

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Локтионова Оксана Геннадьевна

Должность: проректор по учебной работе

Дата подписания: 16.09.2023 20:15:33

Уникальный программный ключ:

0b817ca911e6668abb13a5d426d39c5f1c11eabbf73e943df4a4851fda56d089

## МИНОБРАЗОВАНИЯ И НАУКИ РОССИИ

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ

Проректор по учебной работе

О.Г. Локтионова

« 14 »

Г.



### МОДЕЛИРОВАНИЕ ОДНОКАНАЛЬНЫХ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ С ОТКАЗАМИ В СИСТЕМЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ARENA

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

УДК 519.6

Составители: В.В. Куц

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Моделирование одноканальных систем массового обслуживания с отказами в системе имитационного моделирования ARENA: методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 37 с.**

Содержат сведения о моделировании одноканальных систем массового обслуживания с отказами в системе имитационного моделирования ARENA. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 2,15 . Уч.-изд. л. 1,95.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

**Цель работы:** Получить практические навыки моделирования одноканальных систем массового обслуживания с отказами в системе имитационного моделирования Arena.

**Задание:**

1) Изучить основные сведения о работе с системой Arena представленные в разделе 1.

2) Выполнить пример моделирования одноканальных систем массового обслуживания с отказами в разделе 2.

3) Для поставленной задачи, в соответствии со своим вариантом (см. раздел 3), построить модель одноканальной системы массового обслуживания с отказами

4) Оформить отчет.

## **1 Система имитационного моделирования ARENA. Basic Process Panel**

Система имитационного моделирования Arena – основной программный продукт корпорации Systems Modeling. В основе ПП Arena лежит язык SIMAN.

Arena позволяет создавать динамические модели разнородных процессов и систем в терминах предметной области, оптимизировать построенную модель. Программный продукт Arena позволяет создавать стохастические динамические модели любого предприятия. Динамические модели позволяют оптимизировать, прогнозировать, планировать деятельность предприятий, а также проводить анализ деятельности предприятия на основании полученных моделей и выдавать рекомендации по улучшению работы конкретного предприятия.

В результате моделирования Arena формирует отчеты, в которых можно посмотреть отчеты по:

1) объектам, находящимся в системе (общее время нахождения в системе, суммарное время ожидания объекта в системе, количество объектов, вошедших/вышедших в систему/из системы);

2) очередям, образующихся в модулях процессов, если ресурс захвачен другим объектом (время ожидания обработки в очереди, количество объектов, ожидающих в очереди);

3) процессам – статистика для каждого повторения;  
ресурсам – статистика по затраченным ресурсам;

4) пользователь сам определяет статистику, которая будет отражена в данном отчете.

По результатам собранной статистики если не устраивают какие-либо параметры, то изменение свойств модулей приведет к изменению всех параметров. Т.е. методом прогона различных вариаций параметров модели можно определить оптимальный вариант работы созданной системы. В этом и есть плюс, т.к. шлифовка работы предприятия делается сначала на модели и только затем внедряется на производство.

Также существует возможность экспорта статических моделей VPWin в динамические модели Arena.

## **1.1 НАЧАЛО РАБОТЫ С ARENA**

### **1.1.1 Создание модели с помощью Arena**

Для того чтобы создать новую модель необходимо открыть ПП Arena через Пуск → Rockwell Software → Arena → Arena. После запуска Arena автоматически открывается новый файл. Модули помещаются на панель методом «drag & drop», соединяются с помощью коннектора (рисунок). Если модуль остается «горячим» (т.е. выделенным), то при помещении нового модуля на рабочую область (окно блок–схемы) эти модули автоматически соединяются друг с другом.

### **1.1.2 Интерфейс ПП Arena**

Среда моделирования Arena представлена на рис. 1.

Окно приложения разделено на три области:

- 1) окно рабочего поля;
- 2) окно свойств модулей;
- 3) окно проекта.

Окно проекта включает в себя несколько панелей:

Basic Process (панель основных процессов) – содержит модули, которые используются для моделирования.



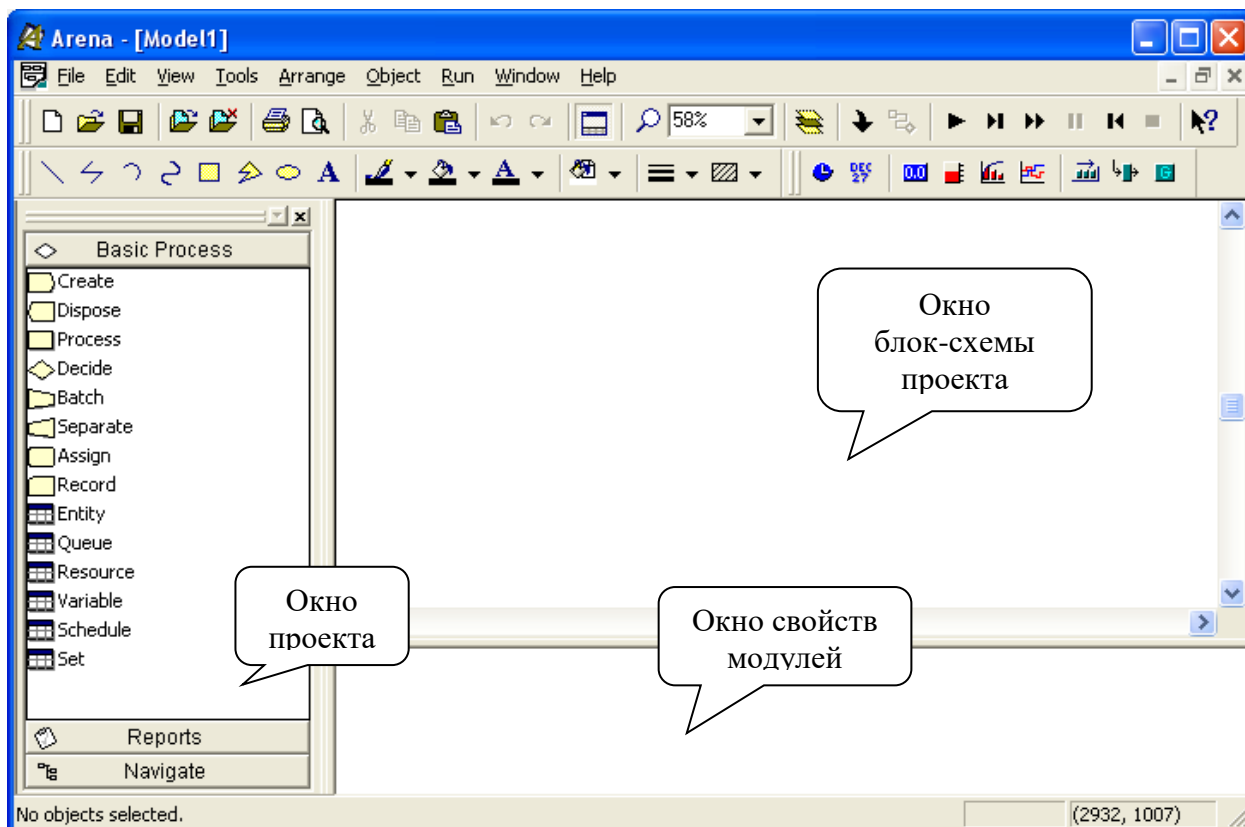


Рисунок 1 – Среда моделирования Arena

Reports (панель отчетов) – панель сообщений: содержит сообщения, которые отображают результаты имитационного моделирования.

Navigate (панель навигации) – панель управления позволяет отображать все виды модели, включая управление через иерархические подмодели.

Окно рабочего поля представляет графику модели, включая блок-схему процесса, анимацию и другие элементы.

Окно свойств модуля служат для настройки параметров модели таких как: время, издержки и другие параметры.

## 1.2 ПАНЕЛЬ ОСНОВНЫХ ПРОЦЕССОВ

### 1.2.1 Графические модули

#### 1.2.1.1 Модуль Create

Этот модуль (рис. 2, табл. 1) является отправной точкой для сущностей в имитационной модели. Сущности – это индивидуаль-

ные элементы, обрабатываемые в системе. Создание сущностей модулем происходит по расписанию, или же основываясь на значении времени между прибытиями сущности в модель. Покидая модуль, сущности начинают обрабатываться в системе. Тип создаваемых сущностей определяется в этом модуле.

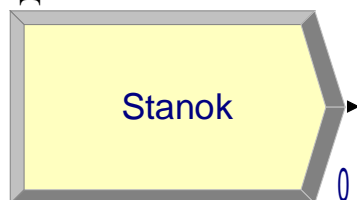


Рисунок 2 – Модуль Create

Применение:

- прибытие различных документов в сфере бизнеса (например: заказы, чеки, документация);
- прибытие клиентов в сфере обслуживания (например: в ресторан, в магазин);
- начало изготовления продукции на производственной линии.

Таблица 1

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Entity Type	Название типа сущности, который будет создаваться модулем
Type	Способ формирования потока прибытия. <i>Type</i> может иметь значение <i>Random</i> (используется экспоненциальное распределение со средним значением, определенным пользователем), <i>Schedule</i> (определяется модулем <i>Schedule</i> ), <i>Constant</i> (будет использоваться, определенное пользователем, постоянное значение; например, 100) или <i>Expression</i> (поток прибытия будет формироваться по определенному выражению)
Value	Определяет среднее значение экспоненциального распределения ( <i>Random</i> ) или постоянное значение времени между прибытиями сущностей (если <i>Type</i> = <i>Constant</i> )
Schedule Name	Имя расписания, которое определяет характер прибытия сущности в систему
Expression	Этот параметр задает тип распределения или выражение, определяющее время между прибытиями сущностей в модель
Units	Единицы измерения времени между прибытиями (день, час, минута, секунда)

Параметры	Описание
Entities per arrival	Количество сущностей входящих в систему за одно прибытие
Max arrivals	Максимальное число сущностей, которое может создать этот модуль
First Creation	Время, через которое прибудет первая сущность в модель от начала симуляции

### 1.2.1.2 Модуль Process

Этот модуль (рис. 3, табл. 2) является основным модулем процесса обработки в имитационной модели. В модуле имеются опции использования ресурсов. Кроме стандартного модуля Process, можно использовать подмодель, придавая ей особую, определенную пользователем, иерархическую логическую схему. В модуле можно также задавать добавочные стоимостные и временные характеристики процесса обработки сущности.

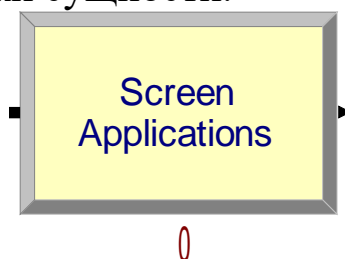


Рисунок 3 – Модуль Process

Наиболее частое применение модуля Process:

- проверка документов;
- выполнение заказов;
- обслуживание клиентов;
- обработка деталей.

Таблица 2

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Type	Определяет логическую схему модуля. <i>Standard</i> означает, что логическая схема находится внутри модуля и зависит от параметра <i>Action</i> . <i>Submodel</i> показывает, что логическая схема будет находиться ниже в иерархической модели. Подмодель может содержать любое количество логических модулей

Параметры	Описание
Action	Тип обработки происходящей внутри модуля. <i>Delay</i> просто показывает, что процесс занимает какое-то время и не отражает использование ресурсов. <i>Seize Delay</i> указывает на то, что в этом модуле были размещены ресурсы и будет происходить задержка, ресурсы будут захватываться (то есть будут заняты обработкой сущности), и их освобождение будет происходить позднее. <i>Seize Delay Release</i> указывает на то, что ресурс(-ы) были захвачены, а затем через время освободились. <i>Delay Release</i> означает, что ресурсы до этого были захвачены сущностью, а в таком модуле сущность задержится и освободит ресурс. Все эти параметры доступны только тогда, когда <i>Type = Standard</i>
Priority	Значение приоритета модулей использующих один и тот же ресурс где угодно в модели. Это свойство не доступно, если Action = <i>Delay</i> или <i>Delay Release</i> , или когда <i>Type = Submodel</i>
Resources	Определяет ресурсы или группы ресурсов, которые будут обрабатывать сущности в этом модуле (см. Модуль Process - Ресурсы)
Delay Type	Тип распределения или процедура, определяющая параметры задержки
Units	Единицы измерения времени задержки (день, час, минута, секунда)
Allocation	Определяет стоимостные характеристики обработки. Value Added - означает учитывать стоимостные характеристики, а Non-Value Added не учитывать
Minimum	Поле, определяющее минимальное значение для равномерного и треугольного распределения
Maximum	Поле, определяющее максимальное значение для равномерного и треугольного распределения
Value	Поле, определяющее среднее значение для нормального и треугольного распределения или значения для постоянной временной задержки
Std Dev	Параметр, определяющий стандартное отклонение для нормального распределения
Expression	Поле, в котором задается выражение, определяющее значение временной задержки, если <i>Delay Type = Expression</i>

### 1.2.1.3 Модуль *Decide*

Этот модуль (рис. 4, табл. 3) позволяет учитывать принятие решений в модели. Он включает опции принятия решений основанных на условии *By Condition* (например, если тип сущности *Car*) или основанных на вероятности *By Chance* (например, 75% - true, а 25% - false). Условия могут быть основаны на значении атрибута *Attribute*, значении переменной *Variable*, типе сущности *Entity Type* или основанные на выражении *Expression*.

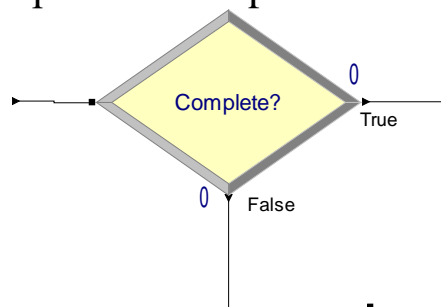


Рисунок 4 – Модуль *Decide*

Если поставленное условие не выполняется то, сущности будут покидать модуль через ветку *False*.

Данный модуль позволяет выполнять проверку не только одного условия, но и нескольких. Это достигается с помощью свойства *Type* → *N-way by Chance/by Condition*. В зависимости от условия сущность идет по нужной ветке.

Применение:

- разделение дел на срочные дела и несрочные;
- перенаправление недоделанных или сделанных неправильно работ на доработку.

Таблица 3

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Type	Тип принятия решения. <i>By Chance</i> - выбор направления основывается на вероятности. <i>By Condition</i> - проверка на выполнение условия
Percent True	Значение, определяющее процент сущностей, который пойдут по направлению True
If	Тип условия, которое будет проверяться на выполнение

Параметры	Описание
Named	Имя переменной, атрибута или типа сущности, который будут проверяться при входе сущности в модуль
Is	Математический знак условия, например больше, меньше, равно и т.д.
Value	Значение, с которым будет сравниваться атрибут или переменная пришедшей сущности. Если тип условия - Expression, то в выражении должен стоять знак условия, например Color<> Red

#### 1.2.1.4 Модуль Batch

Этот модуль (рис. 5, табл. 4) отвечает за механизм группировки в имитационной модели. Группировка может быть постоянной или временной. Временно сгруппированные комплекты позднее могут быть разъединены с помощью модуля Separate. Комплекты могут состоять из любого числа входящих сущностей, определенного пользователем или же сущности могут объединяться в комплект в зависимости от атрибута сущности. Временные и стоимостные характеристики выходящей сущности, представляющей комплект будут равны сумме характеристик вошедших в группу сущностей.

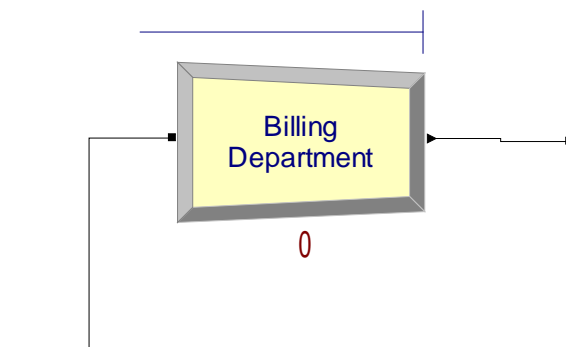


Рисунок 5 – Модуль Batch

Сущности прибывают в модуль, становятся в очередь и остаются там до тех пор, пока в модуле не будет набрано заданное количество сущностей. Когда соберется нужное число сущностей создается сущность представляющая комплект.

Применение:

- собрать необходимое количество данных, прежде чем начинать их обработку;

- собрать ранее разделенные копии одной формы;
- соединить пациента и его больничную карту приема к врачу.

Таблица 4

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок-схеме
Type	Способ группировки сущностей, может быть <i>Temporary</i> (временная), <i>Permanent</i> (постоянная)
Batch Size	Число сущностей, образующих один комплект
Rule	Определяет, по какому признаку будут группироваться. Если <i>Rule = Any Entity</i> , это значит, что первые 3 (если <i>Batch Size = 3</i> ) сущности будут сгруппированы. Если <i>Rule = By Attribute</i> , то будет объединяться заданное количество сущностей с определенным атрибутом. Например, если <i>Attribute Name = Color</i> , то все сущности, имеющие одинаковое значение атрибута <i>Color</i> , будут сгруппированы
Attribute Name	Имя атрибута, по значению которого будут группироваться сущности

### 1.2.1.5 Модуль *Separate*

Этот модуль (рис. 6, табл. 5) может использоваться как для создания копий входящих сущностей, так и для разделения ранее сгруппированных сущностей. Правило для разделения стоимостных и временных характеристик копий сущностей и разделенных сущностей определяется пользователем. Когда временно сгруппированные сущности прибывают в модуль, они раскладываются на составные сущности. Сущности покидают модуль в той же последовательности, в которой они добавлялись в комплект.

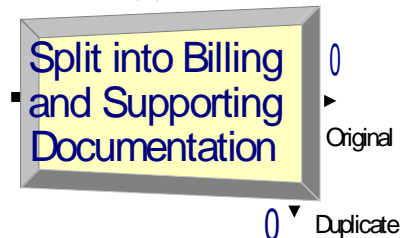


Рисунок 6 – Модуль *Separate*

Если модуль создает копии сущностей, то пользователь может задать количество дубликатов сущности. У дублированной сущности значения атрибута, а также анимационная картинка такие же, как и оригинала. Оригинальная сущность также покидает модуль.

Применение:

- разъединение ранее сгруппированных комплектов документов;
- для параллельной обработки счетов и документов по одному заказу.

Таблица 5

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
# of Duplic	Количество создаваемых копий входящей сущности
Type	Способ разделение входящей в модуль сущности. <i>Duplicate Original</i> - просто делает дубликаты входящей сущности. <i>Split Existing Batch</i> требует чтобы входящая сущность была предварительно временно сгруппирована
Cost to Duplicates	Разделение стоимостных и временных характеристик входящей сущности между выходящими. Это значение определяется пользователем в процентах, т.е. сколько процентов от стоимостных и временных характеристик входящей сущности уйдет копиям (характеристики между копиями делятся поровну)
Allocation Rule	Метод разделения стоимости и времени, если выбран Type=Split Existing Batch. Retain Original Entity Values - сохраняет оригинальные значения сущностей. Take All Representative Values - все сущности принимают одинаковое значение. Take Specific Representative Values - сущности принимают специфическое значение

### 1.2.1.6 Модуль Assign

Этот модуль (рис. 7, табл. 6) предназначен для задания нового значения переменной, атрибуту сущности типу сущности, анимационной картинке сущности или другой переменной в системе.

В одном модуле можно сделать только одно назначение. Пример применения модуля Assign:

- установление приоритета для клиентов;
- присвоение номера вышедшему приказу.



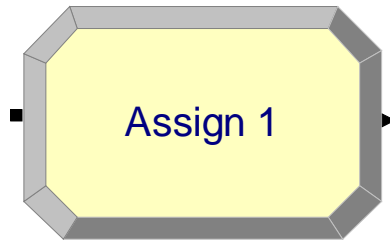


Рисунок 7 – Модуль Assign

Таблица 6

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Type	Тип назначения, которое будет осуществляться. <i>Other</i> может включать в себя встроенные в Арену переменные, такие как вместимость ресурса или конечное время симуляции
Variable Name	Имя переменной, которая будет изменяться в этом модуле
Attribute Name	Имя атрибута, который будет изменяться в этом модуле
Entity Type	Новый тип сущности, присваиваемый сущности в этом модуле
Entity Picture	Новая анимационная картинка для сущности, прошедшей этот модуль
Other	Имя переменной в системе, которая будет меняться
New Value	Присваиваемое новое значение для атрибута, переменной

### 1.2.1.7 Модуль Record

Этот модуль (рис. 8, табл. 7) предназначен для сбора статистики в имитационной модели. Модуль может собирать различные типы статистики, включая время между выходами сущностей из модуля, статистику сущности (время цикла, стоимость), статистику за период времени (период времени от заданной точки до текущего момента). Также доступен количественный тип статистики.

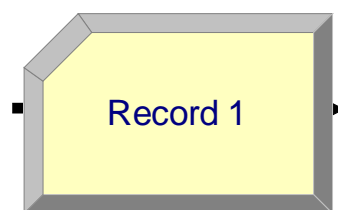


Рисунок 8 – Модуль Record

Частое применение модуля:

- подсчет количества заказов, выполненных с опозданием;
- подсчет количества работы, совершаемого за один час.

Таблица 7

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Type	Определяет тип статистики, которая будет собираться. <i>Count</i> будет увеличивать или уменьшать статистику на заданное значение. <i>Entity Statistics</i> будет собирать общую статистику о сущности, например, время цикла, стоимостные характеристики и т. д. <i>Time Interval</i> будет считать разницу между значением атрибута и текущим временем моделирования. <i>Time Between</i> будет отслеживать время между вхождением сущностей в модуль. <i>Expression</i> будет просто фиксировать значение определяемое выражением
Attribute Name	Имя атрибута, значение которого будет использоваться для интервальной статистики
Value	Значение, которое будет добавляться к статистике, когда в модуль будет прибывать сущность

### 1.2.1.8 Модуль Dispose

Этот модуль (рис. 9, табл. 8) является выходной точкой из имитационной модели. Статистика о сущности может собираться до того момента пока она не выйдет из системы.

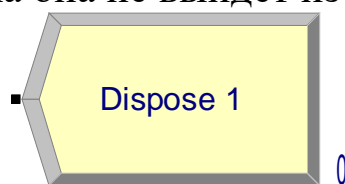


Рисунок 9 – Модуль Dispose

Применение:

- Окончание бизнес процесса;
- Клиенты покидают отдел.

Таблица 8

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Record Entity Statistics	Определяет, будет ли вестись статистика о выходе сущности из системы

## 1.2.2 Модули данных

### 1.2.2.1 Модуль Entity

Этот модуль (табл. 9) определяет тип сущности и ее анимационную картинку в имитационном процессе, также определяет стоимостную информацию.

Для каждого источника должен быть определен тип сущности, который он генерирует.

Применение модуля Entity:

- Документы: факсы, письма, отчеты и т. д.;
- Люди в моделях больницы или магазина.

Таблица 9

Параметры	Описание
Entity Type	Название типа сущности
Initial Picture	Графическое представление сущности в начале имитационного процесса. Это значение может быть впоследствии изменено с помощью модуля Assign. Просмотреть анимационные картинки можно так: Edit/ Entity picture
Holding Cost/Hour	Почасовая стоимость обработки сущности в системе. Эта стоимость учитывается, когда сущность находится в системе, либо в очереди, либо в стадии обработки
Initial VA Cost	Значение, присваиваемое атрибуту сущности «добавочная стоимость». Значение атрибута будет увеличиваться каждый раз, как только сущность будет обрабатываться процессом с добавочной стоимостью
Initial NVA Cost	Значение, присваиваемое атрибуту сущности «не добавочная стоимость». Значение атрибута будет увеличиваться каждый раз, как только сущность будет обрабатываться процессом с не добавочной стоимостью

### 1.2.2.2 Модуль Queue

Этот модуль данных (табл. 10) предназначен для изменения правила расстановки сущностей в очереди. По умолчанию тип очереди First in First out.

Применение:

- Стопка документов, ожидающих освобождения ресурса;
- Место для собирания частей, ожидающих упаковки (группировки).

Таблица 10

Параметры	Описание
Name	Уникальное имя модуля, которое будет отражено в блок схеме
Attribute Name	Имя атрибута, значение которого будет учитываться, если тип = <i>Lowest Attribute Value</i> или <i>Highest Attribute Value</i>
Type	Правило расстановки сущностей в очереди. <i>First in First out</i> - первый вошел, первый вышел. <i>Last in first out</i> - последний пришел, первый вышел. <i>Lowest Attribute Value</i> - первый выйдет из очереди тот, значение атрибута у которого ниже. <i>Highest Attribute Value</i> - первый выйдет из очереди тот, значение атрибута у которого наивысшее

### 1.2.2.3 Модуль Resource

Этот модуль (табл. 11) предназначен для определения ресурсов и их свойств в имитационном процессе, кроме того, модуль включает в себя стоимостную информации о ресурсах и вместимость ресурсов. Ресурсы могут иметь фиксированную вместимость или же основанную на расписании. У ресурсов с фиксированной вместимостью в течение имитационного процесса вместимость изменяться не может.

Применение:

- Люди (клерки, продавцы, бухгалтеры, рабочие и т. д.);
- Оборудование (телефонная линия, станок, компьютер).

Таблица 11

Параметры	Описание
Name	Имя ресурса
Type	Метод, определяющий вместимость ресурса. <i>Fixed Capacity</i> - фиксированная вместимость ресурса. <i>Based on Schedule</i> - вместимость ресурса определяется модулем <i>Schedule</i>
Capacity	Число ресурсов, находящихся в системе
Schedule Name	Имя Schedule модуля, который определяет вместимость ресурса, если Type = <i>Based on Schedule</i>
Busy/ Hour	Почасовая стоимость обработки сущности ресурсом. Время учитывается только тогда, когда ресурс занят обработкой и прекращает учитываться, когда ресурс освобождается
Idle/ Hour	Стоимость ресурса, когда он не занят

Параметры	Описание
Per Use	Стоимость обработки ресурсом одной сущности (не зависит от времени)

#### 1.2.2.4 Модуль *Schedule*

Этот модуль (табл. 12) может использоваться вместе с модулем *Resource* для определения вместимости ресурса. Также модуль используется вместе с модулем *Create* для задания расписания прибытия сущностей.

Применение:

- Расписание работы персонала с перерывами на обед;
- Значение покупателей прибывающих в супермаркет.

Таблица 12

Параметры	Описание
Name	Название расписания
Type	Тип расписания, который может быть <i>Capacity</i> (расписание для ресурсов), <i>Arrival</i> (для модуля <i>Create</i> ) или <i>Other</i> (разнообразные временные задержки или факторы)
Time Units	Масштаб оси времени в графике расписания

#### 1.2.2.5 Модуль *Set*

Этот модуль данных (табл. 13) описывает группу ресурсов, использующихся в модуле *Process*. В группе могут находиться несколько ресурсов. Модуль *set* автоматически создает ресурсы, вместимость которых по умолчанию равна 1 и без всякой стоимостной информации. Следовательно, если для ресурсов входящих в группу не нужно стоимостной информации и вместимость более 1, то можно обойтись созданием только модуля *Set*.

Возможно применение модуля для организации работы группы работников, например, по очереди.

Таблица 13

Параметры	Описание
Name	Название группы

Параметры	Описание
Members	Перечисляет ресурсы, входящие в группу. Порядок перечисления ресурсов важен, когда в модуле Process используется правило выбора Cyclical или Preferred Order
Resource Name	Названия ресурсов входящих в группу

### 1.2.2.6 Модуль Variable

Этот модуль данных (табл. 14) определяет значение переменной. Переменные, относящиеся к модулю Decide или Assign, могут использоваться в выражениях. Если переменная не описана в этом модуле, то ее значение равно 0.

Применение:

- Число документов обрабатываемых в час;
- Присвоение серийного номера для идентификации продукции.

Таблица 14

Параметры	Описание
Name	Имя переменной
Initial Value	Первоначальное значение переменной. Это значение в последствии может меняться модулем <i>Assign</i>
Rows	Число строк в размерной переменной
Columns	Число столбцов в размерной переменной
Clear Option	Определяет время, когда значение переменной сбрасывается в начальное значение. <i>Statistics</i> - сбрасывает переменную в начальное значение в любой момент, когда статистика была расчищена. <i>System</i> - сбрасывает переменную в начальное значение в любой момент, когда система была расчищена. <i>None</i> - никогда не сбрасывает переменную в начальное значение, исключая предшествующую первой репликации
Statistics	Определяет, будет ли вестись статистика по этой переменной

### 1.3 ПАНЕЛЬ ОТЧЕТОВ

С помощью панели отчетов можно просмотреть результаты проигрывания имитационной модели. На панели отчетов представлены несколько видов отчетов. Отчет «Краткий обзор категорий» и отчеты по четырем категориям, такие как Сущности, Процессы, Очереди и Ресурсы.

*Отчет Category Overview категорий (Краткий обзор категорий)*

Отчет Category Overview отражает итоговую информацию о сущностях, процессах, очередях и ресурсах. Также показывает информацию о заданных пользователем переменных и информацию, собранную модулем Record.

*Отчет о сущностях (Entity)*

Отчет о сущностях разделен на несколько частей.

