных (лишних) объемов  $x_{1,3}$  и  $x_{2,3}$ .

$$Z = Z_T = 3 \cdot x_{1,1} + 5 \cdot x_{1,2} + 7 \cdot x_{2,1} + 4 \cdot x_{2,2} + 0 \cdot x_{1,3} + 0 \cdot x_{2,3} \rightarrow \min$$

$$\begin{cases} x_{1,1} + x_{1,2} + x_{1,3} = 307; & x_{1,1} + x_{2,1} = 144; & x_{1,3} + x_{2,3} = 40; & x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2}, x_{1,3}, x_{2,3} \geq 0; \\ x_{2,1} + x_{2,2} + x_{2,3} = 133; & x_{1,2} + x_{2,2} = 256; & & x_{1,1}, x_{1,2}, x_{2,1}, x_{2,2}, x_{1,3}, x_{2,3} - \text{целые.} \end{cases}$$

В формулировке транспортной задачи могут присутствовать дополнительные условия, которые, несомненно, приведут к изменению математической модели. Охватить весь набор таких условий не представляется возможным. Тем не менее рассмотрим некоторые из них.

Например, выработка песка (2 ден. ед. за  $m^3$ ) связана с затратами на его производство  $Z_p$ . Такое условие приведет к изменению целевой функции  $Z$ .

$$Z = Z_T + Z_p \rightarrow \min, \text{ где } Z_p = 2 \cdot (x_{1,1} + x_{1,2} + x_{2,1} + x_{2,2}).$$

Для открытой модели целевая функция идентична с тем условием, что (в случае с фиктивным потребителем) карьеры лишний песок не выработывают (простаивают), иначе

$$Z_p = 2 \cdot (x_{1,1} + x_{1,2} + x_{2,1} + x_{2,2} + x_{1,3} + x_{2,3}).$$

Другой пример. Суточный (смена) простой объекта влечет за собой штраф в размере 400 ден. ед. Такое условие присутствует в моделях с фиктивным поставщиком, в которых недопоставка песка может повлечь за собой простой объекта. В этом случае необходимо предварительно определить штраф от простоя объекта из расчета недопоставки единицы объема песка. Затем изменить целевую функцию

$$Z = Z_T + Z_p + Z_{pR} \rightarrow \min, \text{ где } Z_{pR} = \frac{400}{144} \cdot x_{3,1} + \frac{400}{256} \cdot x_{3,2} \approx 2.778 \cdot x_{3,1} + 1.563 \cdot x_{3,2}.$$

И наоборот. Лишняя выработанная продукция влечет за собой затраты на ее хранение (3 ден. ед. за  $m^3$ ). Такое условие присутствует в моделях с фиктивным потребителем. Так как затраты за хранение единицы объема песка известны, то предварительно ничего не нужно определять. Необходимо изменить целевую функцию следующим образом:

$$Z = Z_T + Z_p + Z_{II} \rightarrow \min, \text{ где } Z_{II} = 3 \cdot (x_{1,3} + x_{2,3}).$$

Если выражение целевой функции привести к подобным слагаемым, то получим следующее:

– для закрытой модели

$$Z = (3 + 2) \cdot x_{1,1} + (5 + 2) \cdot x_{1,2} + (7 + 2) \cdot x_{2,1} + (4 + 2) \cdot x_{2,2} \rightarrow \min;$$

– для модели с фиктивным поставщиком

$$Z = (3 + 2) \cdot x_{1,1} + (5 + 2) \cdot x_{1,2} + (7 + 2) \cdot x_{2,1} + (4 + 2) \cdot x_{2,2} + 2.778 \cdot x_{3,1} + 1.563 \cdot x_{3,2} \rightarrow \min;$$

– для модели с фиктивным потребителем

$$Z = (3 + 2) \cdot x_{1,1} + (5 + 2) \cdot x_{1,2} + (7 + 2) \cdot x_{2,1} + (4 + 2) \cdot x_{2,2} + (2 + 3) \cdot x_{1,3} + (2 + 3) \cdot x_{2,3} \rightarrow \min.$$

С учетом вышеизложенных дополнительных условий коэффициенты целевой функции в открытых моделях в Excel можно представить, как показано на рисунках 38 и 39.

	A	B	C	D
1	Объекты			
2	Карьеры	B1	B2	предложение
3	A1	5	7	207
4	A2	9	6	133
5	ф А3	2,778	1,563	60
6	спрос	144	256	

Рисунок 38 – Транспортная задача с фиктивным поставщиком

	A	B	C	D	E
1	Объекты				
2	Карьеры	B1	B2	ф B3	предложение
3	A1	5	7	5	307
4	A2	9	6	5	133
5	спрос	144	256	40	

Рисунок 39 – Транспортная задача с фиктивным потребителем

## Использование Maple в задачах по теоретической механике

### Решение типовой задачи по теоретической механике

В железнодорожных скальных выемках для защиты кюветов от попадания в них с откосов каменных осыпей устраивается «полка»  $DC$ . Учитывая возможность движения камня из наивысшей точки  $A$  откоса и полагая при этом его начальную скорость  $v_0=0$ , определить наименьшую ширину полки  $b$  и скорость  $v_C$ , с которой камень падает на нее. По участку  $AB$  откоса, составляющему угол  $\alpha$  с горизонтом и имеющему длину  $l$ , камень движется  $\tau$  с.

При решении задачи считать коэффициент трения скольжения  $f$  камня на участке  $AB$  постоянным, а сопротивление воздуха пренебречь.