Cycle Time: в этой части отчета показано среднее, максимальное и минимальное время существования сущности. Время существования сущности считается с момента её прибытия в систему и до того момента, когда сущность попадает в модуль Dispose. Ниже представляется гистограмма среднего времени цикла для каждого типа сущности.

NVA Cost: в этой части показано среднее, максимальное и минимальное значение не добавочной стоимости сущностей по каждому типу сущностей. Не добавочная стоимость рассчитывается на основании значения *NVA Time*.

Total Cost: в этой части показано среднее, максимальное и минимальное значение общей стоимости сущностей по каждому типу сущностей. Общая стоимость вычисляется путем сложения стоимости ожидания, добавочной стоимости и не добавочной стоимости для каждой сущности. Ниже представляется сравнительные гистограммы средних значений общей стоимости сущности для каждого типа сущности.

VA Cost: в этой части показано среднее, максимальное и минимальное значение добавочной стоимости сущностей по каждому типу сущностей. Добавочная стоимость рассчитывается на основании *VA Time*.

Wait Cost: в этой части показано среднее, максимальное и минимальное значение стоимости ожидания сущностей по каждому

типу сущностей. Стоимость ожидания подсчитывается исходя из времени ожидания, стоимости ресурса и стоимостью нахождения сущности в системе.

Wait Time: в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей. Время ожидания – это период времени с момента поступления сущности в очередь (либо в модуле Process ожидает ресурс, либо в модуле Batch ожидает группировки) и до момента выхода из нее (начнет обрабатываться, либо будет сгруппирована).

WIP (Work In Process): в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей.

#### Отчет о процессах

Отчет о процессах разделен на несколько частей.

Cycle Time: в этой части отчета показано среднее, максимальное и минимальное время цикла процесса. Время цикла процесса считается с момента прибытия в модуль Process и до того момента, когда сущность покидает модуль. В разделе представляется гистограмма среднего времени цикла для каждого процесса.

NVA Cost: в этой части показано среднее, максимальное и минимальное значение недобавочной стоимости сущностей по процессу. Не добавочная стоимость рассчитывается на основании *NVA Time*.

Total Cost: в этой части показано среднее, максимальное и минимальное значение общей стоимости сущностей по процессу. Общая стоимость вычисляется путем сложения стоимости ожидания, добавочной стоимости и не добавочной стоимости для каждой сущности. В разделе представляются сравнительные гистограммы средних значений общей стоимости сущности для каждого типа сущности.

VA Cost: в этой части показано среднее, максимальное и минимальное значение добавочной стоимости сущностей по каждому типу сущностей. Добавочная стоимость рассчитывается на основании *VA Time*.

Wait Cost: в этой части показано среднее, максимальное и минимальное значение стоимости ожидания сущностей по каждому типу сущностей. Стоимость ожидания подсчитывается исходя из



времени ожидания, стоимости ресурса и стоимостью нахождения сущности в системе.

Wait Time: в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей. Время ожидания - это период времени с момента поступления сущности в очередь (либо в модуле Process ожидает ресурс, либо в модуле Batch ожидает группировки) и до момента выхода из нее (начнет обрабатываться, либо будет сгруппирована).

WIP: в этой части показано среднее, максимальное и минимальное значение времени ожидания сущностей по каждому типу сущностей.

Раздел «Ресурсы» содержит информацию о ресурсах. Раздел «Очереди» содержит информацию о очередях.

## **1.4 ПАНЕЛЬ НАВИГАЦИИ**

С помощью панели навигации можно быстро передвигаться по различным уровням модели, быстро менять виды. Можно задать быстрые клавиши для изменения вида. Виды подмоделей создаются автоматически, но также возможно добавить новые виды с помощью команды Add View. Можно передвигаться не только по различным уровням модели, но также быстро получать нужный масштаб какой-либо части модели.

## **1.5 ПОСТРОИТЕЛЬ ВЫРАЖЕНИЙ**

ПП Arena позволяет строить сложные выражения. Это достигается с помощью Expression Builder. Построитель выражений имеет внешний вид, показанный на рис. 10.

Построитель выражений имеет 2 секции:

1 - окно возможных выражений элементов, организованных по типам;

2 - окно выражений.

Также имеются кнопки операторов (сложение, вычитание, и т. элементы сравнения, логические операторы и т.д.)

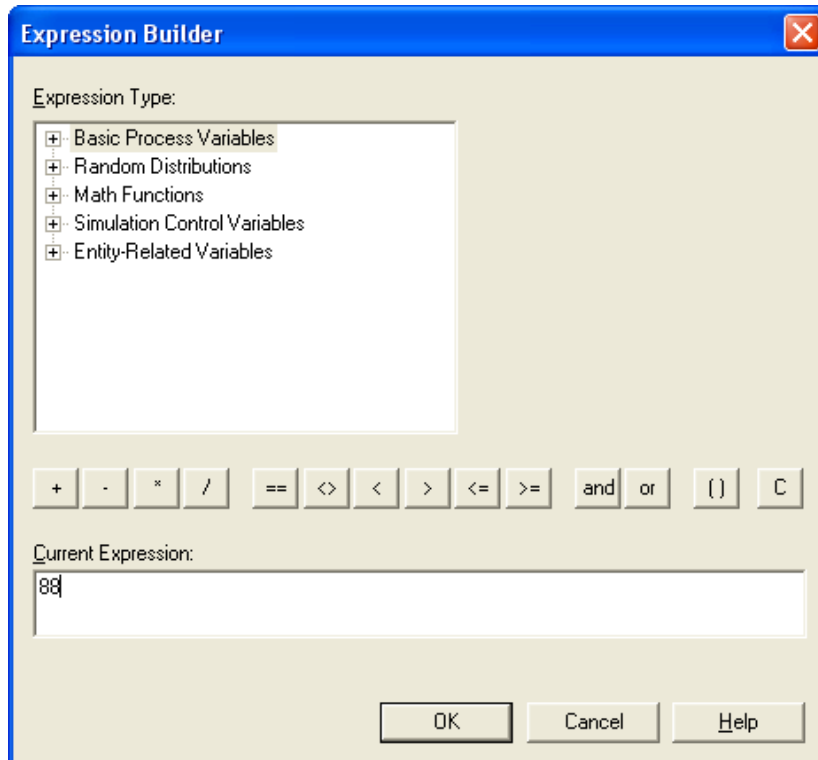


Рисунок 10 – Внешний вид построителя выражений

## 1.6 ФУНКЦИИ ВОСПРОИЗВОДЯЩИЕ СЛУЧАЙНЫЕ СОБЫТИЯ И ИХ ПАРАМЕТРЫ

Эти функции предназначены для задания интервалов времени между появлениями случайных событий происходящих в различных графических модулях. Так, например в модуле Create они позволяют задать интервал времени между поставками экземпляров сущности в систему, а в модуле Process задать интервал времени задержки сущности. Используемые функции представлены в табл. 15.

Таблица 15

Обозначение функции	Описание
1 EXPO(Mean)	Экспоненциальное распределение (показательное распределение) с параметром Mean – среднее значение интервала времени. С помощью этой функции воспроизводятся простейшие потоки событий
2 NORM(Mean, StdDev)	Нормальное распределение с параметрами: Mean – среднее значение интервала времени; StdDev – среднее квадратическое отклонение.
3 TRIA(Min, Mode, Max)	Треугольное распределение (распределение Симпсона) для которого задаются: Min – минимальное значение интервала времени;

Обозначение функции	Описание
	Mode – наиболее вероятное значение временного интервала; Max – максимальное значение интервала времени.
4 UNIF(Min , Max)	Равномерное распределение с параметрами: Min – минимальное значение интервала времени; Max - максимальное значение интервала времени.
5 LOGN(LogMean, LogStd)	Логонормальное распределение с параметрами: Mean – среднее значение интервала времени; StdDev – среднее квадратическое отклонение.
6 POIS(Mean)	Распределение Пуассона с параметром Mean – средним значением интервала времени.

## 2 ПРИМЕР МОДЕЛИРОВАНИЯ ОДНОКОНАЛЬНОЙ СИСТЕМЫ МАССОВОГО ОБСЛУЖИВАНИЯ С ОТКАЗАМИ

### 2.1 Построение модели

Технологическая система состоит из одного станка. На станок поступают заявки на изготовление деталей в среднем через 0,5 часа ( $\bar{t}_3=0,5ч.$ ). Среднее время изготовления одной детали равно  $\bar{t}_{об}=0,6ч.$  Если при поступлении заявки на изготовление детали станок занят, то она (деталь) направляется на другой станок. Найти абсолютную и относительную пропускную способности системы и вероятность отказа по изготовлению детали.

Модель системы строим в следующей последовательности:

1) Запускаем программу Arena (рис. 11).

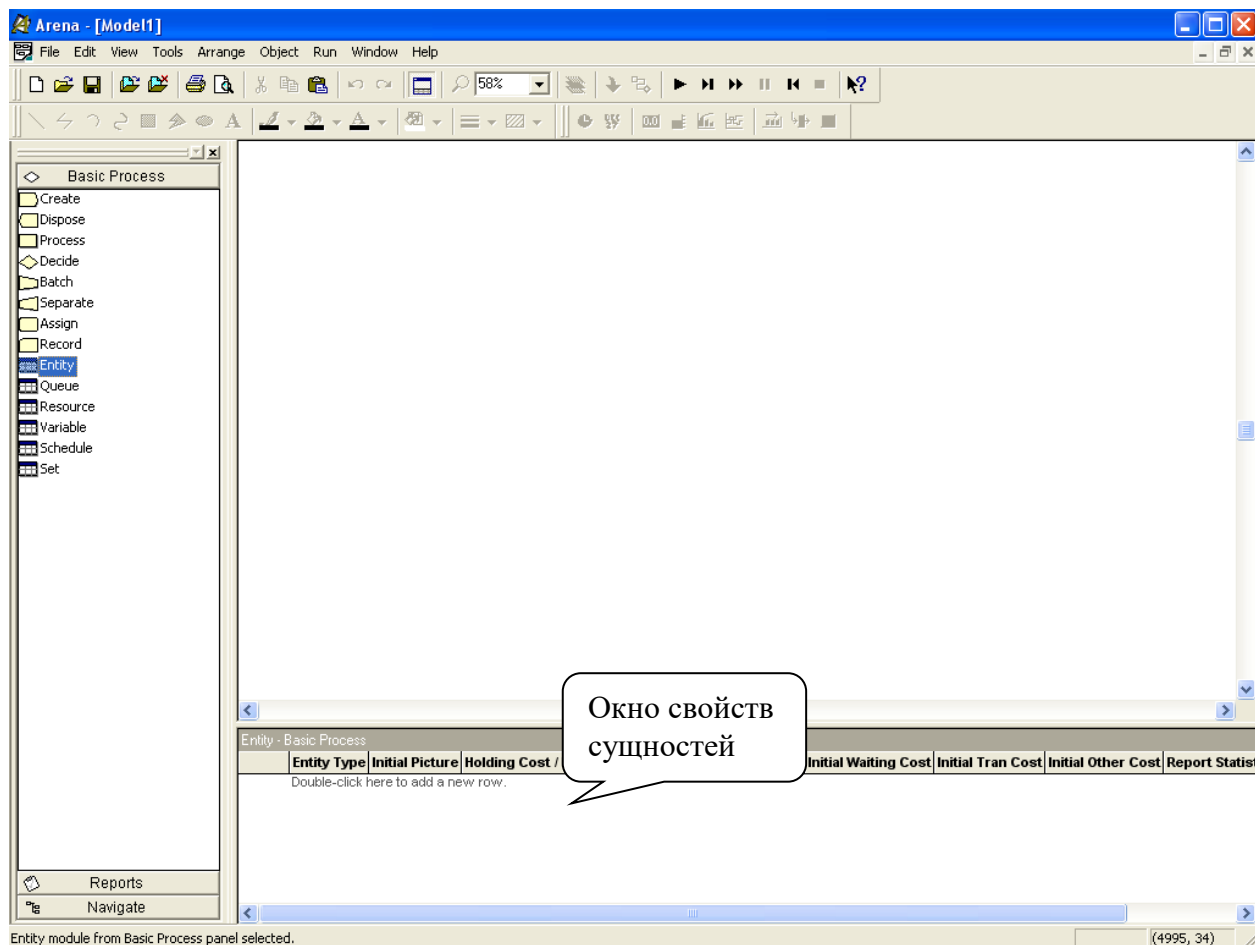


Рисунок 11 – Начало построения модели

2) В модуле данных «Create» создаем сущность «Detal» формирующую поток заявок, в столбце Entity Type задаем имя сущности (рис. 12).

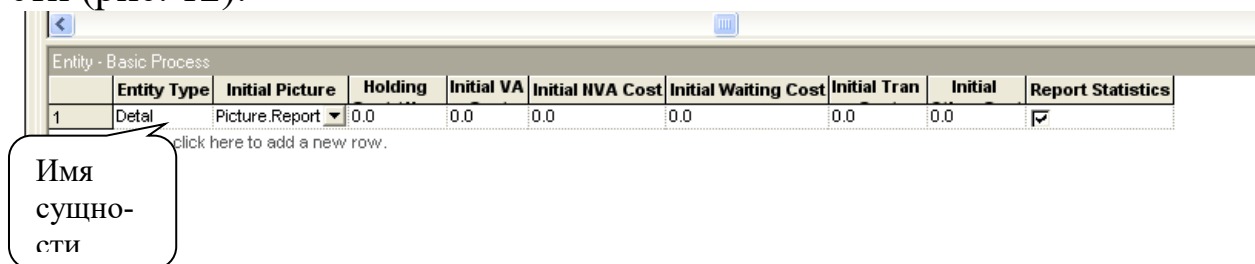


Рисунок 12 – Создание сущности «Detal»

3) В модуле данных «Resource» создаем ресурс или канал обслуживания «Stanok» обрабатывающего детали (рис. 13).

Resource - Basic Process									
	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	Stanok	Fixed Capacity	1	0.0	0.0	0.0		0 rows	<input checked="" type="checkbox"/>

Double-click here to add a new row.

Рисунок 13 – Создание ресурса «Stanok»

4) В модуле данных «Variable» создадим переменные (рис.14):  
**AA – Абсолютная пропускная способность** (среднее число заявок, обслуживаемых в единицу времени);

**QQ – Относительная пропускная способность** (средняя доля заявок, обслуживаемых системой);

**Potk – Вероятность отказа** (вероятность того, что заявка покинет СМО необслуженной).

Variable - Basic Process						
	Name	Rows	Columns	Clear Option	Initial Values	Report Statistics
1	AA			System	0 rows	<input checked="" type="checkbox"/>
2	QQ			System	0 rows	<input checked="" type="checkbox"/>
3	Potk			System	0 rows	<input checked="" type="checkbox"/>

Double-click here to add a new row.

Рисунок 14 – Задание оценочных переменных модели

5) Добавим в модель модуль «Create» и назовем его «Postavka» (рис. 15).

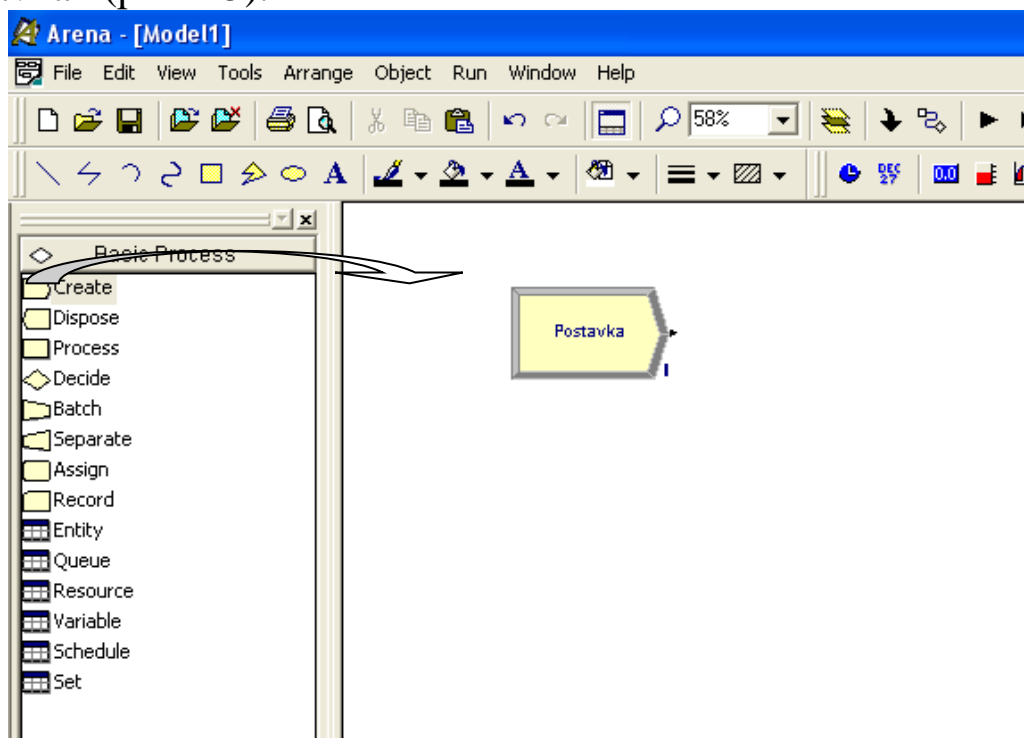


Рисунок 15 – Создание модуля Pstavka

Заполняем свойства модуля (рис. 16). Т.к. предполагается, что рассматриваемый процесс простейший, то в столбце Expression зададим экспоненциальный закон распределения интервалов времени между поставками сущностей в модель со средним временем 0.5 ч.

	Name	Entity Type	Type	Expression	Units	Entities per Arrival	Max Arrivals	First Creation
1	Postavka	Detal	Expression	EXPO( 0.5 )	Hours	1	1000	0.0

Имя модуля

Имя создаваемой сущности

Единица измерения времени

Максимальное число сущностей создаваемое модулем

Рисунок 16 – Задание свойств модуля Postavka

б) Добавим в модель модуль «Process» и назовем его «Obrabotka» (рис. 17).

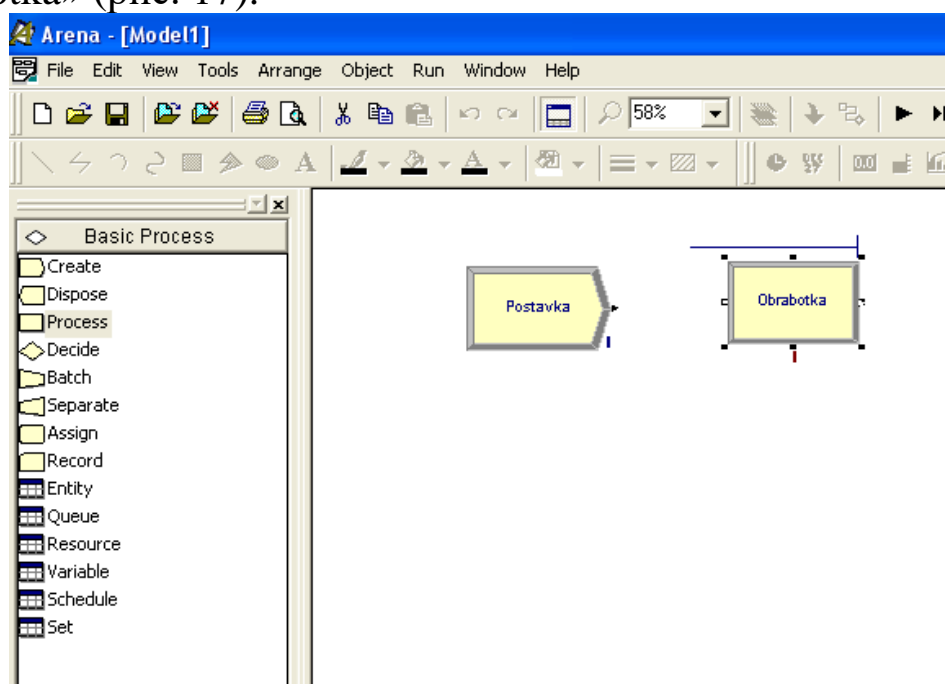


Рисунок 17 – Создание модуля Obrabotka

Для задания свойств процесса наведем курсор мыши на соответствующую строку и нажмем правую клавишу (рис. 18). В появившемся меню выбираем «**Edit via Dialog...**» и в открывшемся окне «Process» устанавливаем необходимые свойства (рис. 19). В поле «Expressions» установим экспоненциальный закон распреде-

ления интервалов времени обработки детали со средним временем 0.6 ч.

	Type	Action	Priority	Resources	Delay Type	Units	Allocation	Expression	Report
1	Standard	Seize Delay Release	Medium(2)	1 rows	Expression	Hours		EXPO( 0.6 )	<input checked="" type="checkbox"/>

Рисунок 18 – Задние свойств в модуле Obrabotka

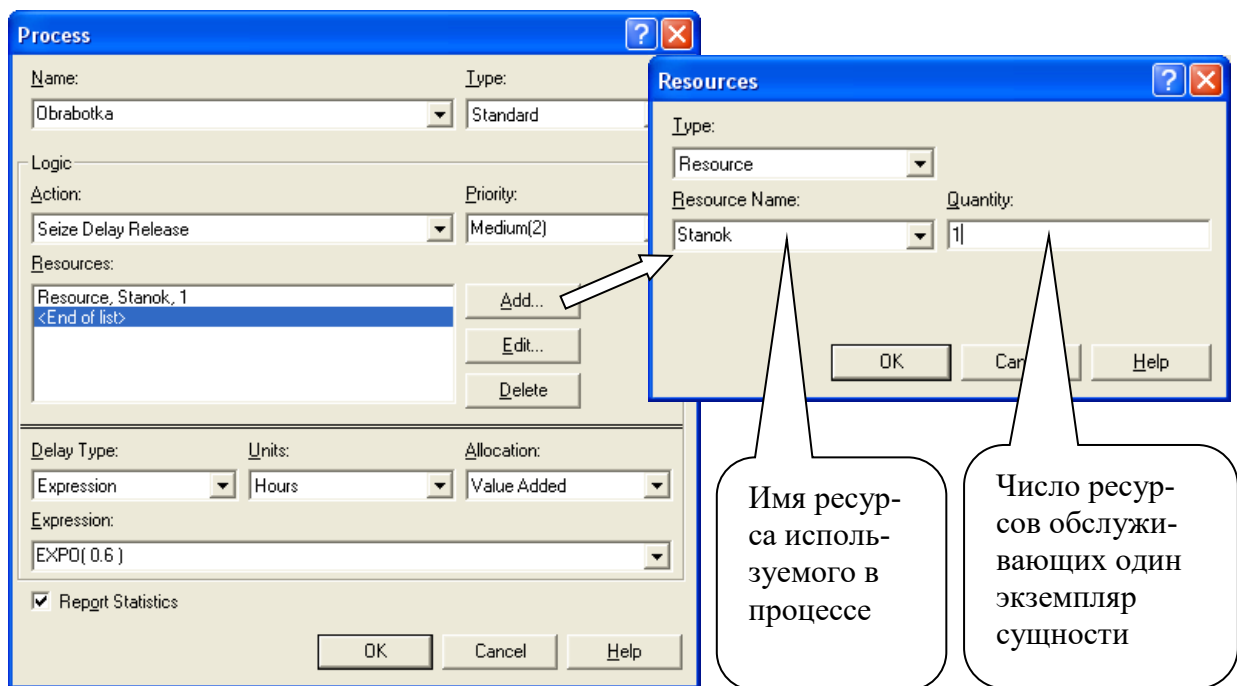


Рисунок 19 – Задание свойств в окне «Process»

7) Для анализа загрузки станка в момент прихода новой детали добавляем в модель модуль «Decide». Задаем ему имя «Analis stanka» (рис. 20) и устанавливаем остальные свойства (рис. 21). Для заполнения свойств модуля можно воспользоваться окном «Decide» (рис. 22), которое появляется после двойного щелчка указателем мыши по графическому элементу модуля (рис. 20).

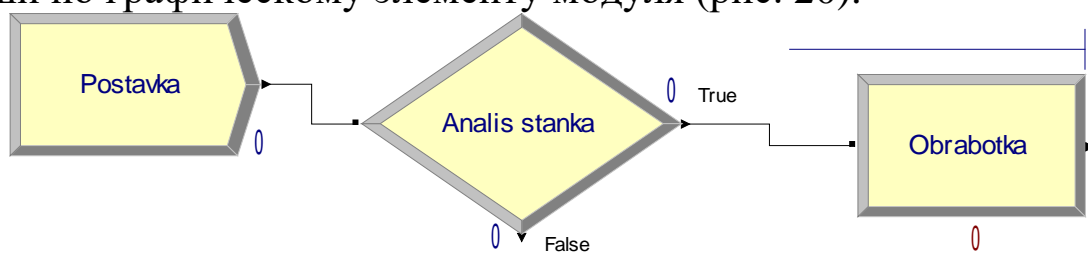


Рисунок 20 – Добавление модуля «Analis stanka»

	Name	Type	If	Variable Name	Is	Value
1	Analys stanka	2-way by Condition	Variable	Obrabotka.NumberIn	==	Obrabotka.Numberout

Рисунок 21 – Свойства модуля «Analys stanka»

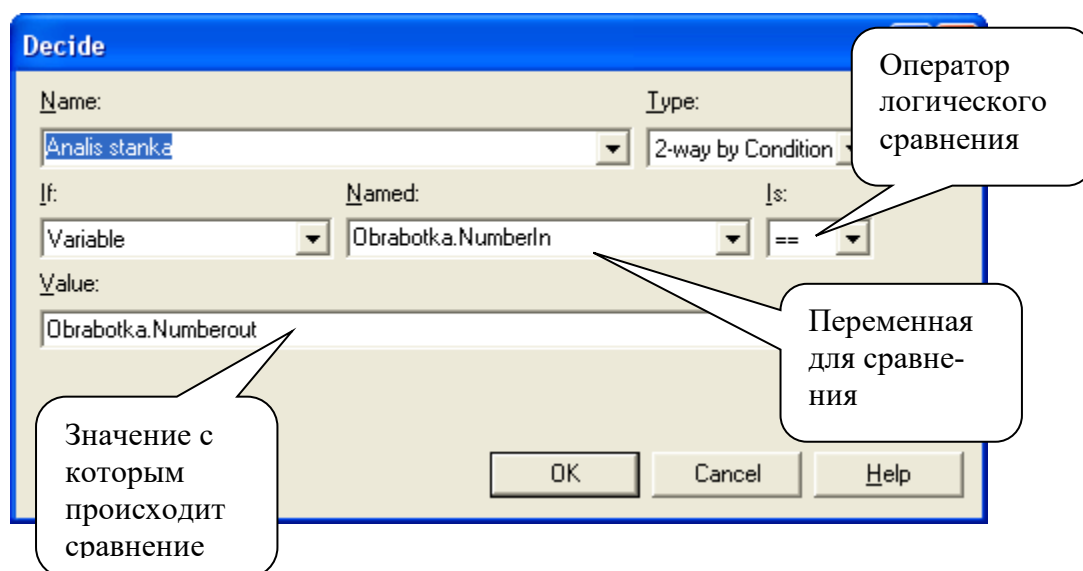


Рисунок 22 – Окно свойств модуля «Analys stanka»

Суть данного модуля заключается в том, что в момент времени, когда сущность «Detal» попадает в модуль происходит анализ состояния процесса «Obrabotka», т.е. сопоставляется число деталей поступивших (Obrabotka.NumberIn) на станок и покинувших станок (Obrabotka.NumberOut). Если эти числа равны, значит станок свободен и деталь направляется на обработку (выход True), в противном случае это означает, что станок занят и деталь покидает систему (выход False).

8) Добавляем в модель модуль «Assign» и задаем ему имя «Оценка processa» (рис. 23). В этом модуле мы задаем формулы для расчета оценочных параметров модели – AA, QQ, Pokl. Эти параметры мы рассчитаем следующим образом:

**Абсолютную пропускную способность** (среднее число деталей, обслуживаемых в единицу времени) мы будем определять, как отношение числа обработанных деталей к текущему системному времени модели.



**Относительная пропускная способность** (средняя доля деталей, обслуживаемых системой) мы будем определять, как отношение числа обработанных деталей к общему числу деталей прошедших через систему.

**Вероятность отказа** (вероятность того, что деталь покинет СМО необслуженной) мы определим, как  $1-QQ$ .

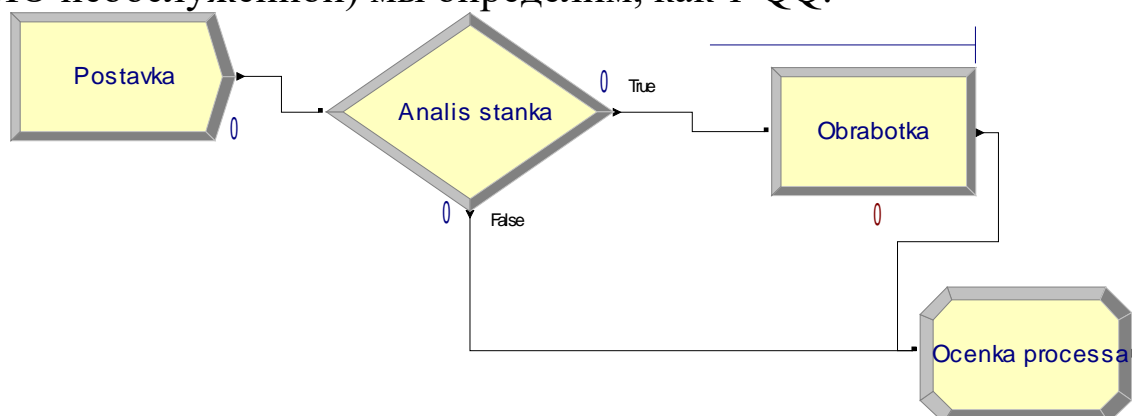


Рисунок 23 – Создание модуля «Оценка processa»

Для задания расчетных формул оценочных параметров наведем курсор на модуль «Оценка processa» и дважды щёлкаем левой клавишей мыши. В открывшемся окне «Assign» (рис. 24) добавляем расчетные формулы нажимая кнопку «Add...»