Дано:  $v_A = 0$ ;  $\alpha = 60^\circ$ ;  $l = 4$  м;  $\tau = 1$  с;  $f \neq 0$ ;  $h = 5$  м;  $\beta = 75^\circ$ .

Определить  $b$  и  $v_C$ .

Решение. Рассмотрим движение камня на участке  $AB$ . Принимая камень за материальную точку, покажем на рисунке действующие на него силы: вес  $\vec{G}$ , нормальную реакцию  $\vec{N}$  и силу трения скольжения  $\vec{F}$ . Составим дифференциальное уравнение движения камня на участке  $AB$ :

$$m\ddot{x}_1 = \sum X_{i1}; \quad m\ddot{x}_1 = G \sin \alpha - F.$$

Сила трения  $F = fN$ , где  $N = G \cos \alpha$ .

Таким образом,  $m\ddot{x}_1 = G \sin \alpha - fG \cos \alpha$  или

$$\ddot{x}_1 = g \sin \alpha - fg \cos \alpha.$$

Интегрируя дифференциальное уравнение дважды, получаем

$$\dot{x}_1 = g(\sin \alpha - f \cos \alpha)t + C_1$$

$$x_1 = [g(\sin \alpha - f \cos \alpha)/2]t^2 + C_1t + C_2.$$

Для определения постоянных интегрирования воспользуемся начальными условиями задачи: при  $t=0$   $x_{10} = 0$  и  $\dot{x}_{10} = 0$ .

Составим уравнения, полученные при интегрировании, для  $t=0$ :  $\dot{x}_{10} = C_1$ ;  $x_{10} = C_2$ .

Найдем постоянные:  $C_1 = 0$ ,  $C_2 = 0$ . Тогда

$$\dot{x}_1 = g(\sin \alpha - f \cos \alpha)t; \quad x_1 = [g(\sin \alpha - f \cos \alpha)/2]t^2.$$

Для момента  $\tau$ , когда камень покидает участок,  $\dot{x}_1 = v_B$ ;  $x_1 = l$ , т.е.

$$v_B = g(\sin \alpha - f \cos \alpha)\tau; \quad l = [g(\sin \alpha - f \cos \alpha)/2]\tau^2$$

откуда  $v_B = 2l/\tau$ , т.е.  $v_B = 2 \cdot 4/1 = 8$  м/с.

Рассмотрим движение камня от точки  $B$  до точки  $C$ .

Показав силу тяжести  $\vec{G}$ , действующую на камень, составим дифференциальные уравнения его движения:  $m\ddot{x} = 0$ ;  $m\ddot{y} = G$ .

Начальные условия задачи: при  $t=0$

$$x_0 = 0, \quad y_0 = 0; \quad \dot{x}_0 = v_B \cos \alpha, \quad \dot{y}_0 = v_B \sin \alpha.$$

Интегрируем дифференциальные уравнения дважды:

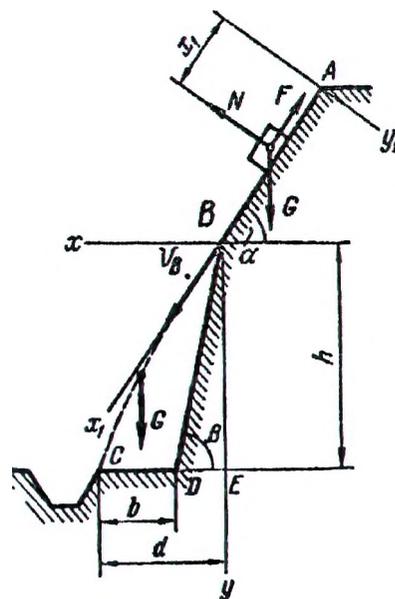
$$\dot{x} = C_3, \quad x = C_3t + C_5; \quad \dot{y} = gt + C_4, \quad y = gt^2/2 + C_4t + C_6.$$

Напишем полученные уравнения для  $t=0$ :

$$\dot{x}_0 = C_3, \quad x_0 = C_5; \quad \dot{y}_0 = C_4, \quad y_0 = C_6.$$

Отсюда найдем, что

$$C_3 = v_B \cos \alpha, \quad C_4 = v_B \sin \alpha, \quad C_5 = 0, \quad C_6 = 0.$$



Получим следующие уравнения проекций скоростей камня:

$$\dot{x} = v_B \cos \alpha, \quad \dot{y} = gt + v_B \sin \alpha$$

и уравнения его движения:  $x = v_B \cos \alpha \cdot t$ ,  $y = gt^2 / 2 + v_B \sin \alpha \cdot t$ .

Уравнение траектории камня найдем, исключив параметр  $t$  из уравнений движения. Определив  $t$  из первого уравнения и подставив его значение во второе, получаем уравнение параболы:

$$y = gx^2 / (2v_B^2 \cos^2 \alpha) + x \cdot tg \alpha.$$

В момент падения  $y=h$ ,  $x=d$ . Определяя  $d$  из уравнения траектории, найдем  $d_1 = 2.11$  м,  $d_2 = -7.75$  м.

Так как траекторией движения камня является ветвь параболы с положительными абсциссами ее точек, то  $d = 2.11$  м. Минимальная ширина полки

$$b = d - ED = d - h / tg 75^\circ, \text{ или } b = 0.77 \text{ м.}$$

Используя уравнение движения камня  $x = v_B \cos \alpha \cdot t$ , найдем время  $T$  движения камня от точки  $B$  до точки  $C$ :  $T = 0.53$  с.

Скорость камня при падении найдем через проекции скорости на оси координат по формуле

$$v = \sqrt{\dot{x}^2 + \dot{y}^2} = \sqrt{(v_B \cos \alpha)^2 + (gt + v_B \sin \alpha)^2}$$

Для момента падения  $t = T = 0.53$  с скорость в точке  $C$   $v_C = 12.8$  м/с.

### Решение типовой задачи по теоретической механике в Maple

Приведем решение вышеизложенной задачи в Maple.

```

> restart
> # Движение камня на участке AB
  G := m·g : N := G·cos(α) : F := f·N :
> DU := m·x1''(t) = G·sin(α) - F :
  NU := x1(0) = 0, x1'(0) = 0 :
> dsolve(DU, x1(t))
# Общее решение ОДУ
      x1(t) = 1/2 ( -cos(α)fg + sin(α)g ) t^2 + C1t + C2
(1)
> Sol := dsolve( {DU, NU}, x1(t) ) :
> odetest(Sol, [DU, NU]) # Проверка решения ОДУ
      [0, 0, 0]
(2)
> x1 := unapply(rhs(Sol), t) :
> 'x1(t)' = x1(t); 'v1(t)' = diff(x1(t), t);
# Частное решение ОДУ и его производная
      x1(t) = 1/2 ( -cos(α)fg + sin(α)g ) t^2
      v1(t) = ( -cos(α)fg + sin(α)g ) t
(3)
> # Движение камня на участке BC
  DU := m·x''(t) = 0, m·y''(t) = G :
> NU := x(0) = 0, x'(0) = vb·cos(α), y(0) = 0, y'(0) = vb·sin(α) :
> dsolve( {DU}, {x(t), y(t)} )
# Общее решение двух ОДУ
      { x(t) = C3t + C4, y(t) = 1/2 g t^2 + C1t + C2 }
(4)

```

```
> Sol := dsolve( {DU, NU}, {x(t), y(t)} )
Sol := { x(t) = t vb cos(α), y(t) = 1/2 g t^2 + vb sin(α) t }
```

(5)

```
> odetest(Sol, [DU, NU]) # Проверка решения двух ОДУ
[0, 0, 0, 0, 0, 0]
```

(6)

```
> x := unapply(rhs(Sol[1]), t); y := unapply(rhs(Sol[2]), t);
vx := unapply(x'(t), t); vy := unapply(y'(t), t);
x := t -> t vb cos(α)
y := t -> 1/2 g t^2 + vb sin(α) t
vx := t -> vb cos(α)
vy := t -> g t + vb sin(α)
```

(7)

```
> 'x(t)' = x(t); 'y(t)' = y(t); 'vx(t)' = vx(t); 'vy(t)' = vy(t);
# Решение двух ДУ и их производные
x(t) = t vb cos(α)
y(t) = 1/2 g t^2 + vb sin(α) t
vx(t) = vb cos(α)
vy(t) = g t + vb sin(α)
```

(8)

```
> # Исходные данные
l := 4 : tb := 1 : g := 9.81 : α := π/3 : h := 5 :
```

```
> # Решение системы уравнений
SLAU := solve([ vb = x l'(tb), l = x l(tb) ], {vb, f})
SLAU := { f = 0.1010620206, vb = 8. }
```

(9)

```
> vb := rhs(SLAU[2])
vb := 8.
```

(10)

```
> # Решение системы уравнений
SLAU := fsolve( {d = x(t), h = y(t)}, {d, t}, t = 0 .. 10)
SLAU := { d = 2.103559659, t = 0.5258899148 }
```

(11)

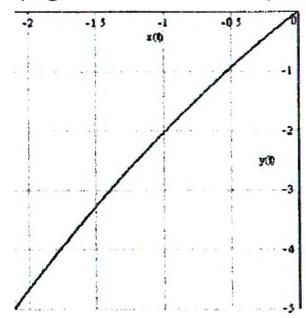
```
> d := rhs(SLAU[1]); tc := rhs(SLAU[2])
d := 2.103559659
tc := 0.5258899148
```

(12)

```
> v := unapply( sqrt(vx(t)^2 + vy(t)^2), t );
> 'v(tc)' = evalf(v(tc))
# Скорость точки в момент падения
v(tc) = 12.73184983
```

(13)

```
> plot( { -x(t), -y(t), t = 0 .. tc }, gridlines, labels = ["x(t)", "y(t)"] )
```



## Лабораторная работа № 1(1)

**ТЕМА.** СКМ Maple: решение дифференциальных уравнений.

**Условие:** Задача Коши для ОДУ первого порядка: \_\_\_\_\_  
 \_\_\_\_\_ начальное условие: \_\_\_\_\_

Задача Коши для системы  
 двух ОДУ первого порядка  
 (с начальными условиями)  $\left\{ \right.$

**Задание: 1.** Сформировать команду в Maple для решения задачи Коши.

[> \_\_\_\_\_

2. Представить аналитическое решение и результат проверки задачи Коши.

$x(t) =$  \_\_\_\_\_ проверка: \_\_\_\_\_

3. Вычислить значения функций  $x(t)$  и  $x'(t)$  в точках  $t=0$  и  $t=T1=$ \_\_\_\_\_.

	$t$	$x(t)$	$x'(t)$
Начальная точка	0		
Конечная точка			

4. Представить аналитическое решение и результат проверки задачи Коши для системы двух ОДУ первого порядка.