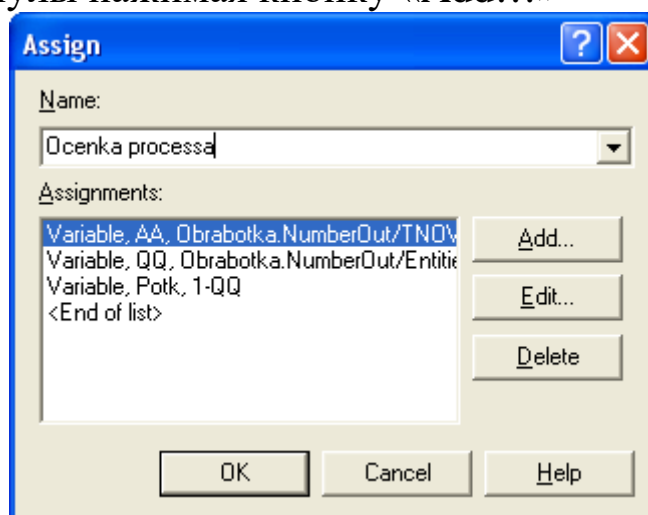


Рисунок 24 – Окно задания расчетных формул переменных

В окне «Assignments» задаем имя переменной (рис. 25) и её новое значение (выражение для расчета). Проектирование выражений для расчета переменных удобно выполнять с помощью постро-

ителя выражений, для его вызова необходимо навести курсор на поле «New Value:», нажать правую клавишу мыши и в появившемся меню выбрать «Build Expression...» (рис.25).

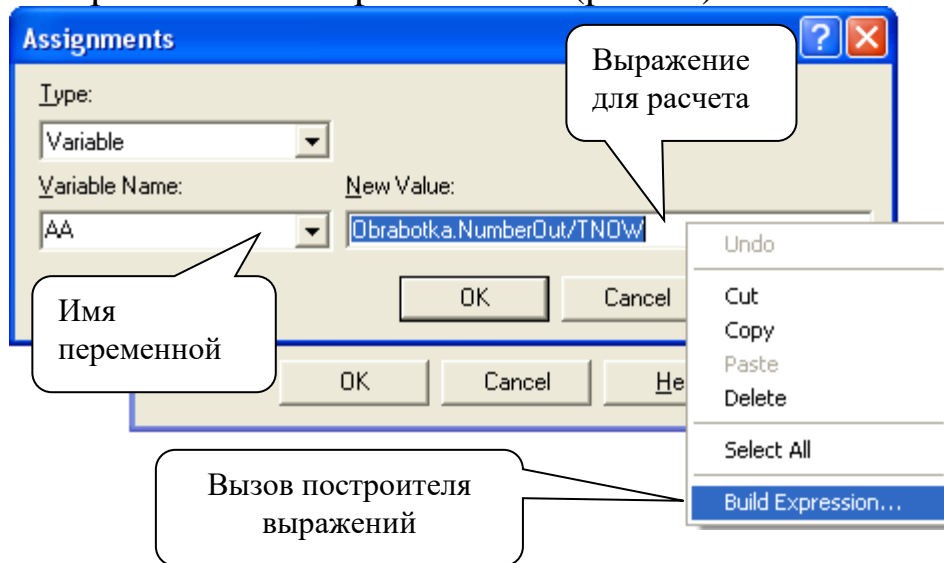


Рисунок 25 – Работа с окном «Assignments»

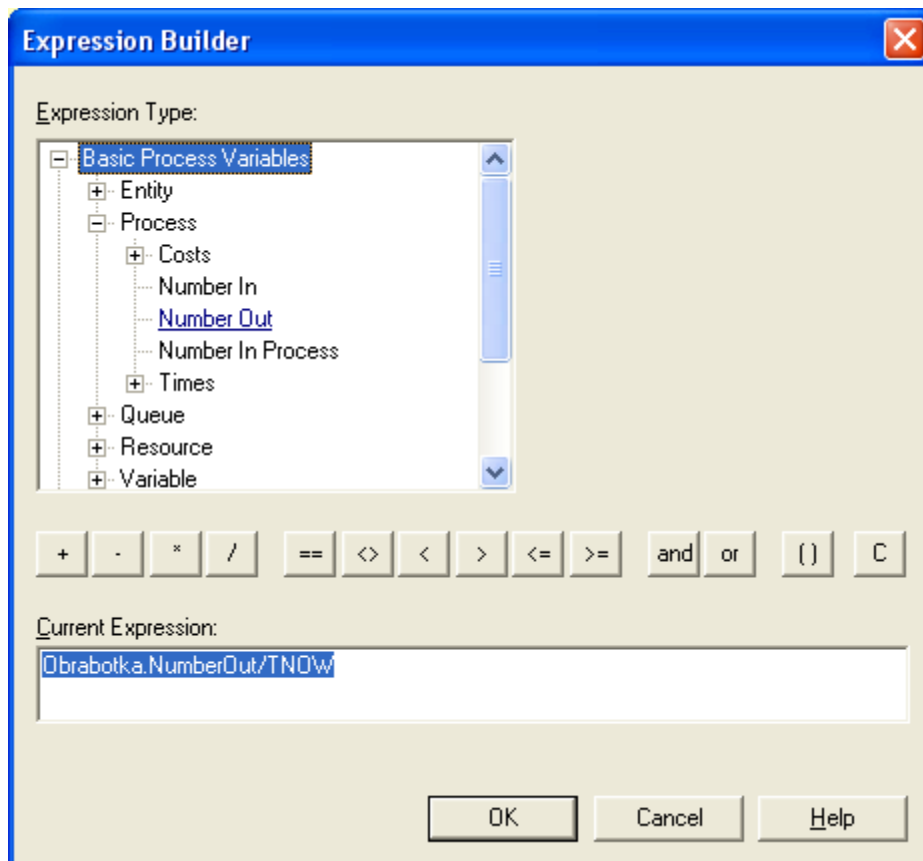


Рисунок 26 – Создание выражения в построителе выражений

При создании выражения для расчета переменных в построителе выражений необходимо:

- для переменной AA выполнить:

1 В поле «Expression Type» последовательно выбрать «Basic Process Variables/Process/ Number Out ». В поле «Current Expression» должна появиться надпись «Obrabotka.NumberOut» (рис. 26);

2 Нажать кнопку «/»;

3 В поле «Expression Type» последовательно выбрать «Simulation Control Variables/ Current Simulation Time» (рис. 27). В поле «Current Expression» должна появиться надпись «Obrabotka.NumberOut/TNOW» (рис. 26);

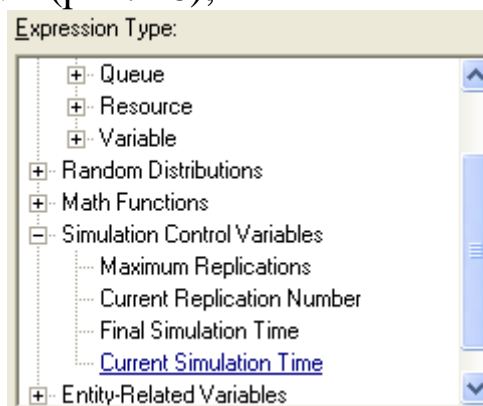


Рисунок 27 – Выбор системной переменной «Текущее время симуляции»

- для переменной QQ выполнить:

1 В поле «Expression Type» последовательно выбрать «Basic Process Variables/Process/ Number Out ». В поле «Current Expression» должна появиться надпись «Obrabotka.NumberOut» (рис. 26);

2 Нажать кнопку «/»;

3 В поле «Expression Type» последовательно выбрать « Basic Process Variables/ Entity/Number In» (рис. 28). В поле «Current Expression» должна появиться надпись «Obrabotka.NumberOut/EntitiesIn(Detal)».

Выражение для переменной Potk можно задать без использования построителя выражений в окне «Assignments» (рис. 25). В поле «New Value:» записываем выражение «1-QQ».

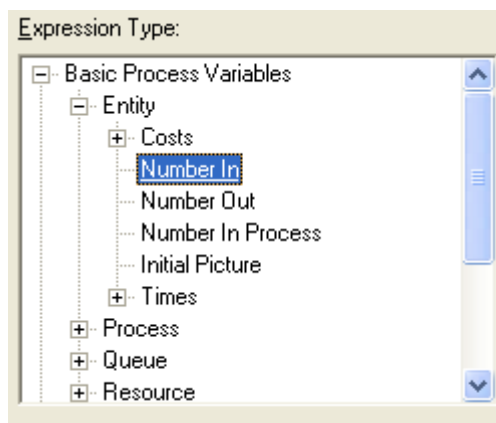


Рисунок 28 – Выбор системной переменной «Число вошедших сущностей»

9) Добавляем в модель модуль «Dispose» и устанавливаем ему имя «Vyhod» (рис. 29)

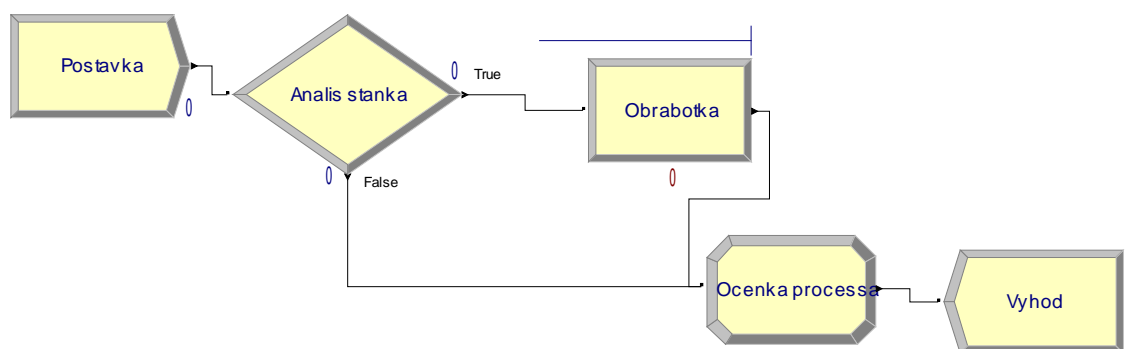




Рисунок 29 – Создание модуля «Vyhod»

10) Создаем в модели поля для динамического отображения значения оценочных параметров модели. Для отображения переменной «AA» необходимо на панели инструментов нажать кнопку  и в окне «Variable» установить настройки отображения (рис. 30) и указать мышью положение поля (рис. 31). Аналогично добавляются поля для остальных переменных.

11) Добавляем в модель индикатор текущего времени симуляции, для этого на панели инструментов нажимаем кнопку , устанавливаем настройки отображения времени (рис. 32) и указываем мышью положение индикатора (рис. 33).

12) Модель готова к расчетам.

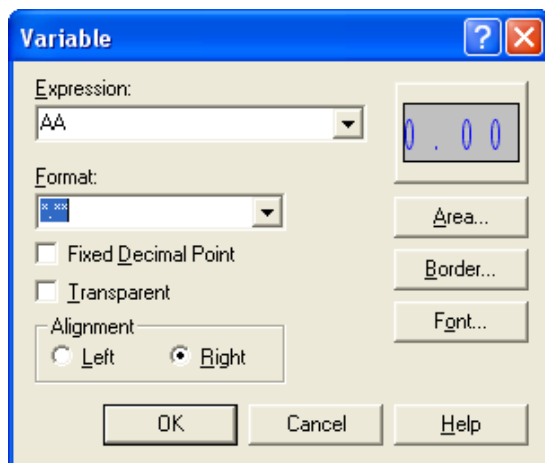


Рисунок 30 – Настройка отображения переменной «AA»

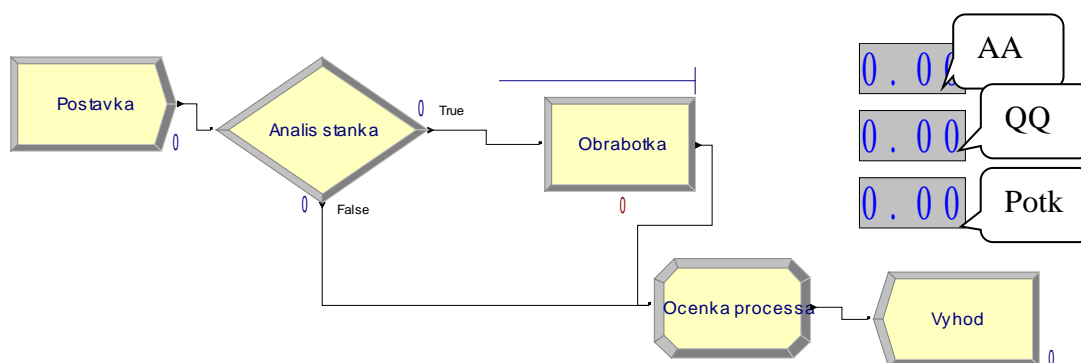


Рисунок 31 – Модель с полями для отображения значений оценочных параметров

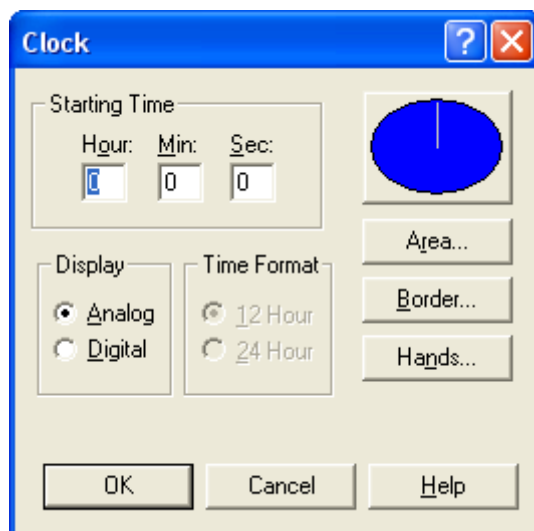


Рисунок 32 – Настройка отображения времени

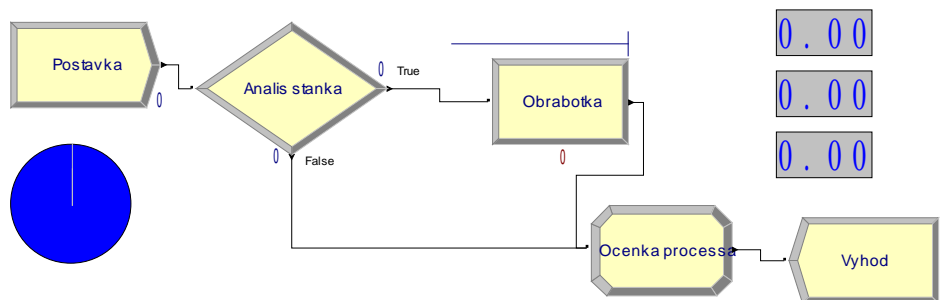





Рисунок 33 – Модель с индикатором времени


## 2.2 Моделирования работы системы


Моделирование работы системы можно выполнять в следующих режимах:

- в режиме моделирования с анимацией перемещения сущностей (на панели инструментов необходимо нажать кнопку 

- в режиме пошагового выполнения моделирования с анимацией перемещения сущностей (на панели инструментов необходимо нажать кнопку 

- в режиме моделирования без анимацией перемещения сущностей (на панели инструментов необходимо нажать кнопку 

Для установки паузы в процессе моделирования на панели инструментов необходимо нажать кнопку .

И так, для выполнения моделирования нажимаем кнопку . Моделирование занимает некоторое время. После покидания системы 1000 деталей система заканчивает процесс моделирования, а в полях оценочных параметров будут записаны их последние значения (рис. 34).

На основании полученных результатов можно сделать вывод, что: абсолютную пропускную способность равна 0,91 детали в час; относительная пропускная способность равна 46%; вероятность отказа равна 54%.

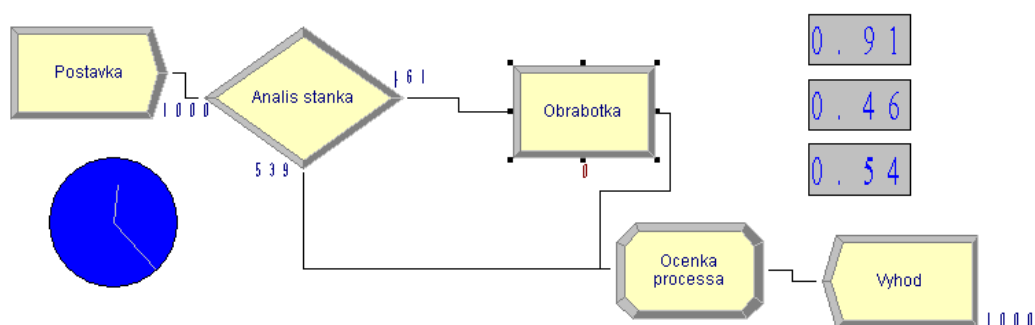


Рисунок 34 – Вид модели после завершения моделирования

### 3 ЗАДАНИЯ

Исходные данные для выполнения лабораторной работы представлены в таблице 16.

**Примечание: При использовании пуассоновского закона распределения среднее время желательно задавать в секундах**

Таблица 16

№ Варианта	№ зад.	Интенсивность потока	Средняя продолжительность обслуживания	Распределение потока заявок	Распределение потока обслуживания
1.	5	6 обращений в час	10 минут	Пуассона	Экспоненциальное
2.	6	7 обращений в час	8 минут	Пуассона	Пуассона
3.	1	0,95 вызова в минуту	1 минута	Экспоненциальное	Экспоненциальное
4.	2	10 обращений в час	6 минут	Экспоненциальное	Пуассона
5.	3	0,5 автомобиля в час	1,2 часа	Пуассона	Пуассона
6.	4	2 вызов в минуту	20 секунд	Экспоненциальное	Пуассона
7.	5	8 обращений в час	8 минут	Пуассона	Экспоненциальное
8.	6	8 обращений в час	7 минут	Экспоненциальное	Пуассона
9.	1	1,1 вызова в минуту	50 секунд	Пуассона	Экспоненциальное
10.	2	10 обращений в час	6 минут	Экспоненциальное	Пуассона
11.	3	1 автомобиль в час	0,9 часа	Экспоненциальное	Пуассона

<b>№ Варианта</b>	<b>№ зад.</b>	<b>Интенсивность потока</b>	<b>Средняя продолжительность обслуживания</b>	<b>Распределение потока заявок</b>	<b>Распределение потока обслуживания</b>
12.	4	3 обращения в час	19 минут	Пуассона	Экспоненциальное
13.	5	9 обращений в час	6 минут	Экспоненциальное	Пуассона
14.	6	9 обращений в час	7 минут	Пуассона	Пуассона
15.	1	1,5 вызова в минуту	40 секунд	Экспоненциальное	Пуассона
16.	2	7 обращений в час	8 минут	Пуассона	Экспоненциальное
17.	3	0,7 автомобиля в час	1,2 часа	Экспоненциальное	Пуассона
18.	4	6 обращений в час	10 минут	Пуассона	Экспоненциальное
19.	5	10 обращений в час	6 минут	Экспоненциальное	Пуассона
20.	6	3 обращения в час	19 минут	Пуассона	Пуассона
21.	1	0,8 вызова в минуту	74 секунды	Экспоненциальное	Пуассона
22.	2	9 обращений в час	7 минут	Пуассона	Экспоненциальное
23.	3	0,9 автомобиля в час	1 час	Экспоненциальное	Пуассона
24.	4	9 обращений в час	7 минут	Экспоненциальное	Пуассона
25.	5	2 обращения в час	25 минут	Пуассона	Пуассона
26.	6	4 обращения в час	13 минут	Экспоненциальное	Экспоненциальное
27.	1	2 вызов в минуту	20 секунд	Пуассона	Экспоненциальное
28.	2	6 обращений в час	10 минут	Экспоненциальное	Пуассона
29.	3	0,6 автомобиля в час	1,2 часа	Пуассона	Пуассона
30.	4	7 обращений в час	8 минут	Пуассона	Экспоненциальное



## Задачи

1. Одноканальная СМО с отказами представляет собой одну телефонную линию. Заявка (вызов), пришедшая в момент, когда линия занята, получает отказ. Определите характеристики СМО в установившемся режиме работы.

2. В одноканальную СМО с отказами поступает поток заявок. Определите характеристики СМО в установившемся режиме работы.

3. Пост диагностики автомобилей представляет собой одноканальную СМО с отказами. Заявка на диагностику, поступившая в момент, когда пост занят, получает отказ. Определите в установившемся режиме характеристики системы.

4. Одноканальная СМО представляет собой электронное справочное окно на железнодорожном вокзале, в котором все желающие могут узнать информацию о прибытии поездов и др. Если при обращении клиента окно занято, то клиент покидает систему и обращается к другому окну. Определите в установившемся режиме характеристики работы электронного справочного окна.

5. Одноканальная СМО представляет собой больницу, в которой больные пытаются попасть на прием к одному врачу. Если при обращении больного врач занят, то больной покидает систему и обращается к другому врачу. Определите в установившемся режиме характеристики работы врача.

6. Одноканальная СМО представляет собой юридическую консультацию в которой клиенты пытаются получить консультацию у юриста. Если при обращении клиента юрист занят, то клиент покидает систему и обращается к другому юристу. Определите в установившемся режиме характеристики работы юриста.

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Железимова  
« 14 » \_\_\_\_\_ 20\_\_ г.



**РАСЧЕТ ФИНАЛЬНЫХ ВЕРОЯТНОСТЕЙ СМО  
С ПОМОЩЬЮ УРАВНЕНИЙ КОЛМОГорова**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

УДК 519.6

Составители: В.В. Куц

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Расчет финальных вероятностей СМО с помощью уравнений Колмогорова:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 9 с.:– Библиогр.: с. 19.

Содержат сведения о составлении и решении уравнений Колмогорова для расчета финальных вероятностей СМО. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 0,52 . Уч.-изд. л. 0,47.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

**1. Цель работы:** научиться определять финальные вероятности систем массового обслуживания с помощью уравнений Колмогорова.

**2. Подготовка к работе:**

- 2.1. Изучить материал и конспект лекций по теме.
- 2.2. Изучить описание практической работы.
- 2.3. Подготовить бланк отчета, содержащий название, цель работы, задание, результаты выполнения п.п.4.1..4.2.

**3. Оборудование:** Персональный компьютер, программа Maple.

**4. Задание:**

Занятия в лаборатории разрешается проводить только в присутствии преподавателя.

- 4.1. Из рисунков Приложения согласно варианту, заданному преподавателем. нарисовать граф состояний.
- 4.2. Составить уравнения Колмогорова для финальных вероятностей заданного графа.
- 4.3. С помощью программы Maple найти вероятности состояний системы.
- 4.4. Сделать вывод об относительном времени пребывания системы в каждом из состояний.

**5. Порядок выполнения работы:**

5.1. Найдем финальные вероятности СМО: система  $S$  состоит из 2-х автоматов, каждый из которых в случайный момент времени может быть либо занятым, либо свободным. Одновременно включить или выключить оба автомата невозможно.

Для построения графа состояний системы  $S$  примем, что устройство  $S$  может находиться в одном из 4-х состояний:  $S_0$  - оба автомата исправны.  $S_1$  - первый автомат ремонтируется, второй исправен.  $S_2$  - второй автомат ремонтируется . первый исправен .  $S_3$  - оба автомата ремонтируются.

Граф системы приведен на рисунке:

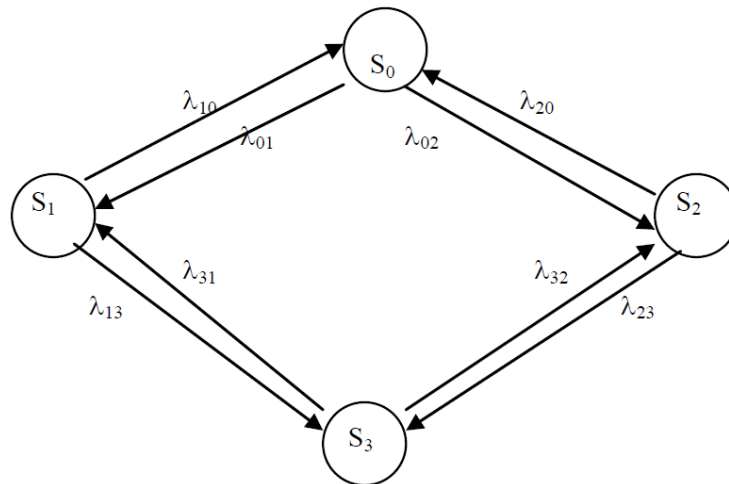


Рис. 1.

5.2. Запишем уравнения Колмогорова для заданного графа:  
 Предельные вероятности системы  $S$  находим, пользуясь системой уравнений Колмогорова:

$$\begin{cases} (\lambda_{01} + \lambda_{02}) p_0 = \lambda_{10} p_1 + \lambda_{20} p_2 \\ (\lambda_{10} + \lambda_{13}) p_1 = \lambda_{01} p_0 + \lambda_{31} p_3 \\ (\lambda_{20} + \lambda_{23}) p_2 = \lambda_{02} p_0 + \lambda_{32} p_3 \\ (\lambda_{31} + \lambda_{32}) p_3 = \lambda_{13} p_1 + \lambda_{23} p_2 \end{cases}$$

Например, при  $\lambda_{01} = 1, \lambda_{02} = 2, \lambda_{10} = 2, \lambda_{13} = 2, \lambda_{20} = 3, \lambda_{23} = 1, \lambda_{31} = 3, \lambda_{32} = 2$  система алгебраических уравнений, описывающих стационарный режим работы для данной системы, имеет вид:

$$\begin{cases} 3 p_0 = 2 p_1 + 3 p_2 \\ 4 p_1 = p_0 + 3 p_3 \\ 4 p_2 = 2 p_0 + 2 p_3 \\ p_0 + p_1 + p_2 + p_3 = 1 \end{cases}$$

5.3. Запишем выражение в программе Maple для нахождения решения системы уравнений:

```
>sols:=solve ({ , , p0+p1+p2+p3=1},{p0,p1,p2,p3 }); Enter
```

5.4. Сделаем вывод об относительном времени пребывания системы в каждом из состояний.

Решив систему, получим  $p_0 = 0.40$  ,  $p_1 = 0.20$  ,  $p_3 = 0.13$  . т.е. в предельном стационарном режиме система  $S$  в среднем 40% времени будет находиться в состоянии  $S_0$  (оба автомата исправны). 20% - в состоянии  $S_1$  (первый автомат ремонтируется, второй работает). 27% - в состоянии  $S_2$  (второй автомат ремонтируется, первый работает) и 13% времени – в состоянии  $S_3$  (оба автомата ремонтируются).

## **6. Содержание отчета:**

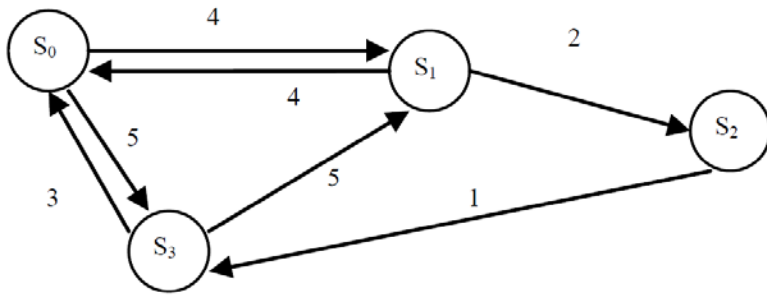
Отчет должен содержать результаты выполнения п.п. 4.1. - 4.4.

## **7. Контрольные вопросы:**

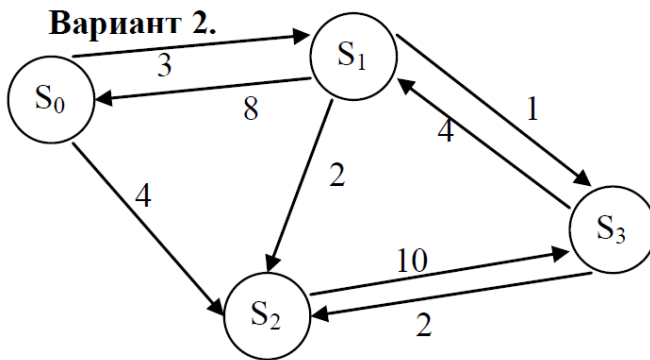
1. Что такое СМО?
2. Приведите классификацию СМО.
3. Приведите примеры параметров СМО.
4. Что такое поток?
5. Какой поток называют простейшим?
6. Какой процесс называют Марковским?
7. Что такое процесс с дискретными состояниями.
8. Что такое процесс с непрерывным временем?
9. Что означает условие отсутствия последствия?
10. Что такое граф состояний?