$u(t) =$  \_\_\_\_\_

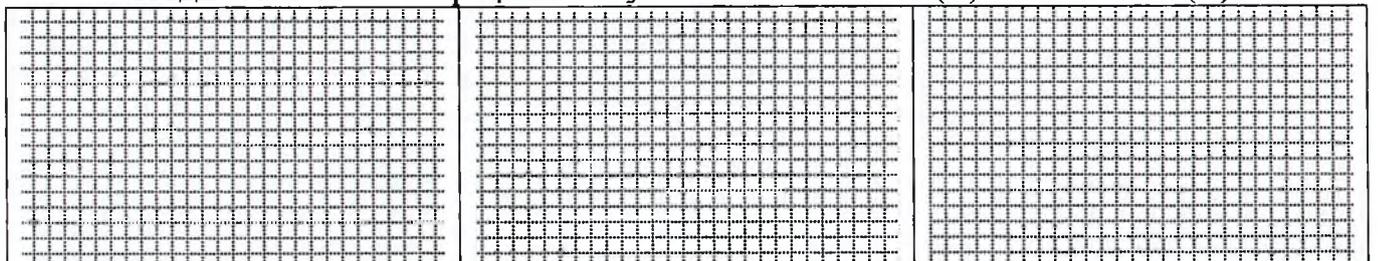
$v(t) =$  \_\_\_\_\_

проверка 1-го ур-ния: \_\_\_\_\_ ; проверка 2-го ур-ния: \_\_\_\_\_.

5. Вычислить значения функций  $u(t)$  и  $v(t)$  в точках  $t=0$  и  $t=T2=$ \_\_\_\_\_

	$t$	$u(t)$	$v(t)$
Начальная точка	0		
Конечная точка			

6. Подписать эскизы графиков и указать начальные (○) и конечные (□) точки.





## Лабораторная работа № 2

**ТЕМА.** СКМ Maple: решение дифференциальных уравнений.

**Условие:** Материальная точка с массой  $m = \underline{\hspace{2cm}}$  кг движется в выбранной системе отсчета под воздействием силы  $F = \underline{\hspace{4cm}}$  Н и силы тяжести  $G = 9.8 \cdot \underline{\hspace{2cm}} = \underline{\hspace{2cm}}$  Н, направление которой совпадает с положительным направлением оси  $Oz$ . Сформировать уравнения движения материальной точки относительно осей координат и построить траекторию движения материальной точки в течение  $T = \underline{\hspace{2cm}}$  с в пространстве.

Начальное положение точки  $x(0) = \underline{\hspace{2cm}}$  м,  $y(0) = \underline{\hspace{2cm}}$  м,  $z(0) = \underline{\hspace{2cm}}$  м.

Начальная скорость точки  $v_x(0) = \underline{\hspace{2cm}}$  м/с,  $v_y(0) = \underline{\hspace{2cm}}$  м/с,  $v_z(0) = \underline{\hspace{2cm}}$  м/с.

**Задание: 1.** Получить уравнение движения материальной точки в направлении оси  $Ox$ .

Дифференциальное уравнение  $\underline{\hspace{6cm}}$  Начальные условия  $\underline{\hspace{2cm}}$   
 $\underline{\hspace{2cm}}$

Уравнение движения  $\underline{\hspace{6cm}}$

**Задание: 2.** Получить уравнение движения материальной точки в направлении оси  $Oy$ .

Дифференциальное уравнение  $\underline{\hspace{6cm}}$  Начальные условия  $\underline{\hspace{2cm}}$   
 $\underline{\hspace{2cm}}$

Уравнение движения  $\underline{\hspace{6cm}}$

**Задание: 3.** Получить уравнение движения материальной точки в направлении оси  $Oz$ .

Дифференциальное уравнение  $\underline{\hspace{6cm}}$  Начальные условия  $\underline{\hspace{2cm}}$   
 $\underline{\hspace{2cm}}$

Уравнение движения  $\underline{\hspace{6cm}}$

**Задание: 4.** Сформировать команду в Maple для построения графика пространственной кривой.

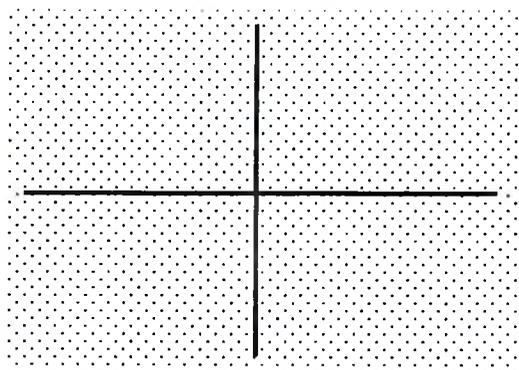
[<  $\underline{\hspace{6cm}}$

**Задание: 5.** Определить положение точки и ее скорость в момент времени  $T$ .

$x(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м       $y(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м       $z(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м  
 $v_x(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м/с       $v_y(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м/с       $v_z(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м/с  
 $v(\underline{\hspace{2cm}}) = \underline{\hspace{2cm}}$  м/с

## Задания на защиту лабораторной работы № 2

**Задание 1.** Определить положение в пространстве и скорость точки массой  $m =$  \_\_\_\_\_ кг в момент времени  $T =$  \_\_\_\_\_ с.



Сила  $F =$  \_\_\_\_\_ Н

$x(0) =$  \_\_\_\_\_ м,  $y(0) =$  \_\_\_\_\_ м,  $z(0) =$  \_\_\_\_\_ м

$v_x(0) =$  \_\_\_\_\_ м/с,  $v_y(0) =$  \_\_\_\_\_ м/с,  $v_z(0) =$  \_\_\_\_\_ м/с

Ответ:

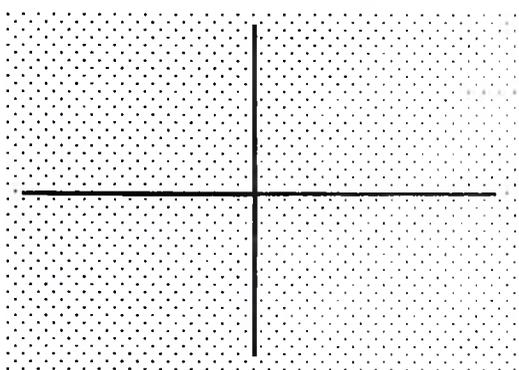
$x(\quad) =$  \_\_\_\_\_ м  $v_x(\quad) =$  \_\_\_\_\_ м/с

$y(\quad) =$  \_\_\_\_\_ м  $v_y(\quad) =$  \_\_\_\_\_ м/с

$z(\quad) =$  \_\_\_\_\_ м  $v_z(\quad) =$  \_\_\_\_\_ м/с

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 2.** Определить положение в пространстве и скорость точки массой  $m =$  \_\_\_\_\_ кг в момент времени  $T =$  \_\_\_\_\_ с.



Сила  $F =$  \_\_\_\_\_ Н

$x(0) =$  \_\_\_\_\_ м,  $y(0) =$  \_\_\_\_\_ м,  $z(0) =$  \_\_\_\_\_ м

$v_x(0) =$  \_\_\_\_\_ м/с,  $v_y(0) =$  \_\_\_\_\_ м/с,  $v_z(0) =$  \_\_\_\_\_ м/с

Ответ:

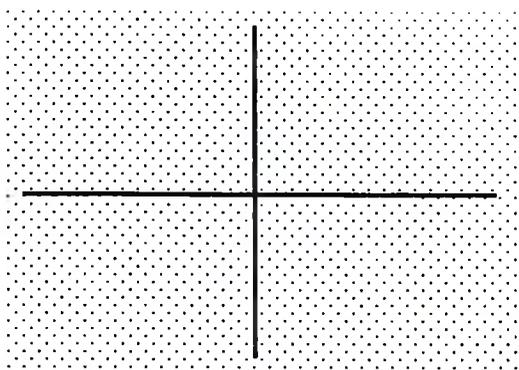
$x(\quad) =$  \_\_\_\_\_ м  $v_x(\quad) =$  \_\_\_\_\_ м/с

$y(\quad) =$  \_\_\_\_\_ м  $v_y(\quad) =$  \_\_\_\_\_ м/с

$z(\quad) =$  \_\_\_\_\_ м  $v_z(\quad) =$  \_\_\_\_\_ м/с

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 3.** Определить положение в пространстве и скорость точки массой  $m =$  \_\_\_\_\_ кг в момент времени  $T =$  \_\_\_\_\_ с.



Сила  $F =$  \_\_\_\_\_ Н

$x(0) =$  \_\_\_\_\_ м,  $y(0) =$  \_\_\_\_\_ м,  $z(0) =$  \_\_\_\_\_ м

$v_x(0) =$  \_\_\_\_\_ м/с,  $v_y(0) =$  \_\_\_\_\_ м/с,  $v_z(0) =$  \_\_\_\_\_ м/с

Ответ:

$x(\quad) =$  \_\_\_\_\_ м  $v_x(\quad) =$  \_\_\_\_\_ м/с

$y(\quad) =$  \_\_\_\_\_ м  $v_y(\quad) =$  \_\_\_\_\_ м/с

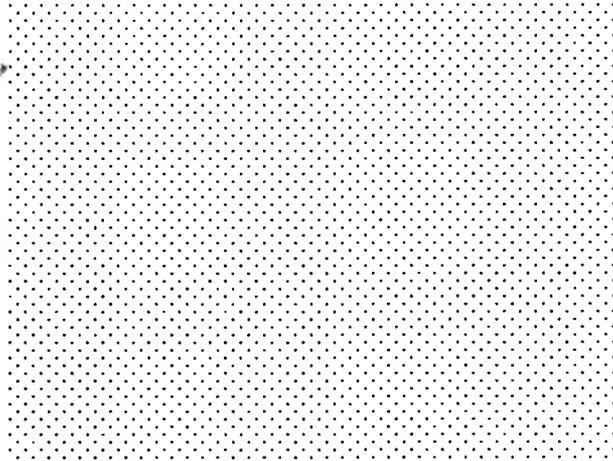
$z(\quad) =$  \_\_\_\_\_ м  $v_z(\quad) =$  \_\_\_\_\_ м/с

выдано: \_\_\_\_\_  
(дата) (подпись)

## Лабораторная работа № 3

**ТЕМА.** СКМ Maple: решение прикладных задач.

**Условие:** Груз  $D$  массой  $m = \underline{\hspace{2cm}}$  кг, получив в точке  $A$  начальную скорость  $v_0 = \underline{\hspace{2cm}}$  м/с, начинает движение в изогнутой трубе  $ABC$  ( $\alpha = \underline{\hspace{2cm}}$ °), расположенной в вертикальной плоскости.