Приложение.

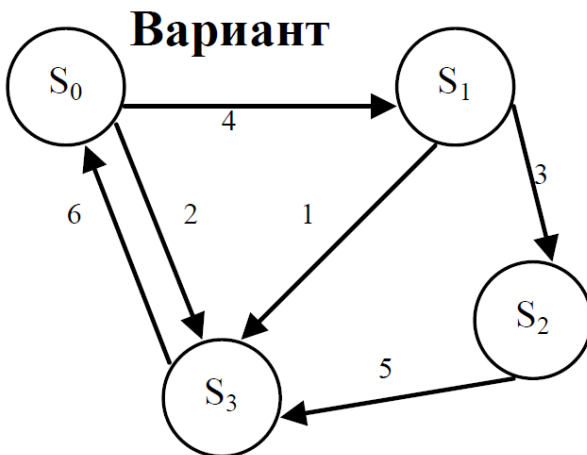
Вариант 1.



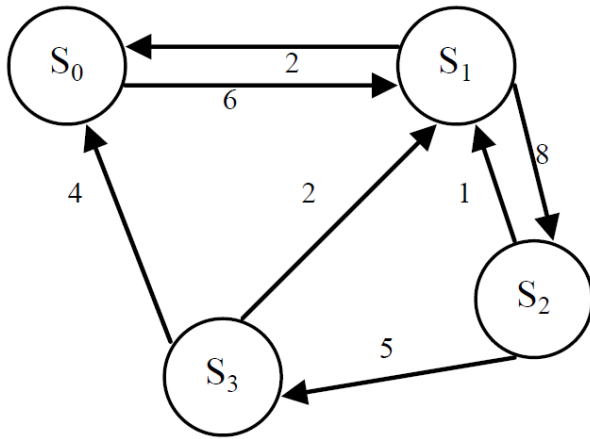
Вариант 2.



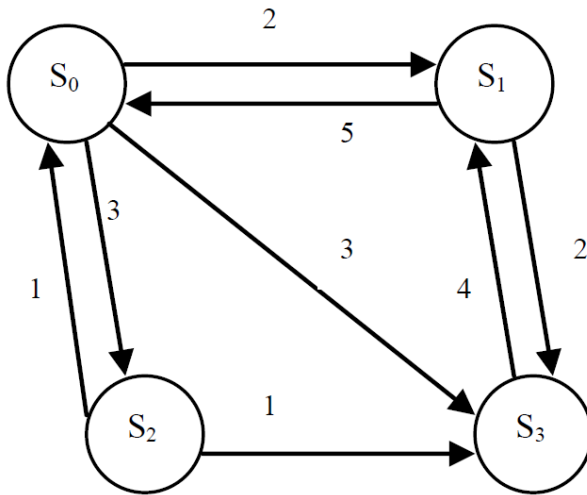
Вариант 3.



Вариант 4.

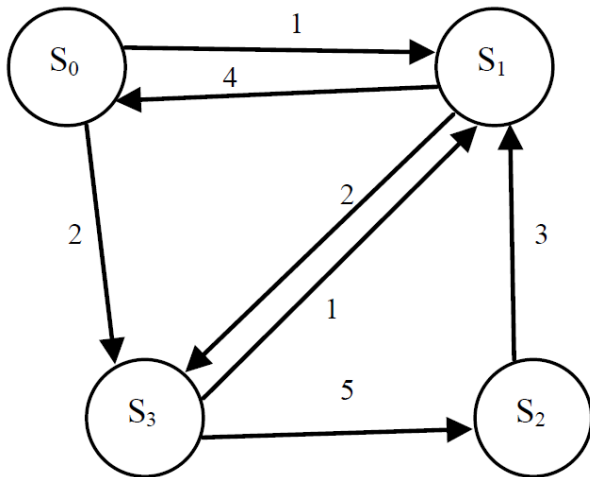


Вариант 5.

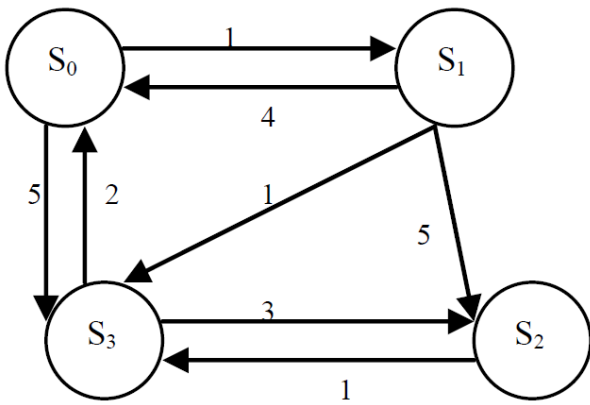


Вариант 6.

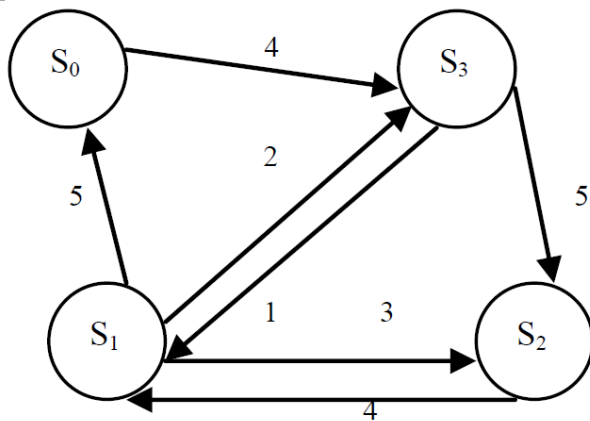




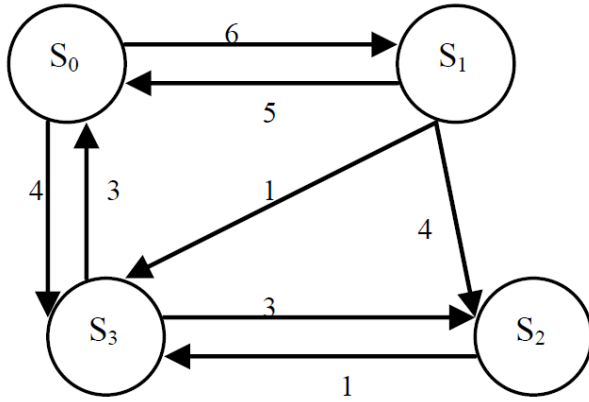
Вариант 7.



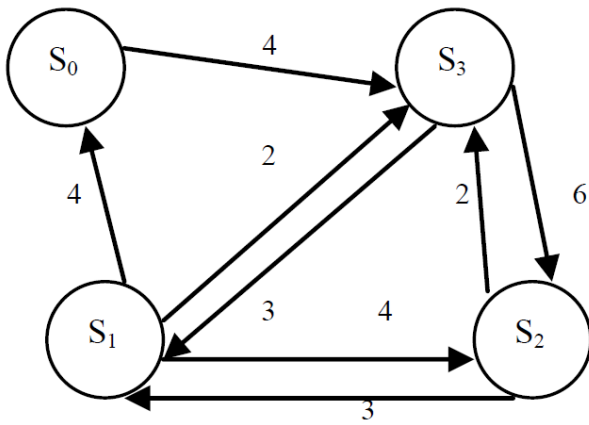
Вариант 8.



Вариант 9.



Вариант 10.



**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Мокшанова  
« 14 » \_\_\_\_\_ г.



**РЕШЕНИЕ ЗАДАЧ КРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ  
В ПРОГРАММЕ MICROSOFT EXCEL**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

УДК 519.6

Составители: В.В. Куц

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Решение задач критериальной оптимизации в программе Microsoft Excel:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 26 с.

Содержат сведения о решении задач критериальной оптимизации в программе Microsoft Excel. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 1,51 . Уч.-изд. л. 1,37.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

## 1. Задание

1.1. Согласно своему варианту построить оптимизационную модель для решения поставленной задачи, а именно: обозначить неизвестные переменные; построить целевую функцию с указанием направления экстремума; построить систему ограничений);

1.2. Решить задачу оптимизации в программе Microsoft Excel

1.3. Оформить отчет

## 2. Пример решения задачи оптимизации

Предположим, что некоему студенту необходимо сдать два зачета в один день. Он поставил себе задачу сдать их как можно лучше, причем так, чтобы оценка за каждый зачет была не меньше 2,5 баллов. У него осталось 24 часа. Студент предположил, что, потратив один час на первый предмет, он повысит возможную оценку на 0,5 балла, потратив тот же час на второй предмет, - повысит оценку по нему на 0,25 балла. Необходимо определить, сколько часов (x) ему надо потратить на первый предмет и сколько (y) на второй предмет, чтобы выполнить поставленную задачу. Математически задача запишется так:

$$0,5x + 0,25y \rightarrow \max \text{ (сумма оценок по предметам),}$$

и должны выполняться следующие ограничения:

$$2,5 \leq 0,5x \leq 5;$$

$$2,5 \leq 0,25y \leq 5;$$

$$x + y \leq 24.$$

### Порядок решения задачи оптимизации в Microsoft Excel

1. Для решения данной запустим MS Excel.
2. Отведем под неизвестные x и y диапазон ячеек B2:C2 (рис.1).

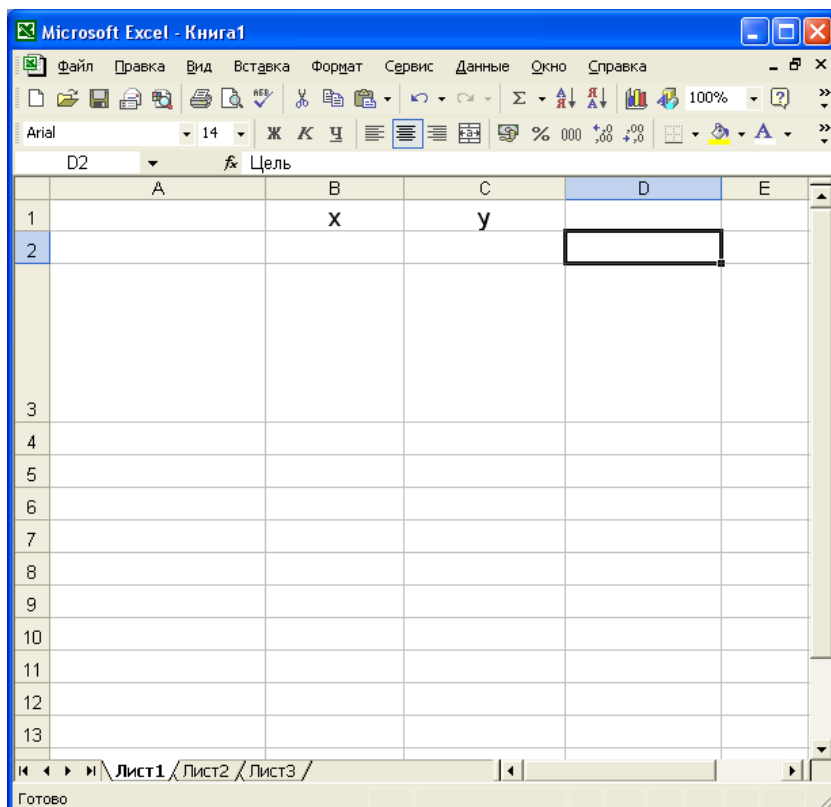


Рисунок 1

3. В ячейки В3:С3 введем значения коэффициентов при неизвестных целевой функции (рис. 2).

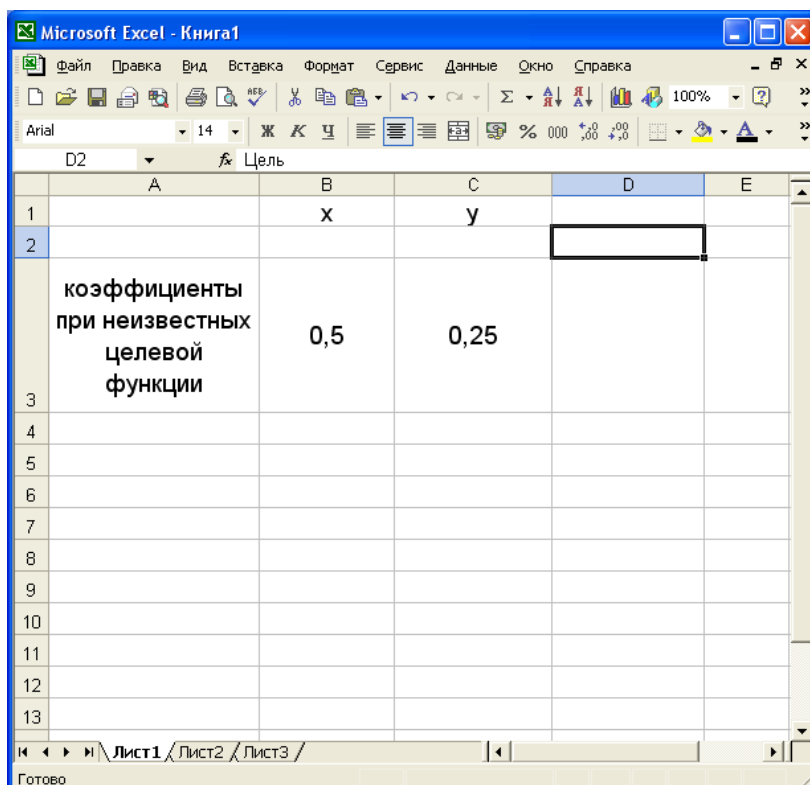


Рисунок 2

4. В ячейку D3 введем целевую функцию  $=B3*B2+C3*C2$ , вычисляющую сумма оценок по предметам (рис. 3).

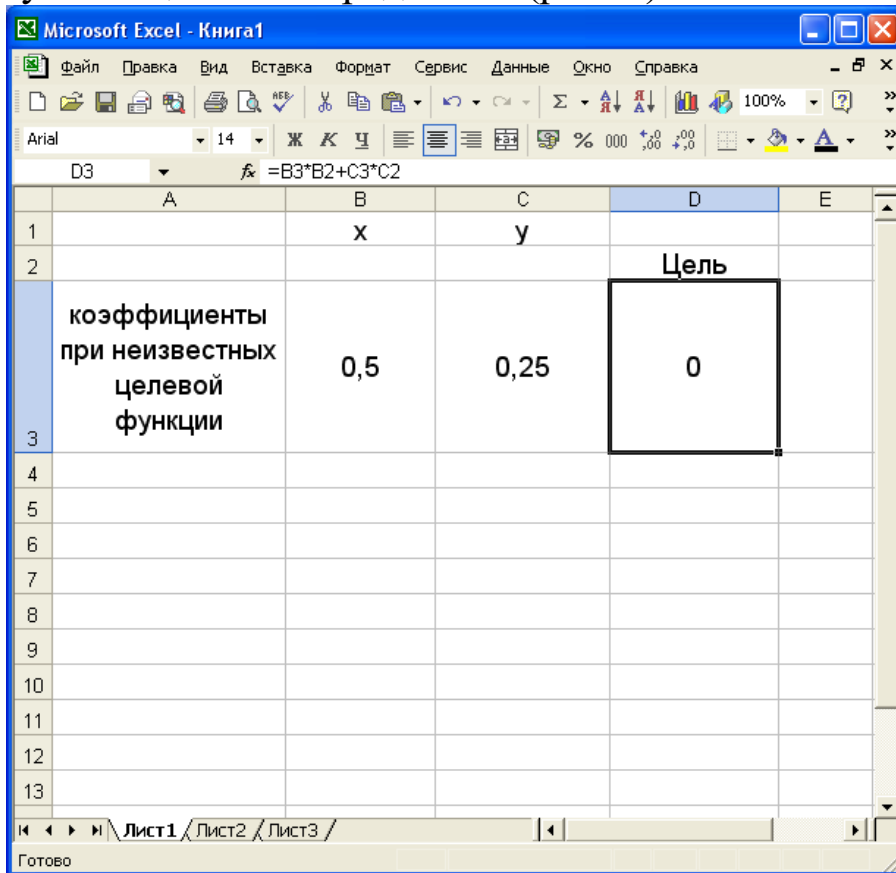


Рисунок 3

5. В ячейки B4:B8 введем формулы

$$=0,5*B2;$$

$$=0,5*B2$$

$$=0,25*C2;$$

$$=0,25*C2;$$

$$=B2+C2,$$

задающие расчетные части ограничений (рис. 4).

	A	B	C	D	E
1		x	y		
2				Цель	
3	коэффициенты при неизвестных целевой функции	0,5	0,25	0	
4	ограничения	0			
5		0			
6		0			
7		0			
8		0			
9					
10					
11					
12					
13					

Рисунок 4

6. В ячейки C4:C8 введем числовые значения в ограничениях (рис. 5).



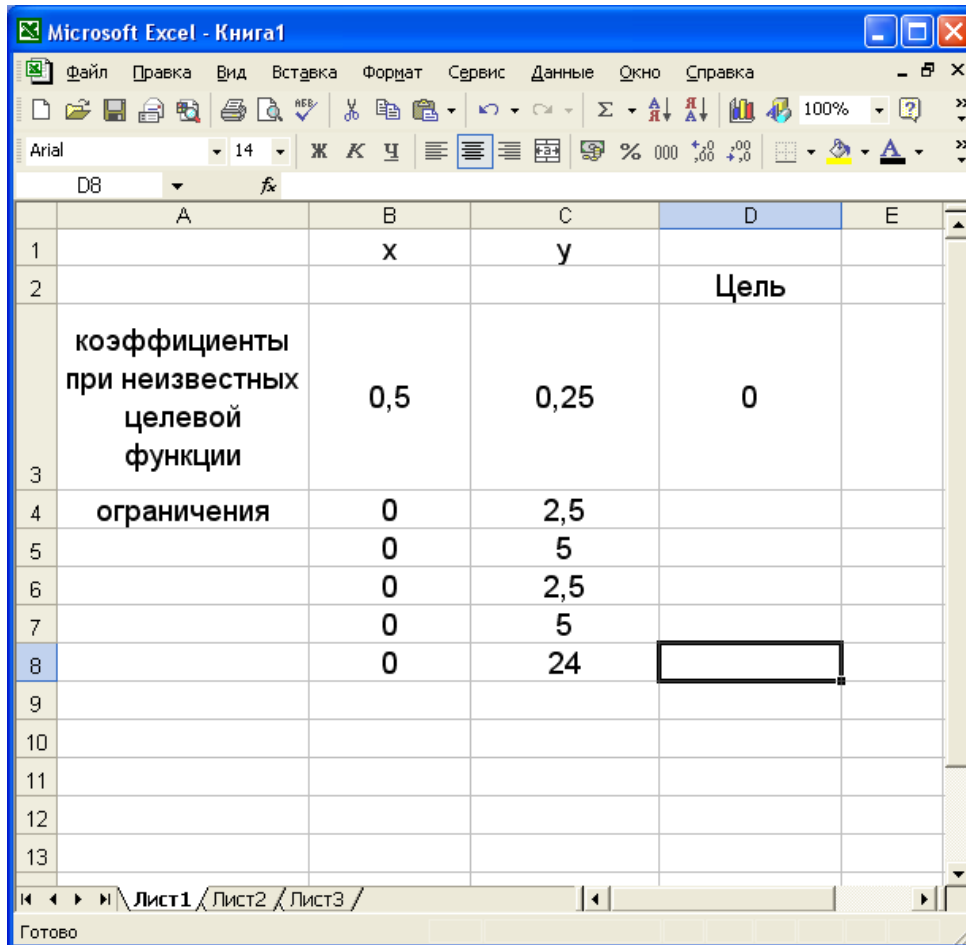


Рисунок 5

7. Затем выбираем команду **Сервис** ⇒ **Поиск решения** и заполним открывающееся диалоговое окно **Поиск решения** как показано на рис. 6.

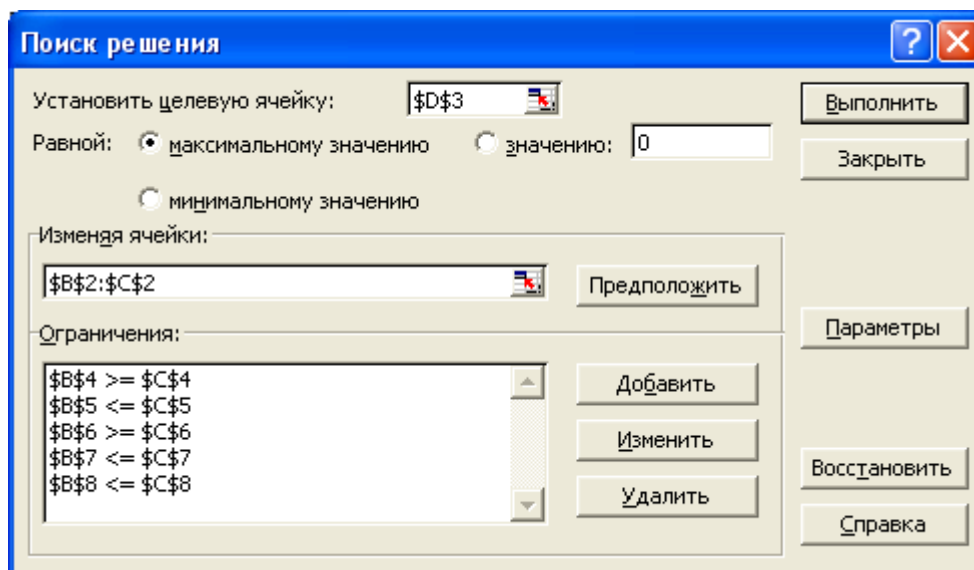


Рисунок 6

8. После заполнения нажимаем кнопку «Выполнить» и в ячейках B2:C2 считываем результат решения (рис 7). Здесь система MS Excel определила, что решением задачи являются значения:  $x = 10$  часов,  $y = 14$  часов.

	A	B	C	D	E
1		x	y		
2		10	14	Цель	
3	коэффициенты при неизвестных целевой функции	0,5	0,25	8,5	
4	ограничения	5	2,5		
5		5	5		
6		3,5	2,5		
7		3,5	5		
8		24	24		
9					
10					
11					
12					
13					

Рисунок 7

## Варианты заданий

### 1. Планирование производства.

Компания Swelte Glove производит и продает два изделия. При продаже каждой единицы изделия 1 компания получает прибыль \$12, а при продаже единицы изделия 2 - \$4. Требования к затратам рабочего времени на производство изделий в трех цехах приводятся в таблице. Начальники цехов оценили, что на следующей неделе ресурс рабочего времени составит в первом цехе 600 ч, в цехе 2 - 800 ч, в цехе 3 - 970 ч. Предполагая, что компания заин-

интересована в максимизации прибыли, разработайте модель линейного программирования для данной задачи.

Цех	Затраты рабочего времени	
	Продукт 1	Продукт 2
1	2	1
2	3	1
3	3	2

## 2. Планирование производства.

Вуд Уолкер владеет небольшим мебельным магазином. Он изготавливает три типа столов: А, В и С. При изготовлении каждого стола необходимо затратить определенное время на производство составных частей, сборку и покраску. Вуд может продать все собранные столы. Кроме того, модель С можно продавать без покраски. Вуд нанял несколько рабочих на условиях неполной занятости, поэтому время, затраченное на изготовление столов, может от недели к неделе меняться. Используя имеющиеся данные, постройте модель линейного программирования, которая поможет Вуду определить ассортимент выпускаемой продукции, максимизирующий его прибыль следующей неделе.

Модель	Изготовление частей, ч	Сборка, ч	Окраска, ч	Удельная прибыль, \$
А	3	4	5	25
В	1	2	5	20
С	4	5	4	50
Неокрашенные столы С	4	5	0	30
Ресурс рабочего времени	150	200	300	

### 3. Финансовое планирование.

Вилли Хснс — президент инвестиционной фирмы, которая управляет инвестиционными портфелями нескольких клиентов. Новый клиент поручает фирме управление своим портфелем и размере \$100 000. Клиент хочет ограничиться покупкой акций трех компаний, характеристики которых представлены в таблице. Постройте модель линейного программирования; которая позволит определить, сколько акций каждой компании должен купить Вилли, чтобы оптимизировать ожидаемых годовой доход.

Акции компании	Цена акции, \$	Ожидаемый годовой доход на акцию, \$	Максимально возможные инвестиции, \$
Gofer Crude	60	7	60 000
Can Oil	25	3	25 000
Sloth Petroleum	20	3	30 000

### 4. Задача составления смеси.

Дуглас Старр управляет питомником домашних животных Heavenly Hound Kennels, Inc. Корм для собак в питомнике готовится из смеси трех зерновых круп, чтобы обеспечить сбалансированное питание. Соответствующие данные приведены в таблице. Дуглас хочет, чтобы каждая собака потребляла ежедневно не менее 3 унций белков, 1 унции углеводов и не более 0,5 унции жиров. Сколько каждой крупы должна получать собака, чтобы минимизировать затраты? (16 унций = 1 фунт.)

Крупа	Стоимость 1 фунта, \$	Белки, %	Углеводы, %	Жиры, %
A	0,45	62	5	3
B	0,38	55	10	2
C	0,27	36	20	1

### 5. Задача составления смеси.

Компания McNaughton, Inc. производит два вида приправ для мяса, Spicy Diablo и Red Baron. Обе приправы готовятся из двух ингредиентов - А и В, причем рецепты приправ допускают определенную вариативность. Допустимое процентное содержание ингредиентов, а также данные о доходах и затратах приводятся в таблице. Можно закупить до 40 кварт ингредиента А и до 30 кварт ингредиента В. Компания в состоянии продать все произведенные приправы. Постройте модель ЛП, цель которой - максимизировать чистый доход от продажи приправ.

Приправа	Ингредиент		Цена за кварту, \$
	А	В	
Spicy Diablo	не менее 25%	не менее 25%	3,35
Red Baron	не более 75%	произвольно	2,85
Цена за кварту, \$	1,60	2,59	

#### 6. Задача составления смеси.

У компании Corey Ander's Spice имеются ограниченные запасы трех ингредиентов, из которых изготавливаются приправы. Ингредиенты НВ01, НВ02 и НВ03 идут на производство двух приправ: куркумы и паприки. Отдел маркетинга сообщил, что компания сможет продать всю произведенную паприку, но не более 1700 бутылок куркумы. Неиспользованные ингредиенты можно продать на свободном рынке. Текущие цены (\$ за унцию): НВ01 - 0,60, НВ02 - 0,70, НВ03 - 0,55. Кроме того, компания заключила контракт на поставку 600 бутылок паприки с Wal-Mart. Другие данные представлены в таблице. Сформулируйте задачу в виде модели ЛП максимизации дохода.

	Ингредиенты			Спрос	Цена, \$
	НВ01	НВ02	НВ03		
Куркума	4	2	1	1700	3,25
Паприка	3	2	3	Неограничено	2,75

ничен

Запас ингредиентов, унции	8 000	9 000	7 000
---------------------------	-------	-------	-------

### 7. Задача составления смеси.

Гай Чанг, администратор зданий и прилегающих территорий университета Готхэма, планирует ранней весной внести удобрения для травы на лужайке. Для нормального роста траве нужны азот, фосфор и калий как минимум в следующих количествах.

Вещество	Минимальное количество, фунты
Азот	10
Фосфор	7
Калий	5

На рынке предлагается три вида минеральных удобрений; содержание требуемых элементов (в фунтах) и цена в расчете на 1000 фунтов представлены в следующей таблице. Гай может купить любое количество каждого из удобрений и смешать их, прежде чем вносить в почву. Постройте модель ЛП, которая позволит определить, сколько следует купить каждого удобрения, чтобы минимизировать затраты.

Удобрение	Содержание азота	Содержание фосфора	Содержание калия	Цена, \$
I	25	10	5	10
II	10	5	10	8
III	5	10	5	7

### 8. Планирование производства.

В выпуске двух продуктов задействованы три станка. Чтобы выпустить фунт продукта каждый станок должен отработать определенное количество часов. Данные приводятся в таблице. Ресурс рабочего времени для станка 1 составляет 10 ч, для станка 2 - 16 ч и

для станка 3 - 12 ч. Удельная прибыль в расчете на 1 фунт составляет \$4 для продукта 1 и \$3 для продукта 2. Определите переменные решения, сформулируйте данную задачу в виде модели ЛП максимизации прибыли и решите ее.

Станок	Количество часов обработки	
	Продукт 1	Продукт 2
1	3	2
2	1	4
3	5	3

9. Компания Solar Car владеет заводом, который может выпускать семейные автомобили (седаны), пикапы и спортивные двухместные автомобили (купе). Удельная прибыль, время производствам фиксированные затраты представлены в таблице.

Модель	Удельная прибыль, \$	Время производства, ч	Фиксированные затраты, \$
Седан	6 000	12	2 000 000
Пикап	8 000	15	3 000 000
Купе	11000	24	7 000 000

Компания к настоящему моменту уже получила заказы на 100 седанов, 200 пикапов и 300 купе, которые она обязана выполнить. Составьте производственный план, который позволит как можно быстрее достичь точки безубыточности, т.е. сделать валовой доход равным валовым фиксированным затратам, минимизировав суммарное время производства. Сформулируйте данную задачу в виде модели ЛП и решите её.

#### 10. Анализ безубыточности.

Риз Эйлер производит воздухоочистители двух видов: Umidaire и Depollinator. Данные о цене и затратах приводятся в таблице. Фирма Риза уже заключила контракт на поставку 500 Umidaire и хочет определить, какое количество воздухоочистителей обоих видов нужно произвести, чтобы достичь безубыточности. Сформулируйте задачу ЛП минимизации затрат и найдите ее решение.

Продукт	Цена 1 шт., \$	Переменные затраты на 1 шт., \$	Фиксирован- ные затраты, \$
Umidaire	450	240	150 000
Depollinator	700	360	240 000

### 11. Управление портфелем активов.

Инвестиционная компания должна определить, куда вложить средства в размере \$10 млн. Цель - максимизировать ожидаемый доход в следующем году. Четыре возможных варианта вложения средств представлены в таблице. Компания также приняла решение, что не менее 30% средств должно быть вложено в обыкновенные акции и долгосрочные казначейские облигации и не более 40% - в фонды денежного рынка и муниципальные облигации. Необходимо инвестировать все имеющиеся \$10 млн. Сформулируйте задачу ЛП, позволяющую определить, куда и сколько вложить средств, и найдите ее решение.

Варианты инвестирования	Ожидаемый доход, %	Максимально возможная сумма инвестиции, млн.долл.
Обыкновенные акции	8	5
Облигации казначейства	6	7
Фонд денежного рынка	12	2
Муниципальные облигации	9	4

### 12. Управление сельскохозяйственным производством.

Организация управляет четырьмя фирмами, производительность которых примерно одинакова. Каждая ферма имеет опреде-



ленную площадь сельхозугодий, ей требуется определенное количество часов рабочего времени, чтобы ухаживать за растениями и убирать урожай. Данные для предстоящего сезона Представлены в таблице.

Ферма	Используемая площадь, акры	Ресурс рабочего времени в месяц, ч
1	500	1700
2	900	3000
3	300	900
4	700	2200

Организация рассматривает варианты выращивания трех культур, которые отличаются ожидаемой удельной прибылью в расчете на один акр занимаемой ими площади и требуемыми трудовыми затратами, что отражено в следующей таблице.

Культура	Максимальная площадь	Трудовые затраты, часы в месяц	Ожидаемая прибыль на акр, \$
A	700	2	500
B	800	4	200
C	300	3	300

Общая площадь, выделенная под каждую определенную культуру, ограничена требованиями к уборочной технике. Чтобы поддерживать примерно одинаковую рабочую нагрузку на фермах, руководство организации решило, что процентные отношения площадей, занятых определенными культурами, должны быть одинаковы на всех фермах. Однако ферма может: 1) выращивать произвольную комбинацию культур, удовлетворяющую всем ограничениям (и том числе и требованию одинаковой нагрузки). Руководство организации хочет узнать, какие площади должны быть выделены под каждую из культур на каждой ферме, чтобы максимизировать ожидаемую прибыль. Сформулируйте задачу в виде модели ЛП и решите ее.

### 13. Задача составления смеси.

Винодел хочет из смеси четырех сортов винограда приготовить три сорта вина. Имеющиеся количества винограда, требования к составу вина к цены, по которым данные вина продаются, приведены в таблице. В частности, сорта винограда 2 и 3 в сумме должны составлять не менее 75% в смеси для приготовления вина А и не менее 35% в смеси для приготовления вина С. Кроме того, смесь А должна содержать не менее 8% винограда сорта 4, а смесь В - не менее 10% сорта 2 и не более 35% сорта 4. Можно продать все произведенные вина. Постройте модель ЛП, которая позволит наилучшим образом использовать выращенный виноград.

Вино	Виноград				Цена за галлон
	1	2	3	4	
А	произвольно	не менее 75% 2 и 3 в любой пропорции		не менее 8%	80
В	произвольно	не менее 10%	произвольно	не более 35%	50
С	произвольно	не менее 35% 2 и 3 в любой пропорции		произвольно	35
Запас (галлон)	130	200	150	350	

### 14. Задача составления расписания.

Ресторан работает 7 дней в неделю. Официанты работают 6 часов в день. Договор с профсоюзом предусматривает, что каждый должен работать 5 дней подряд, а затем 2 дня отдыхать. У всех официантов одинаковый еженедельный оклад. Требования штатного расписания представлены в таблице. Предполагая, что эти требования циклически повторяются, а также игнорируя тот факт, что

число нанятых официантов должно быть целым, постройте модель ЛП, которая позволит руководству составить расписание, удовлетворяющее заданным требованиям при минимальных затратах, и найдите решение полученной задачи.

День недели	Минимальное необходимое количество часов работы официантов
Понедельник	150
Вторник	200
Среда	400
Четверг	300
Пятница	700
Суббота	800
Воскресенье	300

#### 15. Планирование производства.

Завод может производить четыре вида изделий А, В, С и D в произвольных комбинациях. По технологии каждое изделие обрабатывается четырьмя машинами (время обработки в минутах в пересчете на фунт готовой продукции показано в таблице). Каждая машина может работать 60 ч в неделю. Изделия могут продаваться по следующим ценам: А - \$9, В - \$7, С - \$6, D -\$5 за фунт. Переменные затраты на оплату труда составляют \$2 в час для машин 1 и 2 и \$3 в час для машин 3 и 4. Материальные затраты составляют \$4 на каждый фунт продукции А и \$1 на каждый фунт продукции В, С и D. Сформулируйте задачу ЛП, максимизирующую прибыль при заданном максимальном спросе для каждого вида продукции и решите ее.

Продукция	Машина				Максимальный спрос
	1	2	3	4	
А	5	10	6	3	400
В	3	6	4	8	100
С	4	5	3	3	150

D                    4                    2                    1                    2                    500

### 16. Планирование производства.

В данном месяце необходимо выполнить четыре вида работ: А, В, С и D. Каждый вид работ может выполняться любым из трех цехов. Время, требуемое на выполнение конкретного задания определенным цехом, стоимость часа работы каждого цеха и ресурс рабочего времени на данный месяц в каждом цехе представлены в таблице. Каждое задание можно распределить по нескольким цехам в любой пропорции. Например, четверть задания А может быть выполнена за 8 ч цехом 1, а треть задания С - за 9 ч цехом 3. Необходимо распределить выполнение заданий по цехам (сколько часов каждый цех должен выполнять каждое определенное задание), чтобы минимизировать затраты на выполнение всех заданий. Определите переменные решения, сформулируйте задачу ЛП и решите ее.

Цех	Время, ч				Стоимость часа работы, \$	Ресурс рабочего времени, ч
	А	В	С	D		
1	32	151	72	118	89	160
2	39	147	61	126	81	160
3	46	155	57	121	84	160

### 17. Составление расписания.

Базирующийся в Стокгольме авианосец "Майти" с понедельника по пятницу находится на маневрах, а во время уик-энда - в порту. На следующей неделе капитан хочет предоставить увольнение на берег во время уик-энда максимально возможному числу моряков. (Экипаж состоит из 2500 человек.) Однако при этом необходимо выполнить запланированные на неделю маневры и удовлетворить требования военно-морского регламента.

Требования таковы.

а. Моряки работают в первую (от полуночи до полудня) или во вторую (от полудня до полуночи) смену, причем на протяжении

недели каждый моряк все дни работает в одну и ту же смену.

в. Каждый моряк должен дежурить ровно четыре дня, даже если в какой-то день реальной работы недостаточно.

В таблице показано, сколько моряков должно дежурить каждый день в каждую смену. Сформулируйте для данной задачи модель ЛП, чтобы узнать, сколько моряков должно работать каждый день.

	Пн.	Вт.	Ср.	Чт.	Пт.
1 смена	900	1000	450	800	700
2 смена	800	500	1000	300	750

18. Небольшая фирма использует два процесса для изготовления двух продуктов: жидкости для стартера и жидкости для зажигалок. Руководство фирмы пытается решить, сколько часов уделить каждому из процессов. Данные о процессах (в пересчете на один час) приведены в таблице. Пусть  $x_1$  и  $x_2$  - число часов, выделенных компанией на проведение процессов 1 и 2 соответственно. Федеральной программой, компании выделено 300 единиц керосина и 450 единиц бензина. Согласно торговым соглашениям компания должна произвести не менее 600 единиц жидкости для стартера и 225 единиц жидкости для зажигалок. Почасовая прибыль, получаемая при проведении процесса 1 и 2, составляет \$450 и \$390 соответственно. Сформулируйте задачу в виде модели ЛП максимизации прибыли и решите ее.

Процесс	Исходные вещества		Конечные продукты	
	Керосин	Бензин	Жидкость для стартеров	Жидкость для зажигалок
1	3	9	15	6
2	12	6	9	24

19. Составление расписания.

Ресторан работает 7 дней в неделю. По условиям найма официанты работают 6 часов в день. В ресторан приходят отдельные посетители и небольшие компании, их посещения будем называть регулярным спросом. Кроме того, более многочисленные группы (клубы по интересам и т.п.) иногда собираются в ресторане на свои еженедельные встречи. По соглашению с профсоюзом официант работает пять дней подряд, а затем два дня подряд отдыхает. Все официанты получают одинаковую недельную заработную плату. Минимально необходимое ежедневное рабочее время зависит от регулярного ежедневного спроса, к которому добавлено количество рабочего времени, необходимого для обслуживания запланированных на этот день крупных встреч. Регулярный спрос (выраженный в человеко-часах) и число встреч, запланированных на каждый день, представлены в таблице.

День недели	Регулярный спрос	Количество запланированных встреч
Понедельник	125	1
Вторник	200	0
Среда	350	1
Четверг	300	0
Пятница	650	3
Суббота	725	4
Воскресенье	250	2

Чтобы определить, сколько человеко-часов необходимо для обслуживания встреч, управляющий использует следующую таблицу. Необходимо составить расписание работы официантов, удовлетворяющее потребности в обслуживании и минимизирующее затраты, предполагая, что данный цикл неограниченно повторяется, и игнорируя тот факт, что число нанятых официантов должно быть целым. Оптимизируйте полученную модель при существующих значениях данных.

Число запланированных встреч	Требуемое количество человеко-часов
0	0
1	24
2	36
3	52
4	64
5	80

20. У компании Slick Oil есть три склада, с которых она может доставлять продукцию трем розничным торговым точкам. Спрос на продукт Gunkout составляет в первой точке 100 ящиков, во второй - 250, в третьей - 150. Запас данного продукта на складе 1 составляет 50 ящиков, на складе 2 - 275 и на складе 3 - 175. Затраты на транспортировку одного ящика продукта Gunkout со складов в торговые точки представлены в таблице. Постройте модель ЛП, позволяющую определить, сколько ящиков следует доставить в каждую торговую точку с каждого склада, чтобы удовлетворить спрос с минимальными затратами.

Склад	Торговые точки		
	1	2	3
1	5	7	6
2	8	9	10
3	4	3	11

21. Компания Candy's Fragrances выпускает на своей фабрике, расположенной в Санта-Мария, Калифорния, три вида продукции: духи, шампунь и крем для лица. Фабрика работает в две смены. В первую смену час рабочего времени стоит \$15, а ресурс рабочего времени составляет не более 30 000 ч. При работе во вторую смену платится надбавка \$2 в час, поэтому стоимость часа рабочего времени составляет \$17. Ресурс рабочего времени второй смены не

превышает 20 000 ч. Для производства каждого вида продукции используется три вида сырья. Компания платит \$2 за фунт сырья А, \$2.50 за фунт сырья В и \$2 за фунт сырья С. Данные о трудозатратах, расходах сырья и спросе представлены в таблице.

	Духи	Шампунь	Крем для лица
Трудозатраты, ч/упаковка	1,5	2	2
Расход сырья А, фунты/упаковка	5	7	4
Расход сырья В, фунты/упаковка	2	2	5
Расход сырья С, фунты/упаковка	3	5	3
Спрос, упаковка	10 000	7 000	15 000

Компания может также покупать шампунь и крем для лица у зарубежного поставщика. Расходы по доставке импортного шампуня составляют \$50 за упаковку, а крема — \$60 за упаковку. Сформулируйте модель ЛП минимизации затрат при условии удовлетворения (или превышения) спроса на продукцию компании.

22. Компания Bear Lake Electronics выпускает три вида DVD-проигрывателей: базовый, класса люкс и портативный. Компания хочет оптимизировать ассортимент продукции, выпускаемой на трех ее предприятиях. Предположим, что компания может продать все произведенные DVD-проигрыватели. Производственные мощности предприятий по изготовлению комплектующих и сборке представлены в следующей таблице.

Предприятие	Сборка, ч	Изготовление комплектующих, ч
-------------	-----------	-------------------------------



1	20 000	100 000
2	30 000	100 000
3	10 000	70 000
Всего	60 000	270 000

Прибыль от продажи базовой модели составляет \$75, модели класса люкс - \$125, а портативной модели - \$195. Время, необходимое для изготовления комплектующих и сборки различных проигрывателей, приводится в таблице.

Проигрыватель	Сборка, ч на 1 изделие	Изготовление комплектующих, ч на 1 изделие
Базовая модель	3	8
Модель класса люкс	4	11
Портативная модель	8	16

Создайте модель ЛП и найдите оптимальный ассортимент продукции для компании Bear Lake Electronics.

23. Компания Environmental Energy Associates (EEA) владеет небольшим коммунальным предприятием по производству электроэнергии в южной Калифорнии. Предприятие генерирует электричество следующим образом: сжигая три вида топлива, нагревают паровой котел, а струя пара вращает турбину генератора. В качестве топлива для котла могут использоваться навоз, древесные отходы, а также природный газ. Компании необходимо минимизировать затраты на топливо, при этом производство электроэнергии в следующем месяце должно составить не менее 7200 МВт\*ч. Навоз поставляется владельцами расположенных поблизости ранчо и откормочных хозяйств, которые платят компании EEA \$50 за утилизацию тонны навоза. Древесные отходы закупаются на местной лесопилке по цене \$25 за тонну. Природный газ поставляется по трубопроводу по цене \$120 за миллион стандартных кубических

футов. В следующей таблице приводятся выход энергии, ресурс и удельная стоимость каждого вила топлива.

Топливо	Выход энергии	Ресурс	Стоимость
Навоз	10 МВт*ч/т	100 т	\$50 за тонну
Древесные отходы	7 МВт*ч/т	200 т	\$25 за тонну
Природный газ	40 МВт*ч/млн. куб, футов	Неограниченно	\$120 за млн. куб, футов

Разработайте модель ЛП, которая позволит решить поставленную задачу (минимизировать затраты при заданном уровне производства энергии). Сколько топлива каждого вида следует использовать для выработки электроэнергии в следующем месяце?

24. При откормке каждого животного ежедневно должно получаться не менее 9 единиц питательного вещества S1, 8-S2, 12-S3. Содержание количества единиц питательного вещества в 1 кг каждого вида корма и стоимость 1 кг в таблице. Необходимо составить дневной рацион нужной питательности, причем с минимальными затратами.

Вещества	Количество пит. веществ	
	Корм 1	2
S1	4	2
S2	2	1
S3	2	4
Стоимость 1 кг корма	6	5

25. Авиакомпания МОГОЛ по заказу армии должна перевезти на некотором участке 700 человек. В распоряжении компании имеется два типа самолетов, которые можно использовать для перевозки. Самолет первого типа перевозит 30 пассажиров и имеет экипаж 3 человека, второго типа – 65 и 5 соответственно.

Эксплуатация 1 самолета первого типа обойдется 4000\$, а второго 9000\$. Сколько надо использовать самолетов каждого типа (с учетом минимальности затрат), если для формирования экипажей имеется не более 60 человек.

26. Компании потребовалось в течении 12 часов перевезти 150 ящиков своей продукции. Для этого она обратилась в компанию по грузоперевозкам, у которой в наличии свободными оказались два грузовых автомобиля А и Б (см.табл.). Определить какое количество рейсов каждого автомобиля необходимо выполнить, чтобы затраты компании на перевозку были минимальны.

Показатель	Автомобиль	
	А	Б
Стоимость часа эксплуатации	1100	1300
	руб	руб
Вместимость	15 шт.	20 шт.
Время затрачиваемое на один рейс (в т.ч. загрузка и выгрузка товара)	2 ч.	2,5 ч

27. Необходимо не позднее, чем за 5 часов заполнить бассейн объемом 10000 л. Имеется в наличии два насоса (см. табл.). Определить сколько часов должен работать каждый насос, чтобы затраты на электроэнергию были минимальны.

Показатель	Насос	
	А	Б
Производительность, л/ч	1000	1500
Мощность, кВт	2	4.

28. Имеется некоторый материал в виде стандартных листов, которые необходимо раскроить для получения не менее 80шт. деталей типа 1 и не менее 40шт. деталей типа 2. Известны два способа раскроя листа, каждый из которых дает результат, представленный в таблице.

Способы раскроя листа	1	2

Результат	3 детали типа 1; 1 деталь типа 2	2 детали типа 1; 6 деталей типа 2
-----------	-------------------------------------	--------------------------------------

Требуется так провести операцию изготовления деталей, чтобы общий расход листов оказался минимальным.

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Железимова  
« 14 » \_\_\_\_\_ 20\_\_ г.



**ИЗУЧЕНИЕ И ИССЛЕДОВАНИЕ ЧИСЛЕННЫХ МЕТОДОВ  
РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

УДК 519.6

Составители: В.В. Куц

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Изучение и исследование численных методов решения нелинейных уравнений:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 14 с.:

Содержат сведения о решении нелинейных уравнений численными методами. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 0,81 . Уч.-изд. л. 0,74.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

**1. Задание:** Найти с заданное погрешностью  $\varepsilon$  корень заданного уравнения  $f(x)$  на отрезке  $[a;b]$  методами: половинного деления; методом хорд; методом Ньютона.

## 2. Краткие теоретические сведения о численных методах решения уравнений

### 2.1. Постановка задачи

Нахождение корней уравнения — это одна из древнейших математических проблем, которая не теряет своей остроты и в наши дни: она часто встречается в самых разнообразных областях науки и техники.

В более общем случае, если имеется некоторая функция  $f(x)$ , то бывает необходимо найти такие значения аргумента  $x$ , для которых

$$f(x) = 0. \quad (1)$$

Решение этого основного уравнения и является нашей целью. Уравнение будем называть линейным, нелинейным, алгебраическим или трансцендентным в зависимости от того, имеет ли оно одно решение,  $n$  решений или неопределенное число решений. Систему уравнений будем называть линейной или нелинейной в зависимости от математической природы входящих в нее уравнений.

Решение линейного уравнения с одним неизвестным получается достаточно просто и здесь не рассматривается.

Обычно нелинейные уравнения делят на трансцендентные и алгебраические. Трансцендентными называются нелинейные уравнения, содержащие тригонометрические функции или другие специальные функции, например,  $\lg(x)$  или  $e^x$ . Уравнения, содержащие суммы целых степеней  $x$ , называются алгебраическими. Их общий вид:

$$a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \dots + a_1 x + a_0 = 0$$

Так как оба типа нелинейных уравнений часто решаются одними и теми же методами, то далее рассматриваются методы, одинаково приспособленные для решения обоих типов.

Классифицировать уравнения можно в соответствии с нижеприведенной схемой (рис. 1)

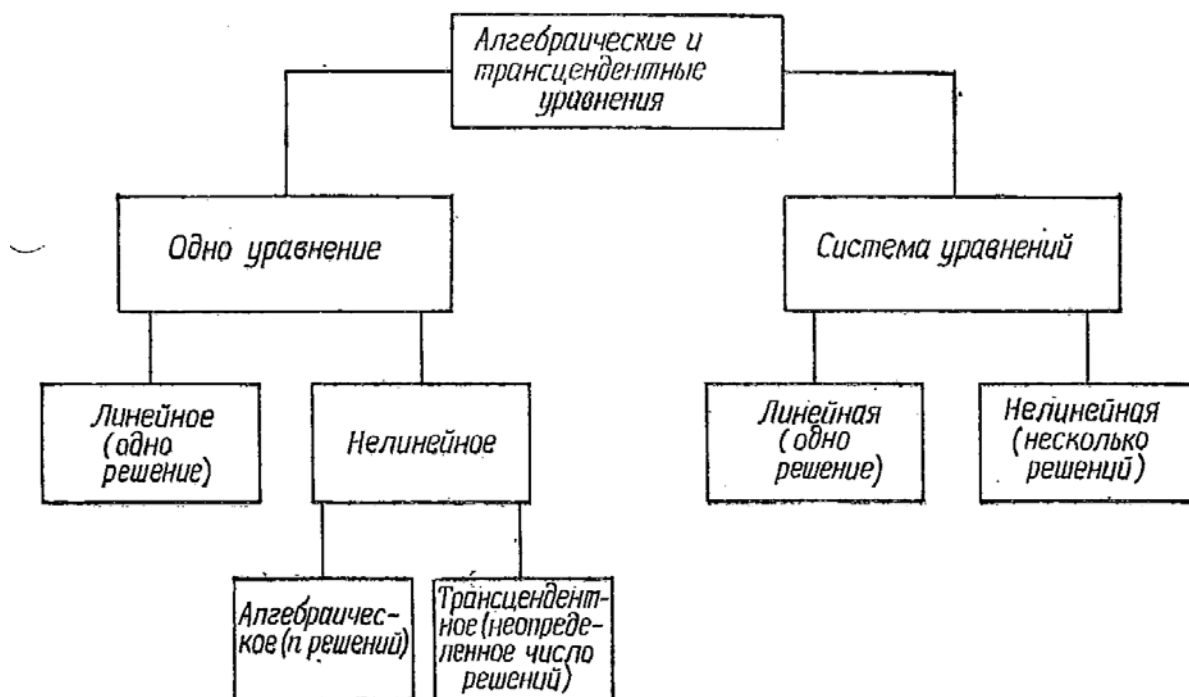


Рис. 1. Классификация уравнений

Инженеру часто приходится решать алгебраические и трансцендентные уравнения, что может представлять собой самостоятельную задачу или являться составной частью более сложных задач. В обоих случаях практическая ценность численного метода в значительной мере определяется быстротой и эффективностью получения решения.

Выбор подходящего алгоритма для решения уравнений зависит от характера рассматриваемой задачи.

Методы решения нелинейных уравнений делятся на прямые и приближенные. Первые позволяют найти решение непосредственно с помощью формул и всегда обеспечивают получение точного решения. Известным примером такого рода является формула корней квадратного уравнения.

Но в общем случае функции, с которыми приходится работать инженеру и ученому, не имеют аналитических формул для своих корней в противоположность, например, тому же квадратному уравнению.

Поэтому приходится пользоваться приближенными способами нахождения корней, которые, как правило, состоят из двух этапов:

1. Отыскание приближенного значения корня.



2. Уточнение приближенного значения до некоторой заданной степени точности.

Очень часто приближенное значение корня бывает известно из физических соображений, в других случаях можно использовать графические методы. Кроме того, существуют специальные методы нахождения приближенного корня для того практически важного случая, когда  $f(x)$  является полиномом. Но чаще всего приходится находить интервал значений неизвестной  $x$ , на котором функция меняет знак, и после этого применять какие-нибудь алгоритмы для уточнения корня.

Строго говоря, к численным методам решения уравнений как раз и относят различные методы, связанные с уточнением первоначального приближения. Численный метод, в котором проводится последовательное, шаг за шагом, уточнение первоначального грубого приближения, называется *методом итераций*. Каждый шаг в таком методе называется *итерацией*. Если при последовательных итерациях получаются значения, которые все ближе и ближе приближаются истинному значению корня, то говорят, что метод итераций *сходится*.

Для реализации итерационных методов задается процедура решения в виде многократного применения некоторого алгоритма. Полученное решение всегда является приближенным, хотя может быть сколь угодно близким к точному.

Итерационные методы наиболее удобны для реализации на компьютере.

В каждом из излагаемых далее методов считается, что решаемая задача состоит в отыскании действительных корней (нулей) уравнения (1). ( $f(x) = 0$ ). Хотя подобные уравнения также могут иметь комплексные корни, способы их отыскания обычно рассматриваются только для алгебраических уравнений.

## ***2.2. Методы решения нелинейных уравнений***

### *2.2.1. Метод половинного деления*

Метод включает нахождение интервала значений переменной  $x$ , на котором функция меняет знак. Для этого вычисляются значения функции через равные промежутки значений  $x$ . Очевидно, что от выбора шага  $x$  зависит правильность нахождения этого интерва-

ла и всегда имеется возможность пропустить корень в случае, когда функция несколько раз меняет знак на отрезке переменной меньше заданного шага.

Блок-схема алгоритма метода половинного деления представлена на рис.2.

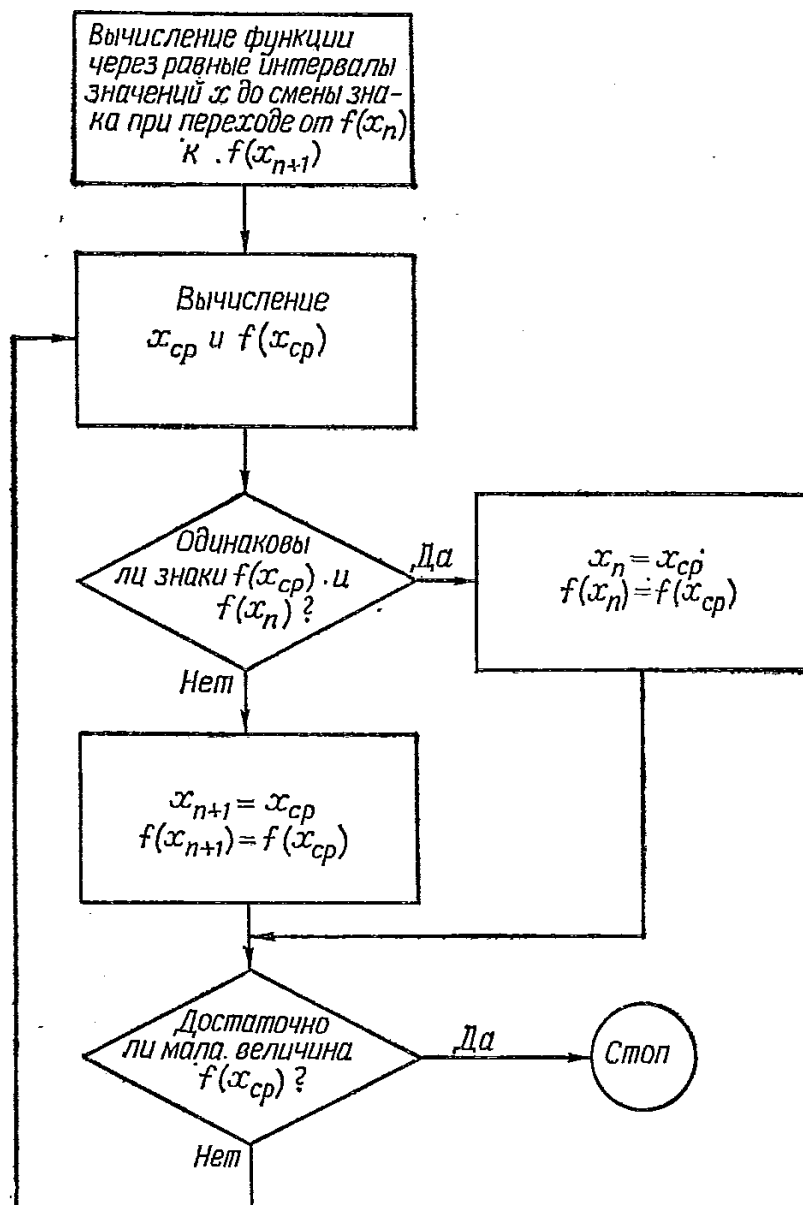


Рис. 2. Блок-схема алгоритма половинного деления

Он состоит из следующих операций. Сначала вычисляются значения функций в точках, расположенных через равные интервалы на оси  $x$ . Это делается до тех пор, пока не будут найдены два последовательных значения функции  $f(x_n)$  и  $f(x_{n+1})$ , имеющие противоположные знаки. Напомним, если функция непрерывна, изменение знака указывает на существование корня.

Затем по формуле  $x_{cp} = \frac{x_{n+1} + x_n}{2}$  вычисляется среднее значение  $x$  в интервале значений  $[x_n, x_{n+1}]$  и находится значение функции  $f(x_{cp})$ .

Если знак  $f(x_{cp})$  совпадает со знаком  $f(x_n)$ , то в дальнейшем, вместо  $f(x_n)$  используется  $f(x_{cp})$ . Если же  $f(x_{cp})$  имеет знак, противоположный знаку  $f(x_n)$ , т.е. её знак совпадает со знаком  $f(x_{n+1})$ , то на  $f(x_{cp})$  заменяется это значение функции. В результате интервал, в котором заключено значение корня, сужается.

Если  $f(x_{cp})$  достаточно близко к нулю, процесс заканчивается, в противном случае он продолжается. Чаще всего для оценки близости к нулю применяется оценка по абсолютной сходимости двух последовательных приближений  $|x_{n+1} - x_n| \leq \varepsilon$ , хотя могут применяться и другие оценки точности.

На рис. 3 эта процедура показана графически.