указать все силы, действующие на груз

На участке  $AB$  длиной  $x_{AB} = \underline{\hspace{2cm}}$  м на груз кроме силы тяжести (ускорение свободного падения  $g$  принять равной  $9.8 \text{ м/с}^2$ ) действует постоянная сила  $Q = \underline{\hspace{2cm}}$  Н. В точке  $B$  груз, не изменяя своей скорости, переходит на участок  $BC$  трубы, где на него кроме силы тяжести действуют сила трения (коэффициент трения груза  $f = 0.1$  о поверхность трубы) и переменная сила  $F$ , проекция которой на ось движения  $x$  представлена функцией  $F(x) = \underline{\hspace{4cm}}$  Н.

**Задание: 1.** Определить закон движения груза  $D$  на участке  $AB$ .

Дифференциальное уравнение

\_\_\_\_\_

Начальные условия

\_\_\_\_\_

Уравнение движения  $x_{AB}(t) = \underline{\hspace{4cm}}$ ;

$v_{AB}(t) = \underline{\hspace{4cm}}$ ;  $a_{AB}(t) = \underline{\hspace{4cm}}$ .

**Задание: 2.** Установить время прохождения грузом  $D$  участка  $AB$ .

$T_{AB} = \underline{\hspace{2cm}}$  с.

**Задание: 3.** Определить закон движения груза  $D$  на участке  $BC$ , используя численное решение дифференциального уравнения.

Дифференциальное уравнение

\_\_\_\_\_

Начальные условия

\_\_\_\_\_

**Задание: 4.** Найти длину участка  $BC$ , скорость и ускорение груза  $D$  в точке  $C$ .

$x_C = \underline{\hspace{2cm}}$  м;  $v_C = \underline{\hspace{2cm}}$  м/с;  $a_C = \underline{\hspace{2cm}}$  м/с<sup>2</sup>;

$BC = \underline{\hspace{2cm}}$  м.



## Лабораторная работа № 4(1,2)

**ТЕМА.** Решение задач линейного программирования.

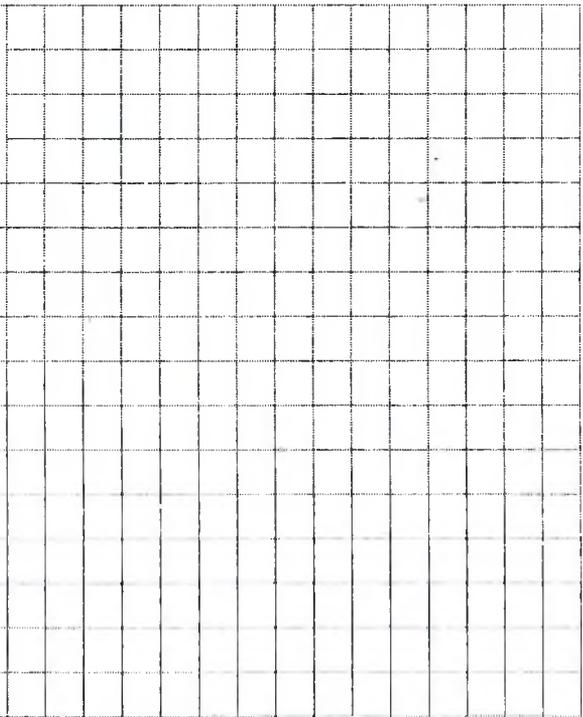
**Условие:** Решить задачу линейного программирования:

\_\_\_\_\_

Целевая функция:  $fZ(x, y) =$  \_\_\_\_\_.

$$\left. \begin{array}{l} \\ \\ \\ \end{array} \right\} x \geq 0, y \geq 0$$

**Задание: 1.** Получить графическое решение задачи.

<ul style="list-style-type: none"> <li>- подписать прямые;</li> <li>- заштриховать ОДЗ;</li> <li>- построить вектор градиента;</li> <li>- указать точки максимума и минимума решения ЗЛП.</li> </ul>	
--	---

$$f1(x) =$$

$$f2(x) =$$

$$f3(x) =$$

$$fc(x, C) =$$

$$grad fZ = \left( \quad \quad \right)$$

**Задание: 2.** Уточнить в MathCAD координаты вершины путем решения СЛАУ.

Пересечение функций \_\_\_\_\_ и \_\_\_\_\_. ЦФ (min):  $fZ =$  \_\_\_\_\_.

Пересечение функций \_\_\_\_\_ и \_\_\_\_\_. ЦФ (max):  $fZ =$  \_\_\_\_\_.

**Задание: 3.** Уточнить в MathCAD и Excel координаты вершины.

Исследование на минимум. Начальные значения  $x =$  \_\_\_\_\_,  $y =$  \_\_\_\_\_.

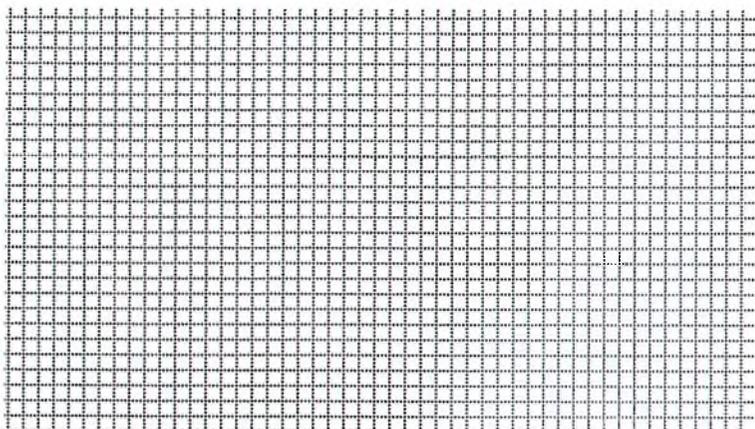
Уточненные значения:  $x =$  \_\_\_\_\_,  $y =$  \_\_\_\_\_. ЦФ (min):  $fZ =$  \_\_\_\_\_.