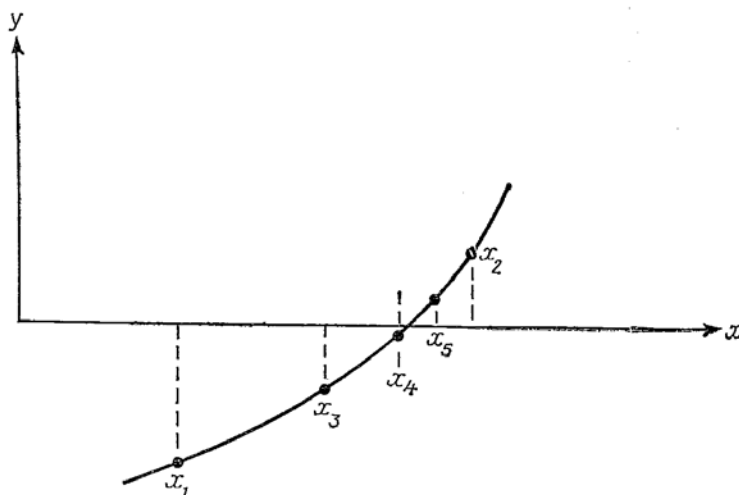


Рис. 3. Метод половинного деления

Хотя метод половинного деления не обладает высокой вычислительной эффективностью, с увеличением числа итераций он обеспечивает получение все более точного приближенного значения корня. После того как впервые найден интервал, в котором заключен корень, его ширина после  $N$  итераций убывает в  $2^N$  раза.

### 2.2.2. Метод хорд

Как и предыдущий метод, метод хорд включает предварительное нахождение интервала значений переменной  $x$ , на котором функция меняет знак.

В основе этого метода лежит линейная интерполяция по двум значениям функции, имеющим противоположные знаки. При отыскании корня этот метод нередко обеспечивает более быструю сходимость, чем предыдущий. Блок-схема алгоритма метода ложного положения дана на рис. 2.4. Счет ведется следующим образом. Сначала определяются значения функции в точках, расположенных на оси  $x$  через равные интервалы. Это делается до тех пор, пока не будет найдена пара последовательных значений функции  $f(x_n)$  и  $f(x_{n+1})$ , имеющих противоположные знаки.

Прямая, проведенная через эти две точки, пересекает ось  $x$  при значении

$$x^* = x_n - f(x_n) \frac{x_{n+1} - x_n}{f(x_{n+1}) - f(x_n)}.$$

Это значение аргумента используется для определения значения функции  $f(x^*)$ , которое сравнивается со значениями функций  $f(x_n)$  и  $f(x_{n+1})$  и в дальнейшем используется вместо того из них, с которым оно совпадает по знаку.

Если значение  $f(x^*)$  недостаточно близко к нулю, то вся процедура повторяется до тех пор, пока не будет достигнута необходимая степень сходимости. На рис. 5 процесс решения показан графически.

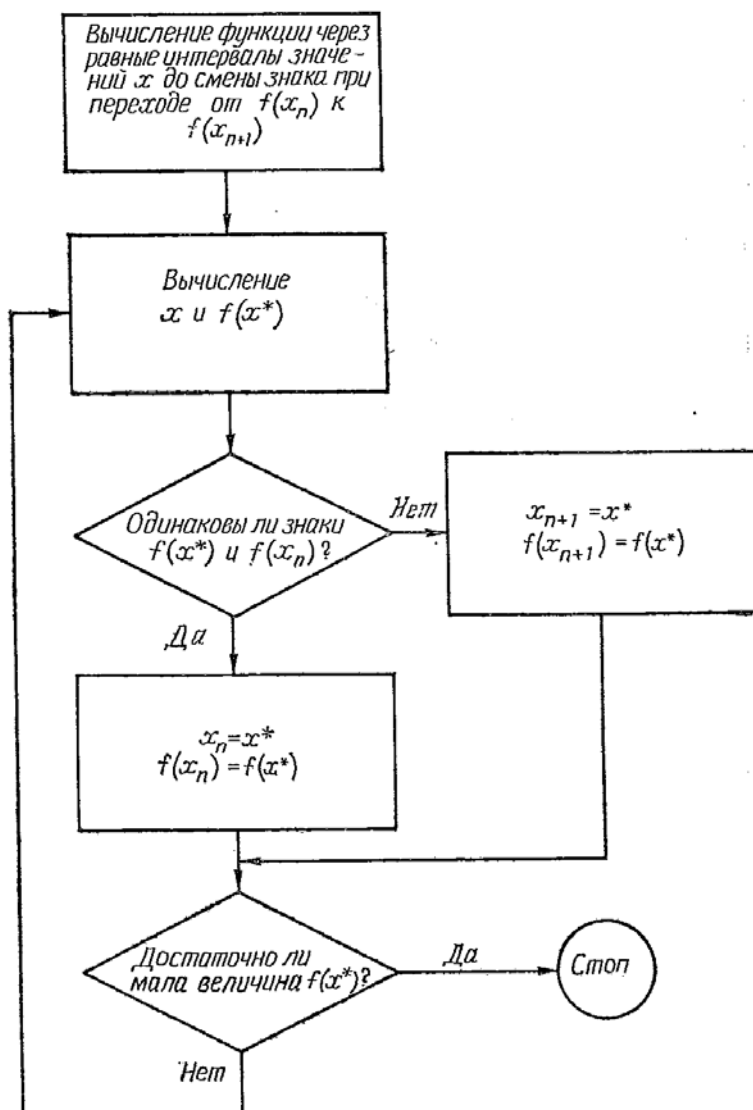


Рис. 4. Блок-схема алгоритма метода хорд

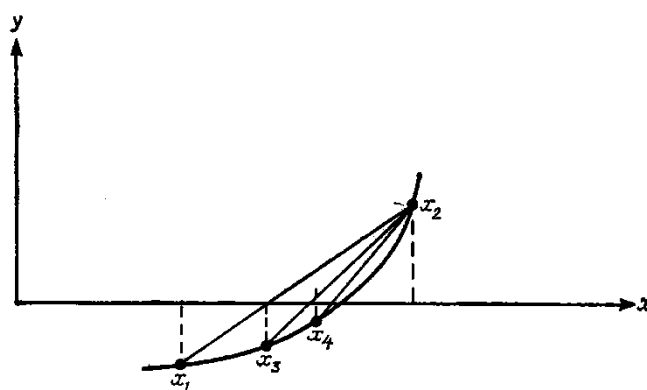


Рис. 5. Метод хорд

### 2.2.3. Метод Ньютона (касательных)

Метод последовательных приближений, разработанный Ньютоном, очень широко используется при построении итерационных алгоритмов. Его популярность обусловлена тем, что в отличие от двух предыдущих методов для определения интервала, в котором заключен корень, не требуется находить значения функции с противоположными знаками. Вместо интерполяции по двум значениям функции в методе Ньютона осуществляется экстраполяция с помощью касательной к кривой в данной точке.

На рис. 6. ниже показана блок-схема алгоритма этого метода, в основе которого лежит разложение функции  $f(x)$  в ряд Тейлора

$$f(x_n + h) = f(x_n) + hf'(x_n) + \frac{1}{2}h^2 f''(x_n) + \dots$$

Члены, содержащие  $h$  во второй и более высоких степенях, отбрасываются; используется соотношение  $x_n + h = x_{n+1}$ . Предполагается, что переход от  $x_n$  к  $x_{n+1}$  приближает значение функции к нулю так, что  $f(x_n + h) = 0$ . Тогда

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Значение  $x_{n+1}$  соответствует точке, в которой касательная к кривой в точке  $x_n$  пересекает ось  $x$ . Так как кривая  $f(x)$  отлична от прямой, то значение функции  $f(x_{n+1})$  скорее всего не будет в точности равно нулю. Поэтому вся процедура повторяется, причем вместо  $x_n$  используется  $x_{n+1}$ . Счет прекращается по достижении достаточно малого значения  $f(x_{n+1})$ . На рис. 7 процесс решения уравнения методом Ньютона показан графически. Совершенно ясно, что быстрота сходимости в большой мере зависит от удачного выбора исходной точки. Если в процессе итераций тангенс угла наклона касательной  $f(x)$  обращается в нуль, то применение метода осложняется. Можно также показать, что в случае бесконечно большого  $f''(x)$  метод также не будет достаточно эффективным. Так как условие кратности корней имеет вид  $f(x) = f'(x) = 0$ , то в этом случае метод Ньютона не обеспечивает сходимость. Отметим, что иногда используется другой способ контроля сходимости, состоящий в сравнении  $x_n$  и  $x_{n+1}$ .

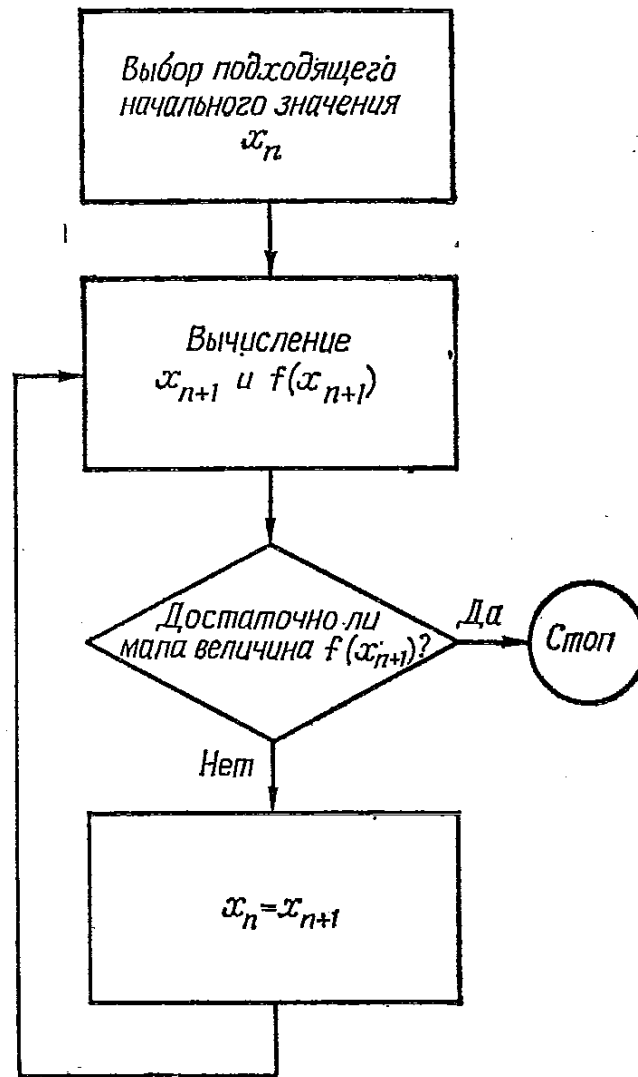


Рис. 6. Блок-схема алгоритма метода Ньютона

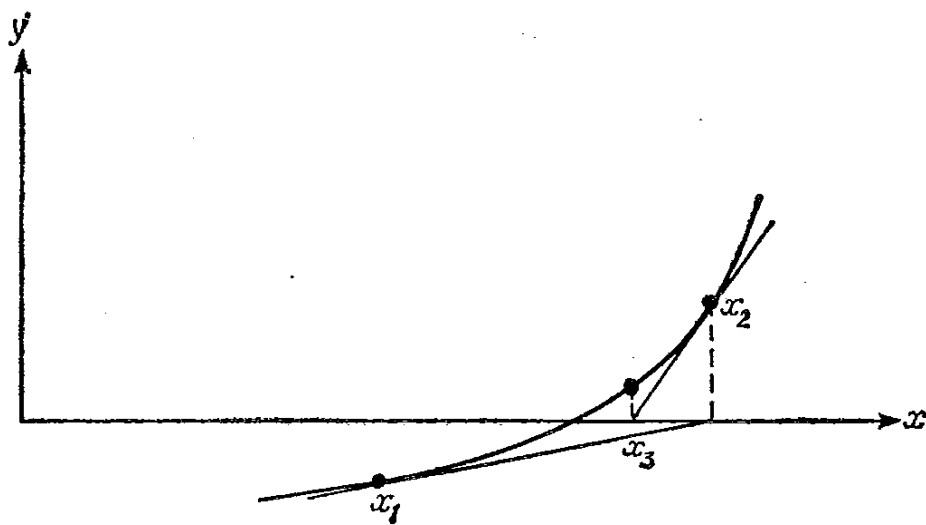


Рис. 7. Метод Ньютона

### 3. Пример выполнения работы

Найдем с погрешностью  $|f(x)| \leq 0.0001$  корень уравнения  $f(x) = x^2 - \sqrt{x+4} = 0$  на отрезке  $[1;2]$ .

На рис. 8-10 показаны листы Excel в которых выполнено пошаговое решение уравнения методами: половинного деления; методом хорд и методом Ньютона.

	A	B	C	D	E	F	G	H
	Шаг	a	b	d	f(a)	f(b)	f(d)	f(a)f(d)
1	1	1,0000	2	1,5	-1,23607	1,55051	-0,095208	1
2	2	1,5000	2	1,75	-0,09521	1,55051	0,664584	-1
3	3	1,5000	1,75	1,625	-0,09521	0,664584	0,268917	-1
4	4	1,5000	1,625	1,5625	-0,09521	0,268917	0,082911	-1
5	5	1,5000	1,5625	1,53125	-0,09521	0,082911	-0,007134	1
6	6	1,5313	1,5625	1,546875	-0,00713	0,082911	0,037642	-1
7	7	1,5313	1,546875	1,539063	-0,00713	0,037642	0,015192	-1
8	8	1,5313	1,539063	1,535156	-0,00713	0,015192	0,004013	-1
9	9	1,5313	1,535156	1,533203	-0,00713	0,004013	-0,001564	1
10	10	1,5332	1,535156	1,53418	-0,00156	0,004013	0,001224	-1
11	11	1,5332	1,53418	1,533691	-0,00156	0,001224	-0,000171	1
12	12	1,5337	1,53418	1,533936	-0,00017	0,001224	0,000526	-1
13	13	1,5337	1,533936	1,533813	-0,00017	0,000526	0,000178	-1
14	14	1,5337	1,533813	1,533752	-0,00017	0,000178	0,000004	-1
15								

Рис. 8. Решение уравнения методом половинного деления



$=B2-E2*(C2-B2)/(F2-E2)$

	A	B	C	D	E	F	G	H
1	Шар	a	b	d	f(a)	f(b)	f(d)	f(a)f(d)
2	1	1,0000	2	1,44358	-1,23607	1,55051	-0,24923	1
3	2	1,4436	2	1,52063	-0,24923	1,55051	-0,03728	1
4	3	1,5206	2	1,53189	-0,03728	1,55051	-0,00532	1
5	4	1,5319	2	1,53349	-0,00532	1,55051	-0,00075	1
6	5	1,5335	2	1,53371	-0,00075	1,55051	-0,00011	1
7	6	1,5337	2	1,53375	-0,00011	1,55051	-0,00002	1
8								

Рис. 9. Решение уравнения методом хорд

$=B2-D2/F2$        $=B2+0,0000000001$        $=(E2-D2)/(C2-B2)$

	A	B	C	D	E	F
1	Шар	a	a+delta	f(a)	f(a+delta)	f'(a)
2	1	1	1,00000000010000	-1,23607	-1,23607	1,77639222
3	2	1,695831	1,69583055142248	0,48925	0,48925	3,182160938
4	3	1,542084	1,54208366502726	0,02386	0,02386	2,871778133
5	4	1,533776	1,53377557897227	0,00007	0,00007	2,855000444

Рис. 9. Решение уравнения методом Ньютона

## 4. Задания для самостоятельной работы

№	Ур-е	Интервал	№	Ур-е	Интервал
1	$x - \sin(x) = 0,25$	$0 \div \pi$	13	$x \cdot \ln(x + 1) - 0,3 = 0$	$0,1 \div 10$
2	$x^3 = e^x - 1$	$4 \div 6$	14	$x^2 - \sin(10 \cdot x) = 0$	$\pi/16 \div 19\pi/128$
3	$\sqrt{x} - \cos(x) = 0$	$0 \div \pi$	15	$ctan(x) = x$	$\pi/16 \div 7\pi/8$
4	$x^2 + 1 = \arccos(x)$	$-0,5 \div 0,5$	16	$tg(3x) + 0,4 = x^2$	$7\pi/32 \div 15\pi/32$
5	$\lg(x) - \frac{7}{2x+6} = 0$	$3 \div 10$	17	$x^2 + 1 = tg(x)$	$-\pi/2 \div \pi/2$
6	$tg(0,5x + 0,2) = x^2$	$-\pi \div 0$	18	$x^2 - 1 = \ln(x)$	$0,8 \div 5$
7	$3x - \cos(x) - 1 = 0$	$-2\pi \div 2\pi$	19	$0,5^x + 1 = (x - 2)^2$	$2 \div 10$
8	$x + \lg(x) = 0,5$	$0 \div 10$	20	$(x + 3) \cos(x) = 1$	$\pi/4 \div \pi$
9	$x^2 = \arcsin(x - 0,2)$	$0 \div 0,5$	21	$x^2 \cos(2x) = -1$	$5\pi/8 \div \pi$
10	$x^2 + 4 \sin(x) = 2$	$0 \div \pi$	22	$\cos(x + 0,3) = x^2$	$0 \div 2\pi$
11	$ctg(x) - x^2 = 0$	$0 \div \pi$	23	$2^x(x - 1)^2 = 2$	$1 \div 5$
12	$tg(x) = \cos(x) - 0,1$	$-\pi/2 \div \pi/2$	24	$x \cdot \ln(x + 1) = 0,5$	$0 \div 10$

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Желтинова  
« 14 » \_\_\_\_\_ г.



**ОПТИМИЗАЦИЯ ТРАЕКТОРИЙ ВСПОМОГАТЕЛЬНЫХ  
ПЕРЕМЕЩЕНИЙ ИНСТРУМЕНТА НА ОСНОВЕ ТЕОРИИ  
ГРАФОВ**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

**УДК 519.6**

Составители: В.В. Куц, А.В. Олещицкий

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Оптимизация траекторий вспомогательных перемещений инструмента на основе теории графов:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц, А.В. Олещицкий. – Курск, 2023. – 46 с.:– Библиогр.: с. 21.

Содержат сведения о решении задачи оптимизации траекторий вспомогательных перемещений инструмента на основе теории графов. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 2,67 . Уч.-изд. л. 2,42.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.  
305040, г. Курск, ул. 50 лет Октября, 94.

## **1 Цель работы**

Получить практические навыки по оптимизации траекторий вспомогательных перемещений инструмента на основе теории графов в системе Maple.

## **2 Задание**

2.1. Изучить краткие теоретические сведения

2.2. Изучить основные функции работы с графами в системе Maple.

2.3 Создать и сохранить новый документ в системе Maple.

2.4 В соответствии с заданием построить графовую модель и на её основе сформировать в созданном документе Maple граф и задать команды для построения траектории перемещения инструмента.

2.5 Оформить отчёт

## **3. Краткие теоретические сведения**

### **3.1. Постановка задачи коммивояжера**

Машино-вспомогательное время  $t_{м.в.}$  занимает большую долю от оперативного времени затрачиваемого на выполнение технологической операции. Поэтому при комплексной оптимизации производственного процесса сокращение вспомогательного времени на холостые перемещения инструмента и смену инструмента часто дают больший эффект, чем повышение режимов резания.

Задача нахождения оптимальной траектории движения (маршрута) является одной из знаменитых задач теории комбинаторики и имеет давнюю историю. Она была известна в XIX веке и отсюда получила свое название – *задача коммивояжера*.

В классической постановке она звучала так:

«Коммивояжер (бродячий торговец) должен выйти из первого города, посетить по разу в неизвестном порядке города  $2, 1, 3 \dots n$  и вернуться в первый город. Расстояния между городами известны. В каком порядке следует обходить города, чтобы замкнутый путь коммивояжера был кратчайшим?»

В науке и технике существует большой класс задач, которых можно свести к задаче коммивояжера. В металлообработке задача поиска оптимального маршрута возникает, например, проектировании многоцелевой операции с ЧПУ, когда необходимо проводить центрование, сверление и пр. большого числа отверстий в заготовке.

Приведем задачу коммивояжера к математическому виду. Для этого введем некоторые обозначения.

Итак, имеется  $n$  точек, которые пронумерованы  $j \in T (1, 2, 3, \dots, n)$ . Обозначим расстояния между парой точек  $i, j$  через  $C_{ij}$ . Маршрут можно описать последовательностью точек  $t = (j_1, j_2, \dots, j_n, j_1)$ , причем все  $j_1 \dots j_n$  – разные номера, а повторяющийся в начале и в конце номер точки  $j_1$  показывает, что последовательность замкнута.

Таким образом, задача оптимизации заключается в нахождении такой последовательности  $t = (j_1, j_2, \dots, j_n, j_1)$ , чтобы суммарная длина всех отрезков между точками оказалась бы минимальной, т.е.

$$L = L(t) = \sum_{k=1}^n C_{j_k j_{k+1}} \rightarrow \min. \quad (1)$$

Совокупность расстояний  $C_{ij}$  образуют симметричную матрицу  $[C]$ . Каждый элемент  $C_{ij}$  матрицы  $[C]$  должен удовлетворять следующим ограничениям:

$$C_{ij} \geq 0; C_{ii} = \infty; C_{ij} + C_{jk} \geq C_{ik}. \quad (2)$$

Кажущаяся простота задачи, на первый взгляд, заключающаяся лишь в переборе всех возможных вариантов

маршрутов, на практике представляет существенную трудность, так как число комбинаций  $N$  возрастает в зависимости от числа точек в следующей зависимости:

$$N = \frac{(n-1)!}{2}. \quad (3)$$

Что означает для

$$n = 10; \rightarrow N = 181440;$$

$$n = 15; \rightarrow N = 4,36 \cdot 10^{10};$$

$$n = 20; \rightarrow N = 6,08 \cdot 10^{16}.$$

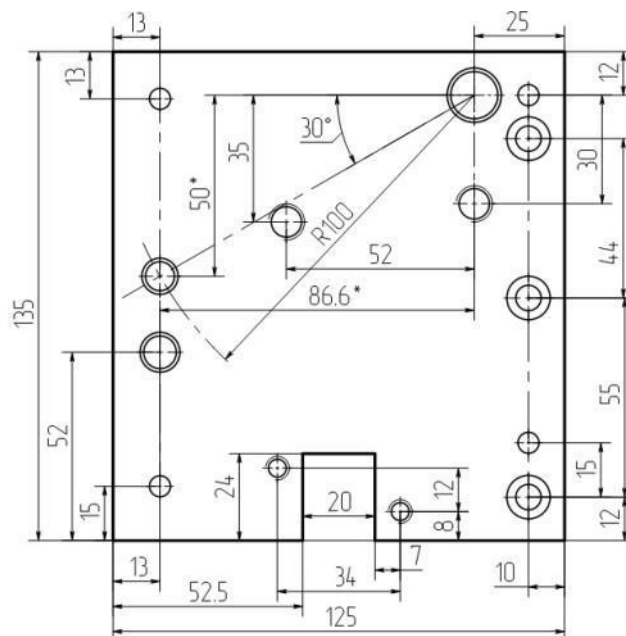
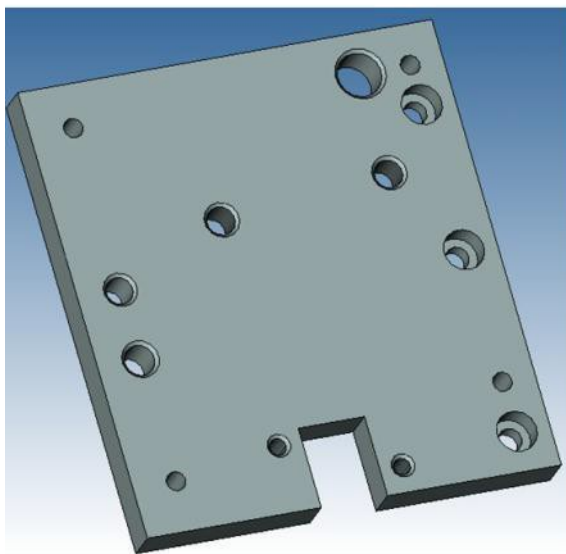
Как видно, применение даже современного мощного компьютера для такой «простой» задачи уже при  $n = 15$  практически невозможно.

Существует множество приближенных методов решения задачи коммивояжера. Рассмотрим их.

### 3.2. Методы решения задачи коммивояжера

Для того чтобы найти «хороший» (не обязательно оптимальный) маршрут задачу коммивояжера следует решать в два этапа. На первом этапе обычно находят некий базовый маршрут, а на втором этапе – его улучшают.

Рассмотрение методов решения задачи коммивояжера проведем на следующем примере.



**Рис. 1** – Эскиз детали

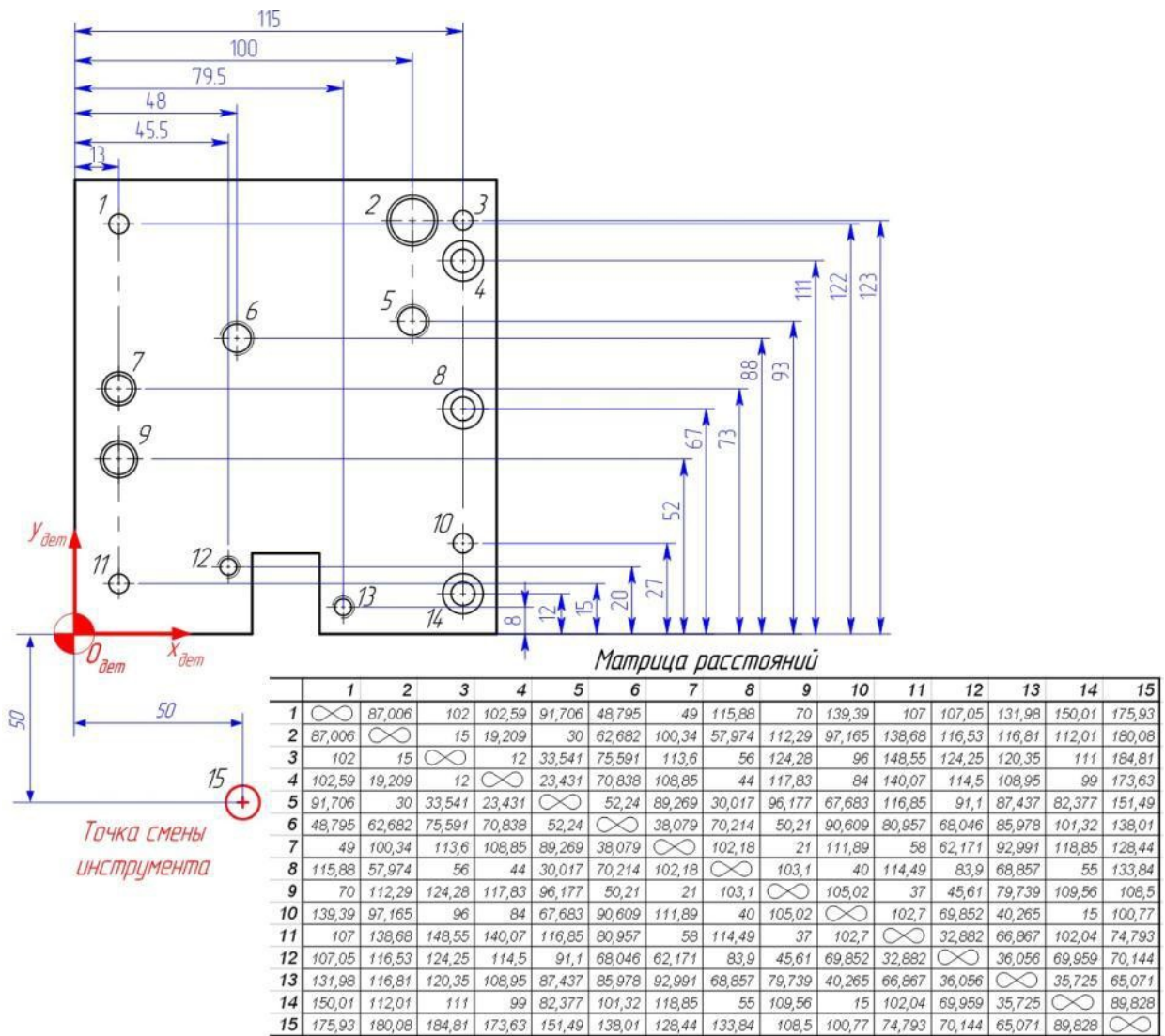
На вертикальном фрезерно-сверлильно-расточном станке производится комплексная обработка детали, показанной на рис. 1. На операции в заготовке необходимо обработать 14 различных крепежных отверстий. Первоначально все отверстия центруются. Требуется определить оптимальную последовательность зацентровки этих отверстий по критерию минимума вспомогательных перемещений. Полагается, что первоначально шпиндель находится в точке смены инструмента (т. 15), и после зацентровки 14-ти отверстий на глубину  $h = 4$  мм должен вернуться обратно в исходное положение (т. 15).

В отличие от классической задачи коммивояжера, здесь явно не задана матрица расстояний  $[C]$ , а лишь даны некоторые значения межцентровых расстояний в горизонтальном и вертикальном направлениях.