Исследование на максимум. Начальные значения  $x =$  \_\_\_\_\_,  $y =$  \_\_\_\_\_.

Уточненные значения:  $x =$  \_\_\_\_\_,  $y =$  \_\_\_\_\_. ЦФ (max):  $fZ =$  \_\_\_\_\_.

## Задания на защиту лабораторной работы № 4(1,2)

**Задание 1.** Решить задачу линейного программирования с целевой функцией \_\_\_\_\_, исключив из основной системы ограничений задачи из лабораторной работы ограничение номер \_\_\_\_\_.



Координаты вершины:

$$x = \underline{\hspace{2cm}},$$

$$y = \underline{\hspace{2cm}}.$$

(отметить на графике)

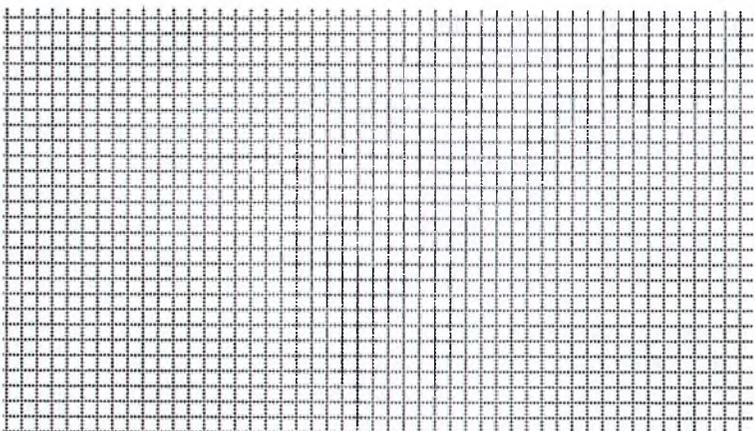
Целевая функция:

$$fZ = \underline{\hspace{2cm}}.$$

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

---

**Задание 2.** Решить задачу линейного программирования из лабораторной работы, изменив знак неравенства в ограничениях номер \_\_\_\_\_ и \_\_\_\_\_ на обратный.



Координаты вершины:

$$x = \underline{\hspace{2cm}},$$

$$y = \underline{\hspace{2cm}}.$$

(отметить на графике)

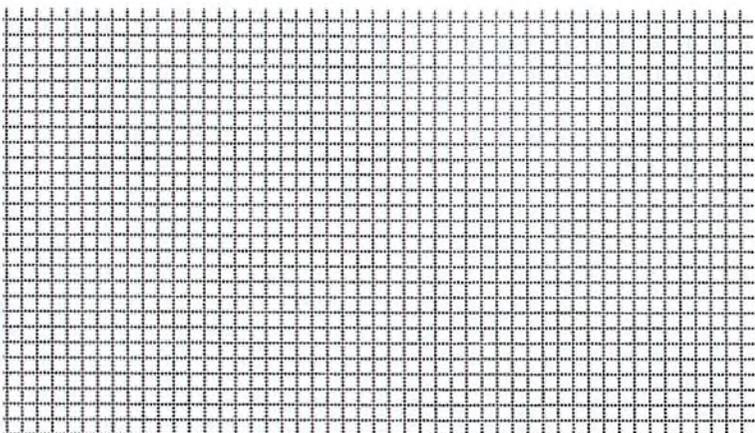
Целевая функция:

$$fZ = \underline{\hspace{2cm}}.$$

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

---

**Задание 3.** Решить задачу линейного программирования из лабораторной работы, изменив знак неравенства в ограничениях номер \_\_\_\_\_ и \_\_\_\_\_ на обратный.



Координаты вершины:

$$x = \underline{\hspace{2cm}},$$

$$y = \underline{\hspace{2cm}}.$$

(отметить на графике)

Целевая функция:

$$fZ = \underline{\hspace{2cm}}.$$

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)



## Задания на защиту лабораторной работы № 5

**Задание 1.** Дополнительное условие к задаче: \_\_\_\_\_

План: \_\_\_\_\_

При этом: общее количество отходов равно \_\_\_\_\_ м,

из них: \_\_\_\_\_ м отходов, полученных от разреза брусков;

\_\_\_\_\_ м отходов, полученных от изготовления лишних заготовок

в количестве  $d_1 =$  \_\_\_\_\_ шт,  $d_2 =$  \_\_\_\_\_ шт,  $d_3 =$  \_\_\_\_\_ шт.

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

**Задание 2.** Дополнительное условие к задаче: \_\_\_\_\_

План: \_\_\_\_\_

При этом: общее количество отходов равно \_\_\_\_\_ м,

из них: \_\_\_\_\_ м отходов, полученных от разреза брусков;

\_\_\_\_\_ м отходов, полученных от изготовления лишних заготовок

в количестве  $d_1 =$  \_\_\_\_\_ шт,  $d_2 =$  \_\_\_\_\_ шт,  $d_3 =$  \_\_\_\_\_ шт.

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

**Задание 3.** Дать пояснения выражениям из математической модели:

---

---

---

---

---

---

---

---

---

---

выдано: \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

## Лабораторная работа № 6

**ТЕМА.** Решение задач математического программирования.

**Условие:** Составить план поставок строительного песка на объекты строительной организации на очередную рабочую смену (8 часов). Поставки планируется осуществлять с  $m$  карьеров  $A_1, \dots, A_m$ , способных производить по  $a_1, \dots, a_m$  т в смену соответственно. Заявки поступили с  $n$  объектов  $B_1, \dots, B_n$  в количествах  $b_1, \dots, b_n$  т соответственно (для непрерывной работы объектов). Известны: стоимость  $c_{ij}$  ден. ед. за перевозку одной тонны песка с каждого  $i$ -го ( $i = 1, \dots, m$ ) карьера на каждый  $j$ -й ( $j = 1, \dots, n$ ) объект; затраты  $s_i$  ден. ед. на добычу одного тонна песка на каждом  $i$ -ом ( $i = 1, \dots, m$ ) карьере; штраф  $p_j$  ден. ед. за один час простоя на  $j$ -ом ( $j = 1, \dots, n$ ) строительном объекте.

Составить план перевозок, который полностью удовлетворяет спрос объектов в строительном песке, а суммарные издержки за добычу песка и его перевозку при этом минимизируются.

**Задание: 1.** Привести математическую модель задачи в скалярной форме

ЦФ = \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  $\rightarrow \min$

Ограничения:

{ \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

**2.** Найти, воспользовавшись надстройкой «поиск решения», в Excel оптимальный план перевозок.

Суммарные издержки корпорации составляют \_\_\_\_\_ ден. ед.,  
из них:

расходы на производство продукции – \_\_\_\_\_ ден. ед.;

транспортные расходы – \_\_\_\_\_ ден. ед.;

штраф за \_\_\_\_\_ – \_\_\_\_\_ ден. ед.

План перевозок от предприятий к потребителям в смену:

	$V_1$	$V_2$					предложение
$A_1$							
$A_2$							
спрос							

Предприятие \_\_\_\_\_

### Задания на защиту лабораторной работы № 6

**Задание 1.** Дополнительное условие к задаче: \_\_\_\_\_

Суммарные издержки корпорации составляют \_\_\_\_\_ ден. ед.,  
из них:

расходы на производство продукции – \_\_\_\_\_ ден. ед.;

транспортные расходы – \_\_\_\_\_ ден. ед.;

штраф за \_\_\_\_\_ – \_\_\_\_\_ ден. ед.

выдано: \_\_\_\_\_  
(дата) (подпись)

**Задание 2.** Дать пояснения выражениям из математической модели:

---



---



---



---



---



---



---



---



---



---

выдано: \_\_\_\_\_  
(дата) (подпись)

## Список дополнительной литературы

1. Савотченко, С.Е. Методы решения математических задач в Maple: Учебное пособие / С.Е. Савотченко, Т.Г. Кузьмичева. – Белгород : Белаудит, 2001. – 116 с.
2. Уокенбах, Дж. Excel 2013. Профессиональное программирование на VBA в Excel 2003. : Пер. с англ. / Дж. Уокенбах. – М. : Издательский дом «Вильямс», 2005. – 800 с.
3. Макаров, Е.Г. Mathcad: Учебный курс (+CD) / Е.Г. Макаров. – СПб. : Питер, 2009. – 384 с.

## Оглавление

Общие указания.....	3
Отметки о защите лабораторных работ .....	4
Методические указания к выполнению лабораторных работ .....	5
СИСТЕМА КОМПЬЮТЕРНОЙ МАТЕМАТИКИ MAPLE .....	5
ЭЛЕКТРОННАЯ ТАБЛИЦА MICROSOFT EXCEL .....	13
СИСТЕМА КОМПЬЮТЕРНОЙ МАТЕМАТИКИ MATHCAD.....	16
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ .....	18
Использование Maple в задачах по теоретической механике.....	23
ДОПОЛНИТЕЛЬНАЯ ИНФОРМАЦИЯ.....	26
Лабораторная работа № 1(1) .....	30
Лабораторная работа № 1(2) .....	31
Задания на защиту лабораторной работы № 1(1,2).....	31
Лабораторная работа № 2.....	32
Задания на защиту лабораторной работы № 2 .....	33
Лабораторная работа № 3 .....	34
Задания на защиту лабораторной работы № 3 .....	35
Лабораторная работа № 4(1,2) .....	36
Задания на защиту лабораторной работы № 4(1,2).....	37
Лабораторная работа № 5 .....	38
Задания на защиту лабораторной работы № 5 .....	39
Лабораторная работа № 6.....	40
Задания на защиту лабораторной работы № 6 .....	41
Список дополнительной литературы .....	42

УЧЕБНОЕ ИЗДАНИЕ

**Составители:**

Кофанов Валерий Анатольевич

Хомицкая Татьяна Георгиевна

Сидак Светлана Васильевна

# ЛАБОРАТОРНЫЙ ПРАКТИКУМ

по дисциплине «Информатика» для студентов  
строительных специальностей очной формы обучения  
третьего семестра

*издание 2-е, дополненное и переработанное*

*Текст печатается в авторской редакции*

Ответственный за выпуск: Кофанов В.А.

Редактор: Боровикова Е.А.

Компьютерная верстка: Соколюк А.П.

---

Подписано в печать 23.08.2017 г. Формат 60x84 <sup>1</sup>/<sub>8</sub>. Бумага «Performer».  
Гарнитура «Times New Roman». Усл. печ. л. 5,11. Уч. изд. л. 5,5. Заказ № 835. Тираж 150 экз.  
Отпечатано на ризографе учреждения образования «Брестский государственный  
технический университет». 224017, г. Брест, ул. Московская, 267.