Для расчета матрицы  $[C]$  необходимо пронумеровать отверстия и изменить простановку размеров – «привязать» их к одной базе – нулю детали *Одет* (см. рис. 2). По полученным координатам  $(x_i, y_i)$  легко определить значения элементов  $C_{ij}$ , используя теорему Пифагора:

$$C_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (4)$$

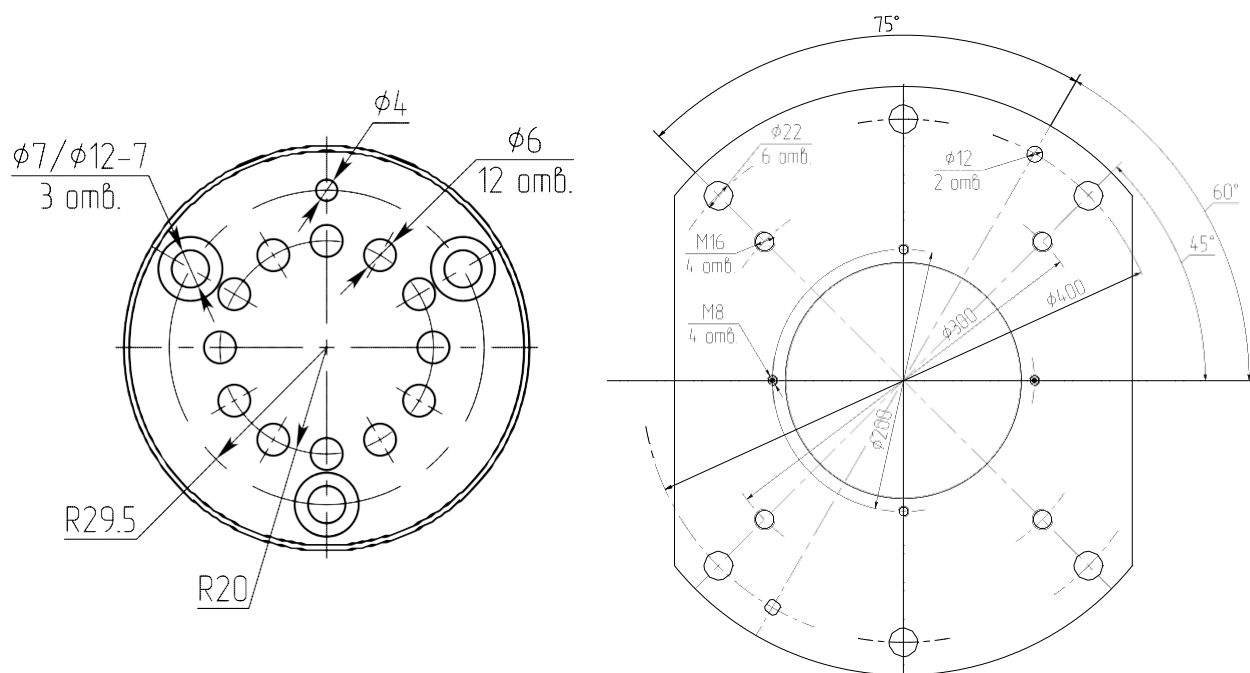




**Рис. 2** – Эскиз детали в преобразованной системе координат и матрица расстояний

Если задача коммивояжера решается для деталей (см. рис. 2.3), отверстия в которых заданы не в прямоугольных  $(x_i, y_i)$ , а в полярных координатах  $(r_i, \varphi_i)$  или  $(d_i, \varphi_i)$ , то для расстояния  $C_{ij}$  можно определять по теореме косинусов:

$$C_{ij} = \sqrt{r_i^2 + r_j^2 - 2r_i r_j \cos(\varphi_i - \varphi_j)} = \sqrt{\frac{d_i^2}{4} + \frac{d_j^2}{4} - \frac{d_i d_j}{2} \cos(\varphi_i - \varphi_j)}. \quad (5)$$



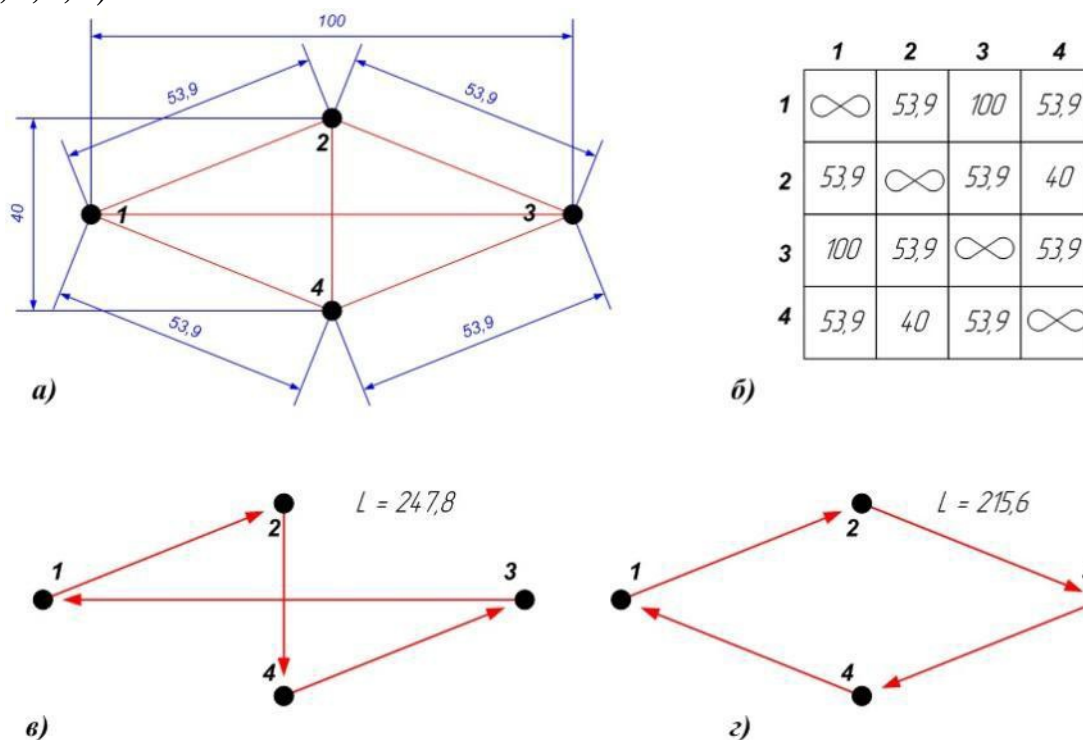
**Рис. 3** – Эскизы деталей с отверстиями, заданными в полярных координатах

### «Жадный» алгоритм

Жадный алгоритм – алгоритм нахождения наикратчайшего расстояния путём выбора самого короткого, ещё не выбранного ребра, при условии, что оно не образует цикла с уже выбранными рёбрами. «Жадным» этот алгоритм назван потому, что на последних шагах приходится жестоко расплачиваться за жадность.

Особенно ярко недостаток метода можно проиллюстрировать на примере, показанном на рис. 4. Здесь точки находятся в вершинах ромба. Выберем в качестве отправной точки – 1. Ближе всего к точке 1 лежат две точки 2 и 4, поэтому можно выбрать любую из них, например, 2. Ближе к точке 2 находится точка 4, а последней в маршрут следует включить точку 3. Однако после этого, возвращаясь в исходную точку 1, придется по самой

длинной диагонали. В результате полученный маршрут  $t = (1,2,4,3,1)$  окажется не оптимальным.



**Рис. 4** – Пример решения задачи коммивояжера  
 а) исходная схема; б) матрица расстояний;  
 в) маршрут, найденный по «жадному» алгоритму; г) оптимальный маршрут;

Если вернуться к задаче о нахождении последовательности центрования 14-ти отверстий (рис. 1) то «жадный» алгоритм даст маршрут (см. рис. 5):

$$t = (15,13,14,10,8,5,4,3,2,6,7,9,11,12,1,15),$$

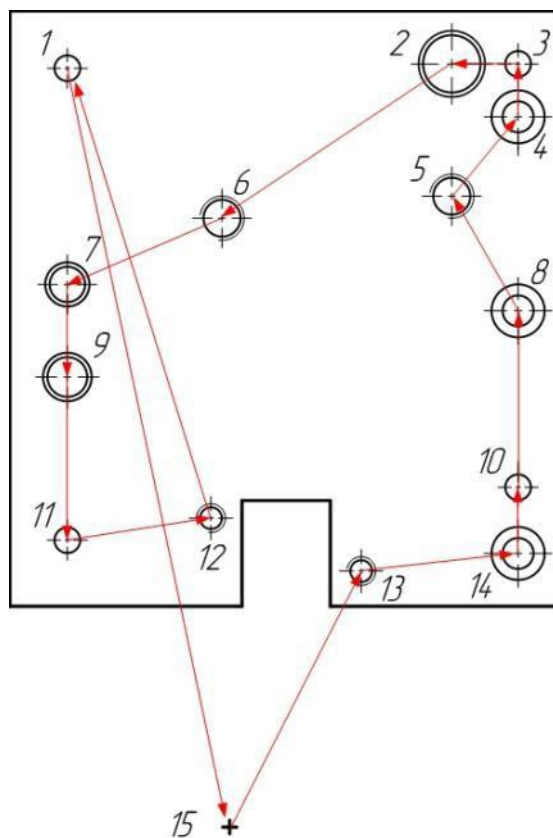
длина которого

равна:

$$L = 710,87 \text{ мм.}$$

Не трудно заметить, что найденный маршрут  $t$  не оптимален и может быть улучшен.

«Жадный» алгоритм прост, легко ложится на язык программирования, однако «ошибается» при поиске кратчайшего пути, причем ошибка может быть произвольно высока. Тем не менее, он является основным способом получения базового маршрута для последующего улучшения.



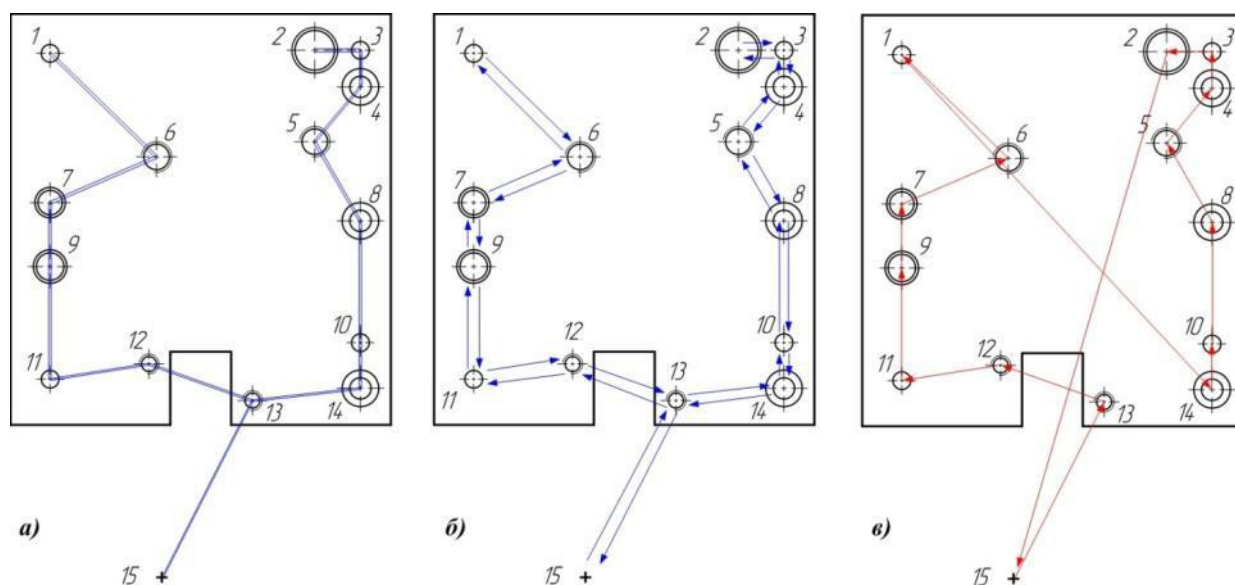
**Рис. 5** – Результат, полученный по «жадному» алгоритму

#### «Деревянный» алгоритм

Поиск маршрута по «деревянному» алгоритму выполняется в три этапа:

1. В начале строится т.н. *кратчайшее остовное дерево* с удвоенными связями между вершинами и с корнем в исходной (начальной) точке. Оно представляет граф связей с минимальными расстояниями между точками.

Для рассматриваемой задачи, остовное дерево показано на рис. 2.б,а.



**Рис. 6** – Иллюстрация этапов «деревянного» алгоритма  
**а)** минимальное остовное дерево с корнем в точке 15;  
**б)** эйлеров цикл;  
**в)** маршрут, полученный из эйлерова цикла удалением повторяющихся точек

2. Далее, используя только ветви построенного дерева, строят т.н. *эйлеров цикл* – замкнутую линию через все вершины.

Эйлеров цикл для рассматриваемой задачи показан на рис. 6,б. Его можно записать как:

ЭЦ =  
 (15,13,12,11,9,7,6,1,6,7,9,11,12,13,14,10,8,5,4,3,2,3,4,5,8,10,14,13,  
 15).

3. Для получения маршрута из эйлерова цикла исключаются все повторяющиеся вершины, кроме исходной, т.е.

(15, 13, 12, 11, 9, 7, 6, 1, ~~6~~, ~~7~~, ~~9~~, ~~11~~, ~~12~~, ~~13~~, 14, 10, 8, 5, 4, 3, 2, ~~3~~, ~~4~~, ~~5~~, ~~8~~, ~~10~~, ~~14~~, ~~13~~, 15)

В результате получим маршрут, показанный на рис. 6,в:

$$t = (15, 13, 12, 11, 9, 7, 6, 1, 14, 10, 8, 5, 4, 3, 2, 15).$$

Длина маршрута оказалась равна  $L = 744,42$  мм.

Как видно маршрут опять не оптимален. «Деревянный» алгоритм так же «ошибается». Однако в отличие от «жадного» алгоритма, длину маршрута  $L$ , находимого с помощью «деревянного» алгоритма можно оценить по отношению к длине оптимального маршрута  $L_{min}$ , с помощью соотношения:

$$L_{min} \leq L \leq 2L_{min}.$$

### Случайные методы

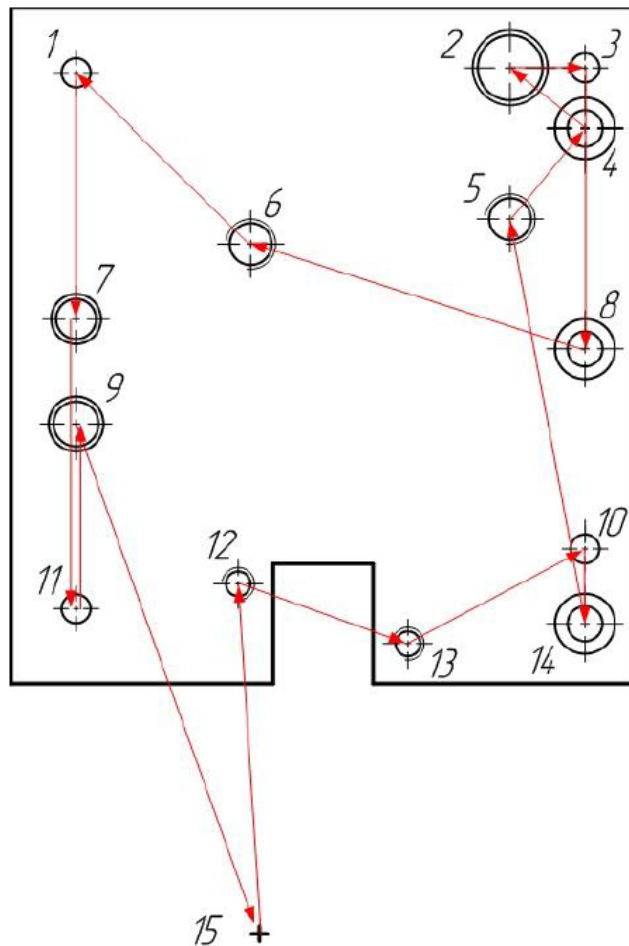
Идея состоит в том, что маршрут выбирается случайно, т.е. маршрут представляет собой случайную перестановку номеров точек. Разумеется, гарантия того, что выбранный маршрут окажется оптимальным, ничтожно мала, однако, последующая процедура улучшения маршрута, поможет существенно уменьшить его длину.

В подходе, известным под названием *метод Монте-Карло*, случайный маршрут выбирается многократно.

Так для рассматриваемой задачи общее число возможных маршрутов по формуле (2.3) будет около 44 миллиардов, и их полный перебор на персональном компьютере просто невозможен. Однако если провести статистический эксперимент случайной перестановки точек, скажем миллион раз (что приемлемо для современных компьютеров), то среди миллиона комбинаций маршрута, можно выбрать тот, который оказался лучше всех. И хотя вероятность получения оптимального маршрута также невысока (примерно, 0,003%), на практике получаемый результат оказывается «неплохим» (см. рис. 2.7):

$$t = (15, 12, 13, 10, 14, 5, 4, 2, 3, 8, 6, 1, 7, 11, 9, 15),$$

$$L = 729 \text{ мм.}$$



**Рис. 7** – Пример лучшего маршрута, найденного методом Монте-Карло при переборе 1 млн. случайных комбинаций

Существуют и другие методы поиска маршрутов – методы «ветвей и границ», алгоритм Литта, алгоритм Дейкстры и пр.

### Методы улучшения базового маршрута

#### *Метод перестановок*

Суть метода перестановок заключается в том, что после нахождения базового маршрута, делаются попытки попарно менять местами соседние точки, и если наблюдается улучшение, то такой измененный маршрут становится базовым. Процесс повторяют до тех пор, пока данный метод не перестанет давать улучшение.

Для иллюстрации рассмотрим улучшение варианта базового маршрута, полученного методом Монте-Карло (см. рис. 7).

Так, если в этом маршруте поменять местами точки 10 и 14 получим:

$$t = (15, 12, 13, 14, 10, 5, 4, 2, 3, 8, 6, 1, 7, 11, 9, 15),$$

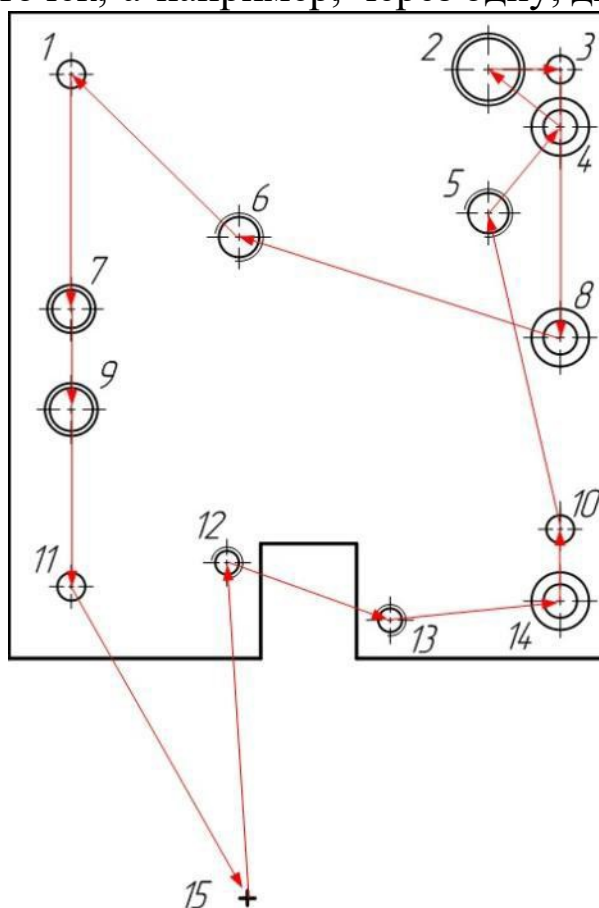
$$L = 709,76 \text{ мм.}$$

Если после этого еще поменять местами точки 11 и 9, то получим, маршрут, показанный на рис. 8:

$$t = (15, 12, 13, 14, 10, 5, 4, 2, 3, 8, 6, 1, 7, 9, 11, 15),$$

$$L = 639,05 \text{ мм.}$$

Данный способ можно продолжить, выполняя перестановки не только соседних точек, а например, через одну, две и т.д.



**Рис. 8** – Пример улучшения маршрута (см. рис. 6) перестановкой 10 и 14 точек, а также 9 и 11 точек

Метод перестановок очень эффективен и легко ложится на язык программирования.

*Метод разворота петель*

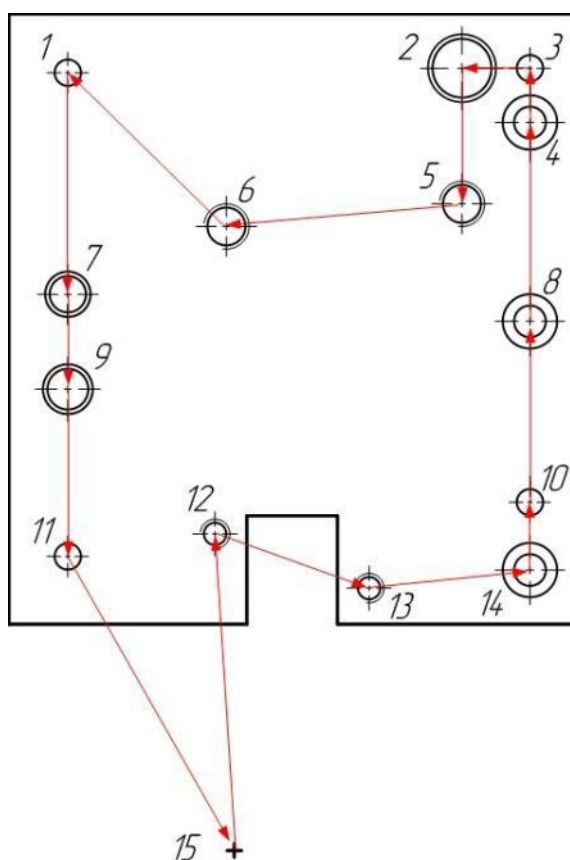


Если в найденном маршруте наблюдаются пересечения, то это свидетельствует о том, что он не оптимален. Поэтому эффективным способом улучшения маршрута является такие перестановки точек в последовательности базового варианта, чтобы маршрут не пересекался.

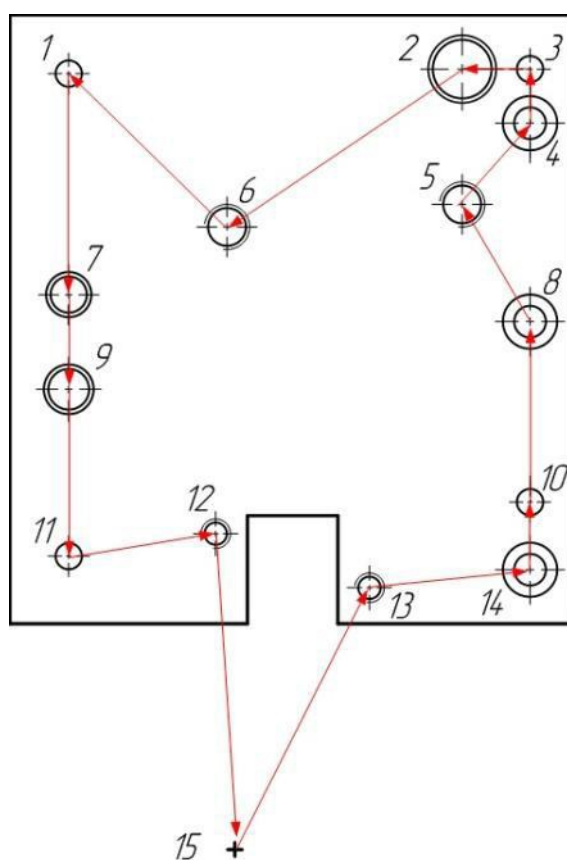
Разворачивая петлю в маршруте на рис. 8, можно получить траекторию, показанную на рис. 9:

$$t = (15, 12, 13, 14, 10, 8, 4, 3, 2, 5, 6, 1, 7, 9, 11, 15),$$

$$L = 580,75 \text{ мм.}$$



**Рис. 9** – Результат устранения пересечения в маршруте, показанном на рис. 7



**Рис. 10** – Оптимальный маршрут (оптимальная траектория вспомогательных перемещений при центровании)

### Комбинированный метод

Рассмотренные методы улучшения целесообразно применять вместе, в результате их эффективность существенно увеличивается. На рис. 10 показан маршрут, который, по-видимому, является оптимальным. Он был получен после выполнения перестановок в



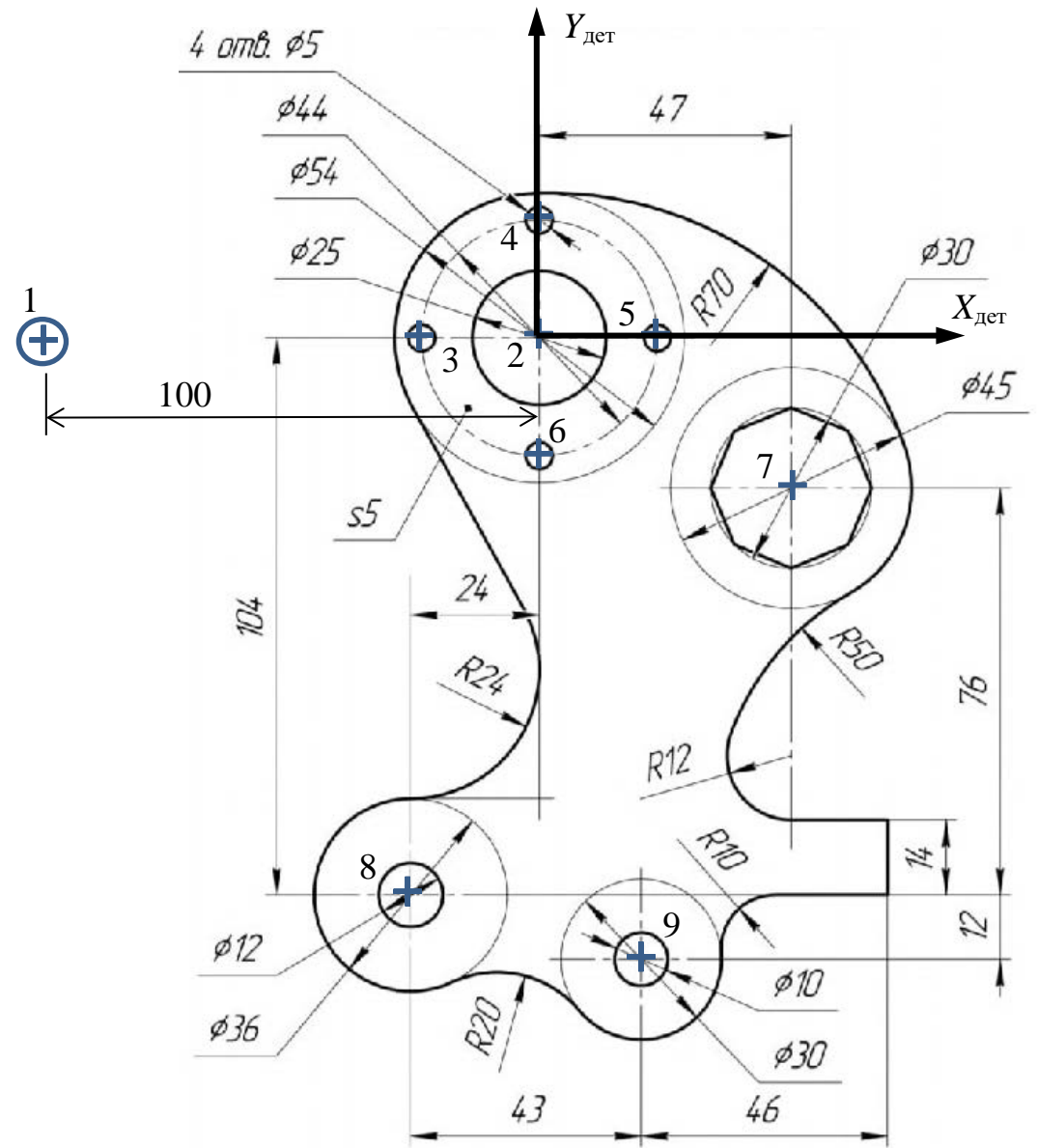


Рис. 12 - Эскиз детали в преобразованной системе координат

Таблица 1 – Координаты точек

N	X	Y
1	-100	0
2	0	0
3	-22	0
4	0	22
5	22	0
6	0	-22
7	47	-28
8	-24	-104
9	19	116

**Таблица 2 - Матрица расстояний**

	1	2	3	4	5	6	7	8	9
1	0,000	100,000	78,000	102,391	122,000	102,391	149,643	128,810	166,184
2	100,000	0,000	22,000	22,000	22,000	22,000	54,708	106,733	117,546
3	78,000	22,000	0,000	31,113	44,000	31,113	74,465	104,019	123,033
4	102,391	22,000	31,113	0,000	31,113	44,000	68,622	128,265	95,901
5	122,000	22,000	44,000	31,113	0,000	31,113	37,537	113,719	116,039
6	102,391	22,000	31,113	44,000	31,113	0,000	47,381	85,440	139,302
7	149,643	54,708	74,465	68,622	37,537	47,381	0,000	104,005	146,697
8	128,810	106,733	104,019	128,265	113,719	85,440	104,005	0,000	224,163
9	166,184	117,546	123,033	95,901	116,039	139,302	146,697	224,163	0,000

> *restart : with(GraphTheory) :*

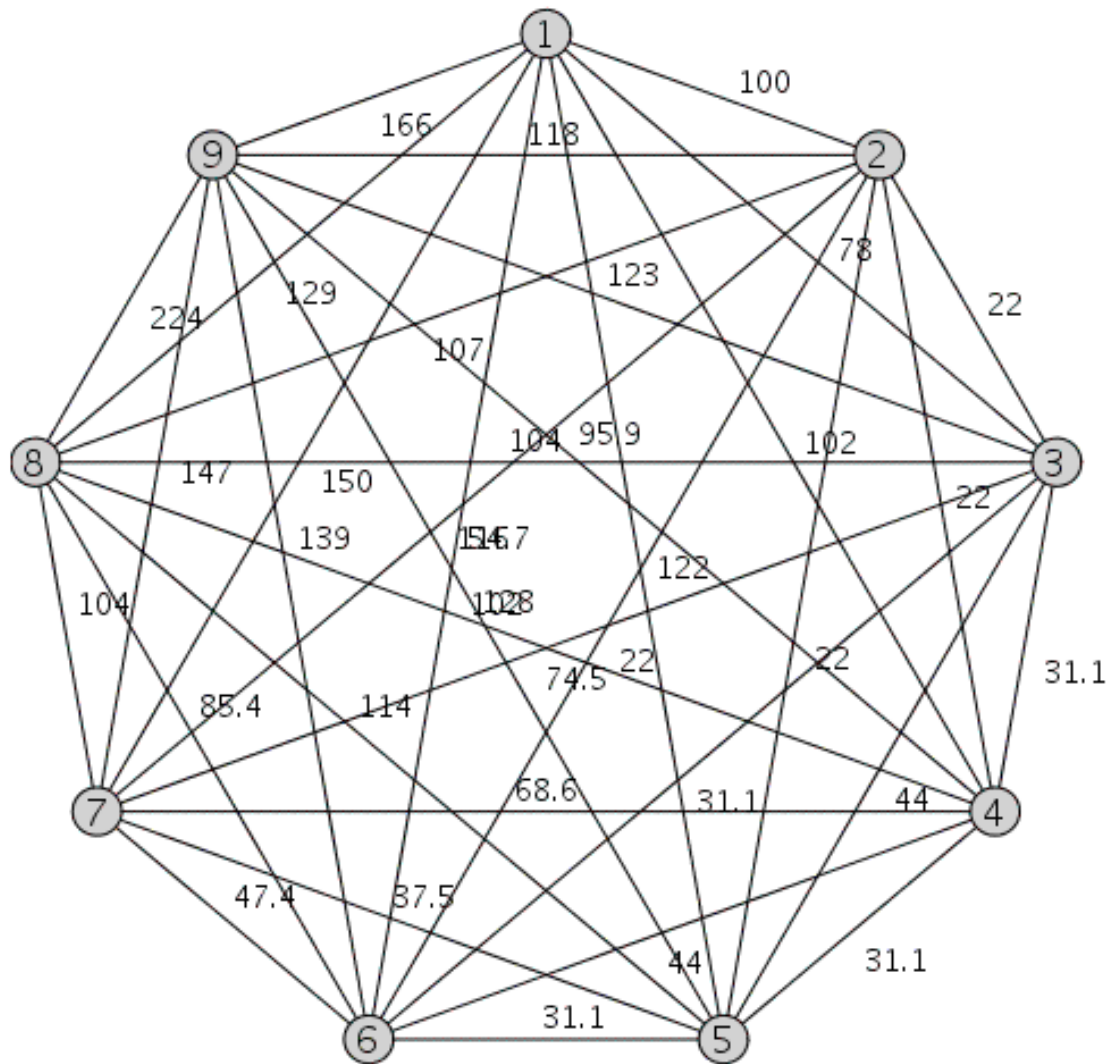
>

```
A2 := convert([[0, 100, 78, 102.3914, 122, 102.3914, 149.6429, 128.8099, 166.1836
],
[100, 0, 22, 22, 22, 22, 54.70832, 106.7333, 117.5457],
[78, 22, 0, 31.1127, 44, 31.1127, 74.46476, 104.0192, 123.0325],
[102.3914, 22, 31.1127, 0, 31.1127, 44, 68.62215, 128.2653, 95.90099],
[122, 22, 44, 31.1127, 0, 31.1127, 37.53665, 113.719, 116.0388],
[102.3914, 22, 31.1127, 44, 31.1127, 0, 47.38143, 85.44004, 139.3018],
[149.6429, 54.70832, 74.46476, 68.62215, 37.53665, 47.38143, 0, 104.0048,
146.697],
[128.8099, 106.7333, 104.0192, 128.2653, 113.719, 85.44004, 104.0048, 0,
224.1629],
[166.1836, 117.5457, 123.0325, 95.90099, 116.0388, 139.3018, 146.697,
224.1629, 0]], Matrix) :
```

> *G2 := Graph(A2)*

*G2 := Graph 1: an undirected weighted graph with 9 vertices and 36 edge(s)*

> *DrawGraph(G2)*



> *Edges(G2, weights)*

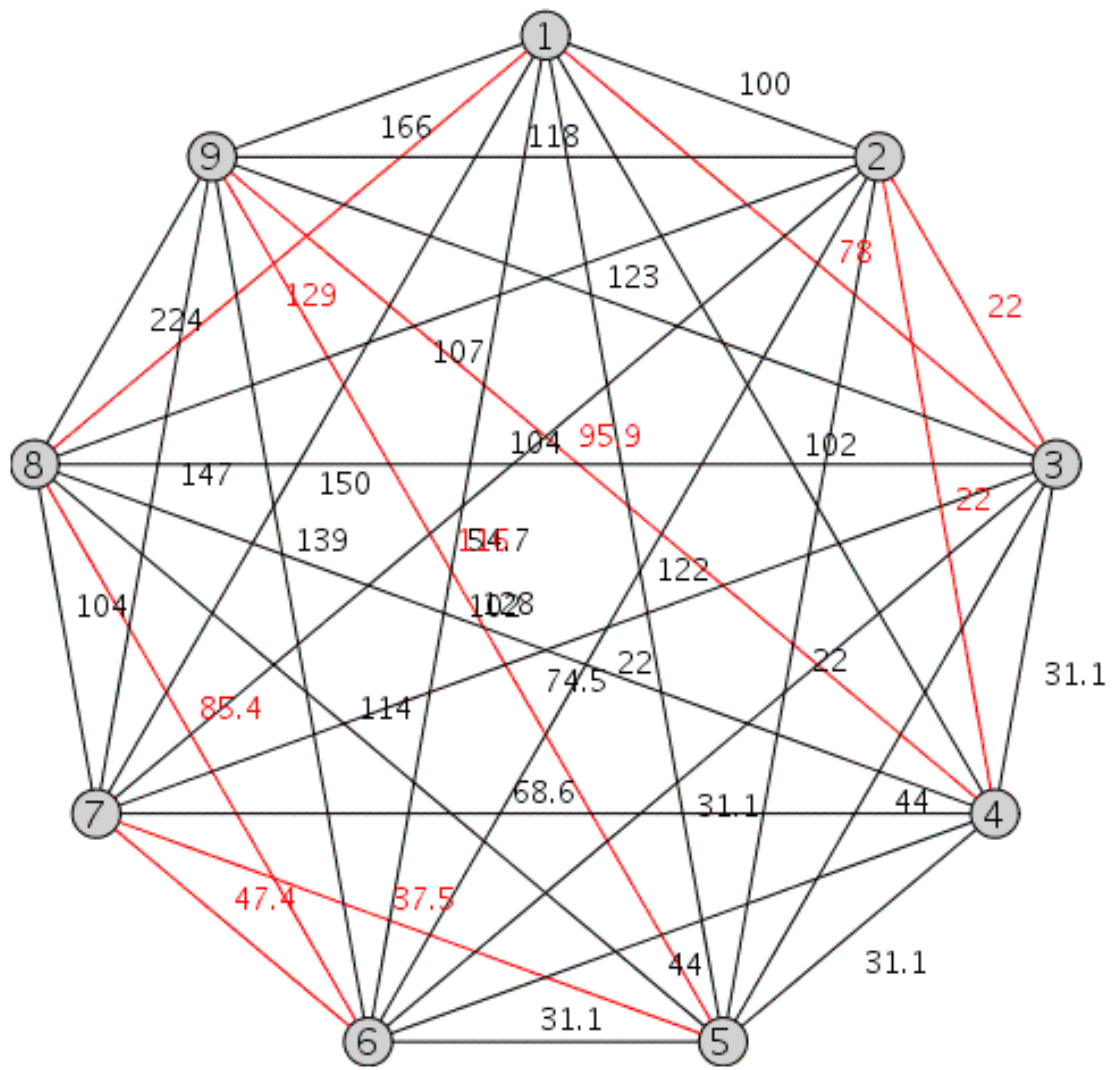
```
{[{1, 2}, 100], [{1, 3}, 78], [{1, 4}, 102.3914], [{1, 5}, 122], [{1, 6}, 102.3914],
  [{1, 7}, 149.6429], [{1, 8}, 128.8099], [{1, 9}, 166.1836], [{2, 3}, 22], [{2,
  4}, 22], [{2, 5}, 22], [{2, 6}, 22], [{2, 7}, 54.70832], [{2, 8}, 106.7333], [{2,
  9}, 117.5457], [{3, 4}, 31.1127], [{3, 5}, 44], [{3, 6}, 31.1127], [{3, 7},
  74.46476], [{3, 8}, 104.0192], [{3, 9}, 123.0325], [{4, 5}, 31.1127], [{4, 6},
  44], [{4, 7}, 68.62215], [{4, 8}, 128.2653], [{4, 9}, 95.90099], [{5, 6},
  31.1127], [{5, 7}, 37.53665], [{5, 8}, 113.719], [{5, 9}, 116.0388], [{6, 7},
  47.38143], [{6, 8}, 85.44004], [{6, 9}, 139.3018], [{7, 8}, 104.0048], [{7, 9},
  146.697], [{8, 9}, 224.1629]}
```

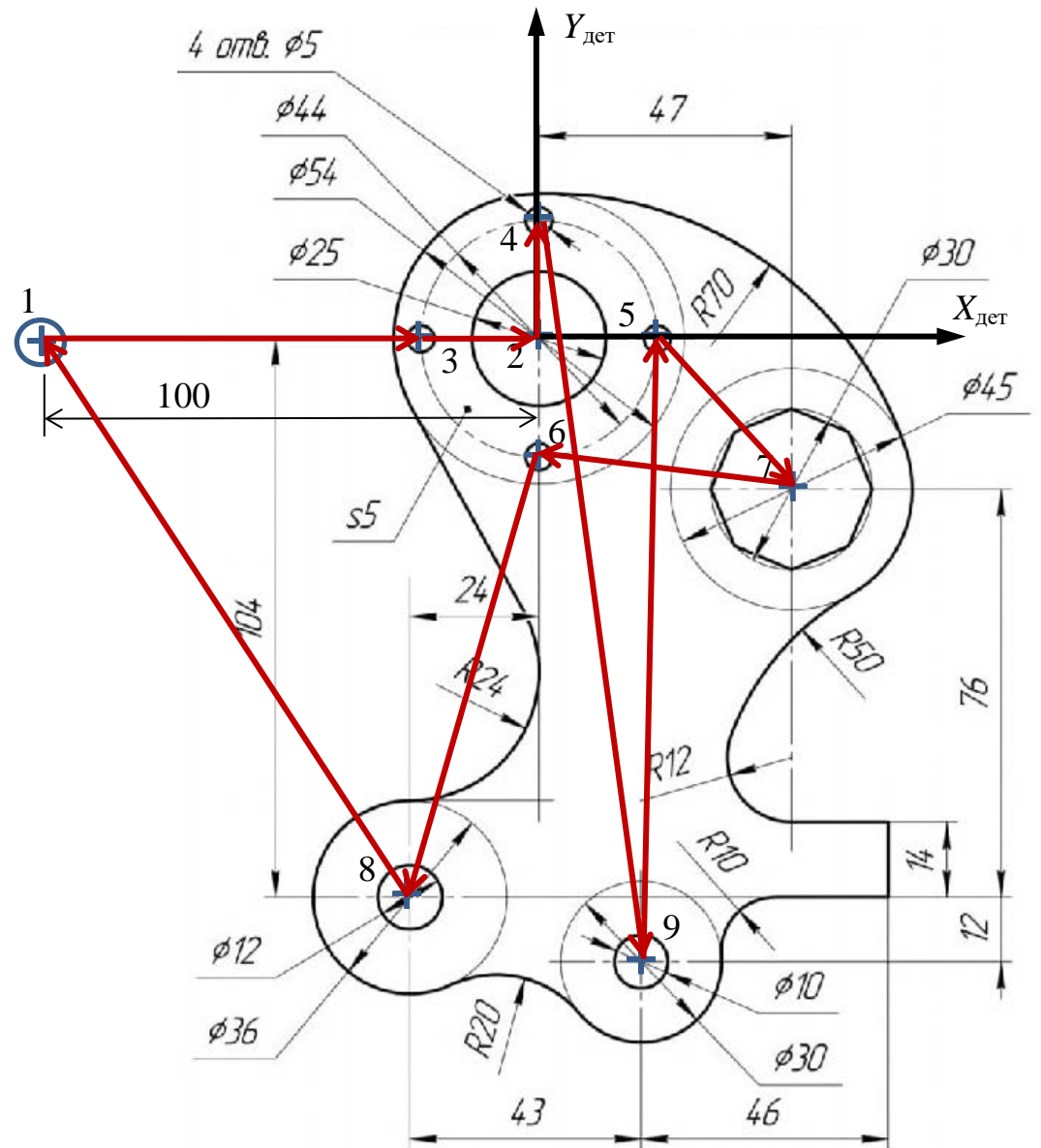
> *w, tur := TravelingSalesman(G2)*

```
w, tur := 633.10781, [1, 3, 2, 4, 9, 5, 7, 6, 8, 1]
```

> *HighlightTrail(G2, tur, red)*

> *DrawGraph(G2, style = circle)*

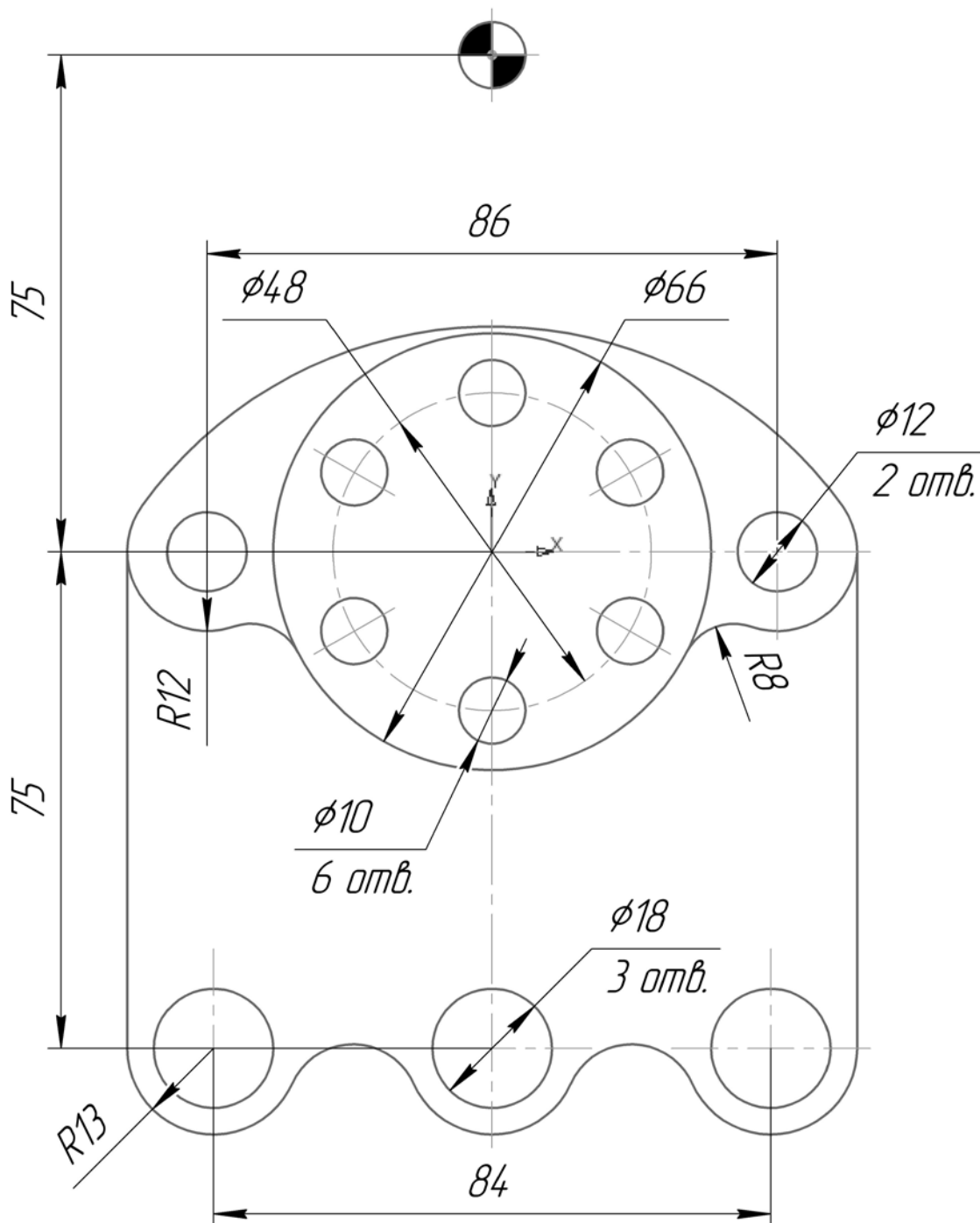




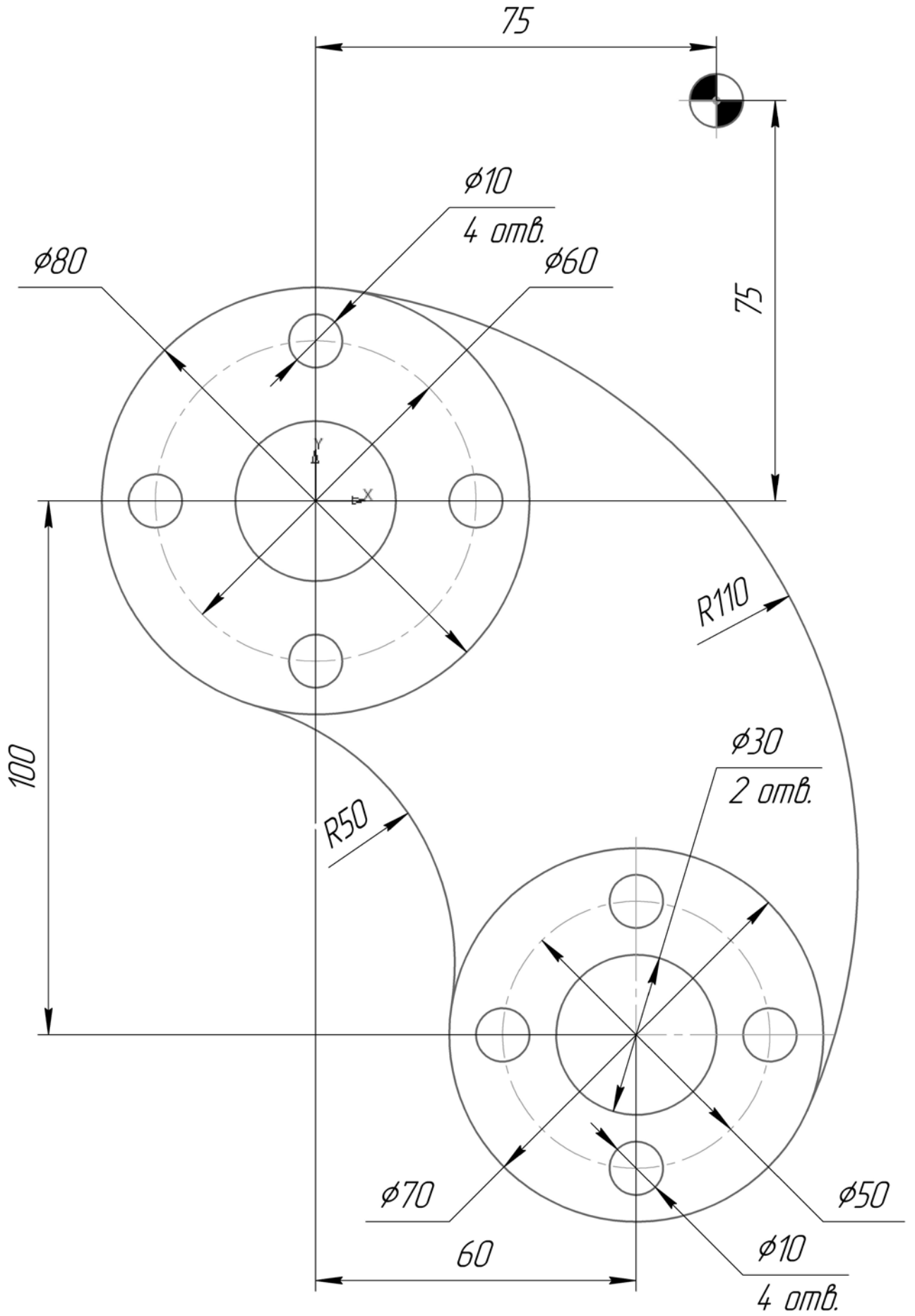
**Рис. 12 - Эскиз детали в преобразованной системе координат**

### **Библиографический список**

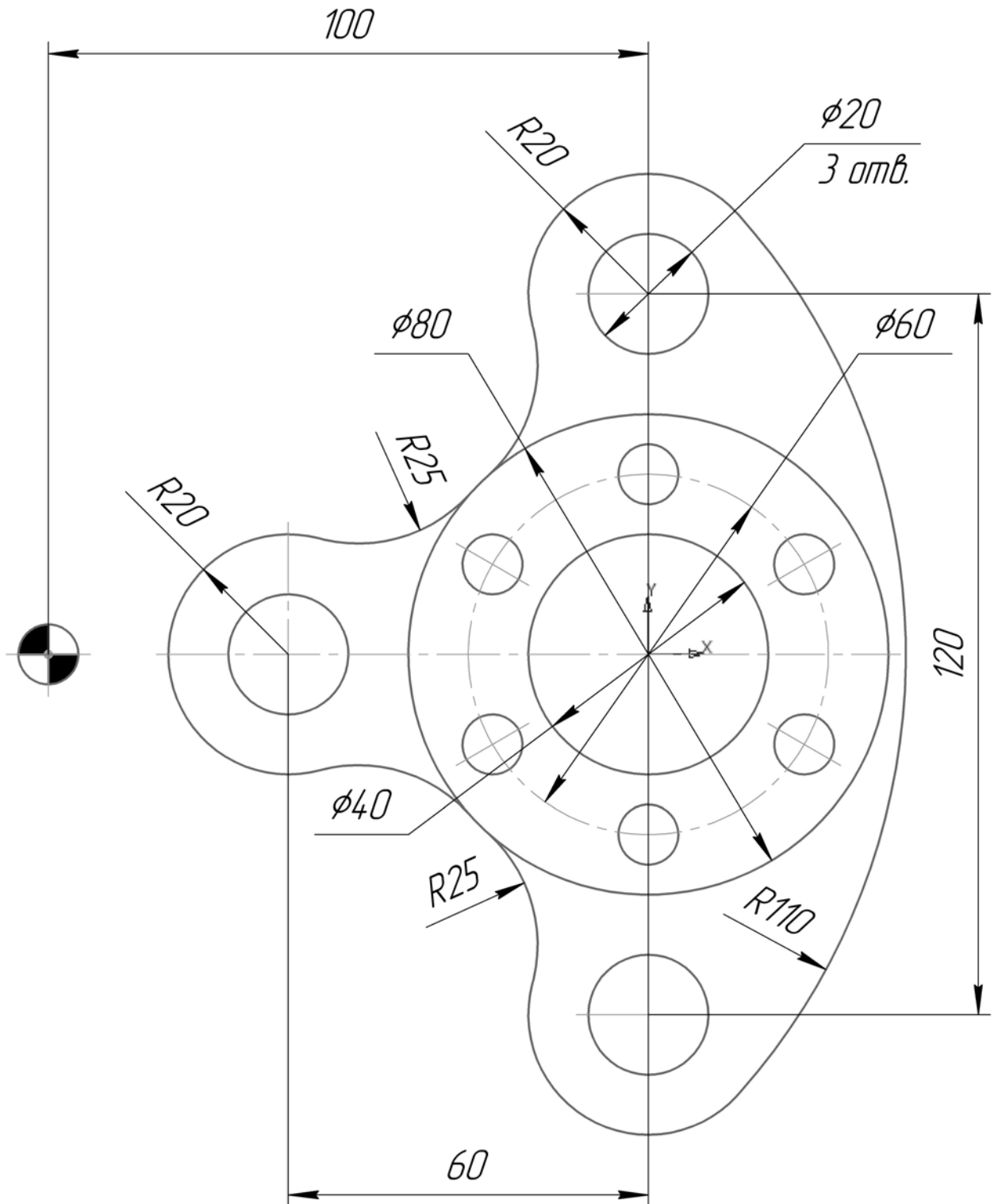
1. Гловацкая А.П. Методы и алгоритмы вычислительной математики: Учеб. пособие для вузов. – М.: Радио и связь, 1999. – 408 с.: ил.
2. Маркова Е.В., Лисенков А.Н. Комбинаторные планы в задачах много- факторного эксперимента. – М., Наука, 1979, 345 с.

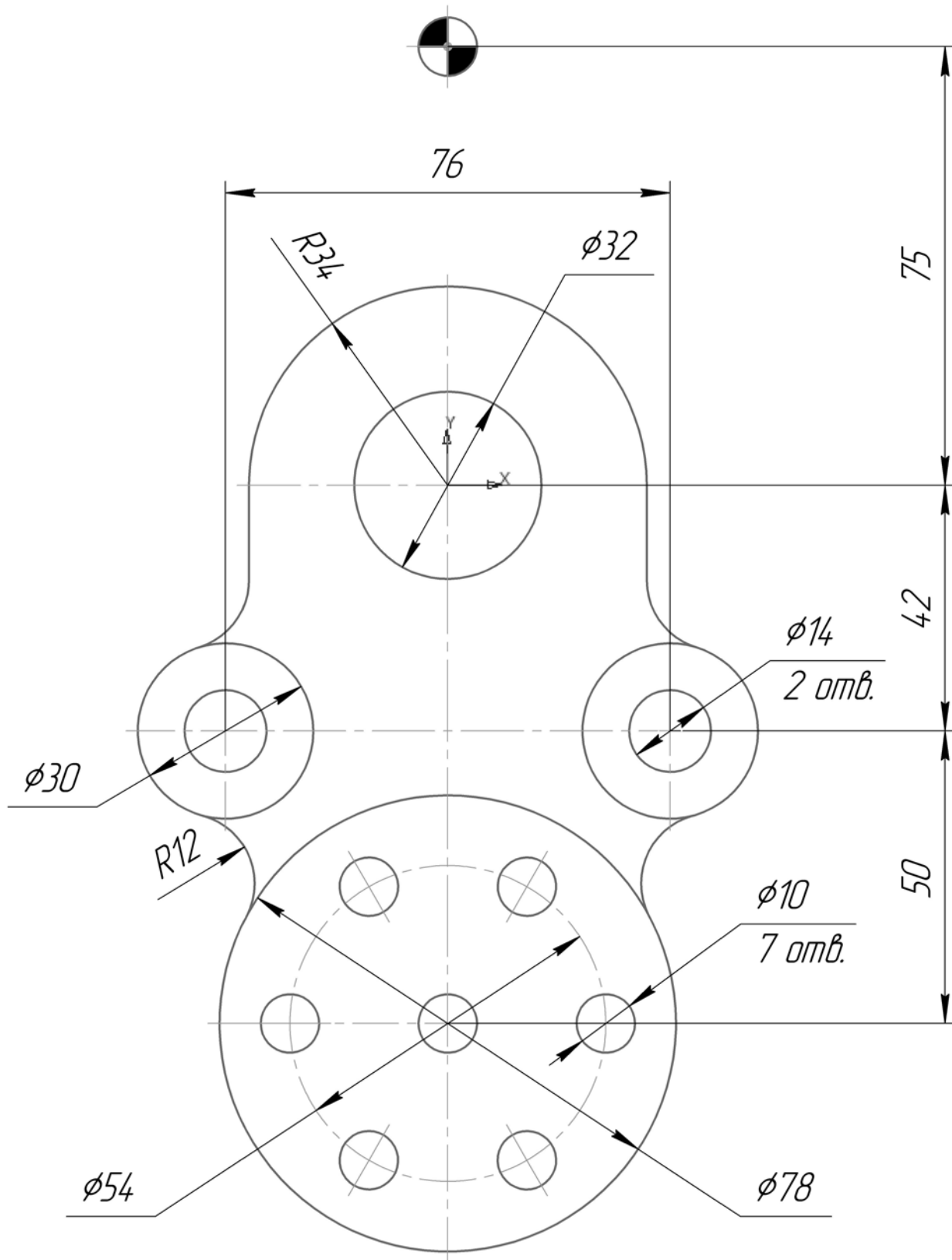


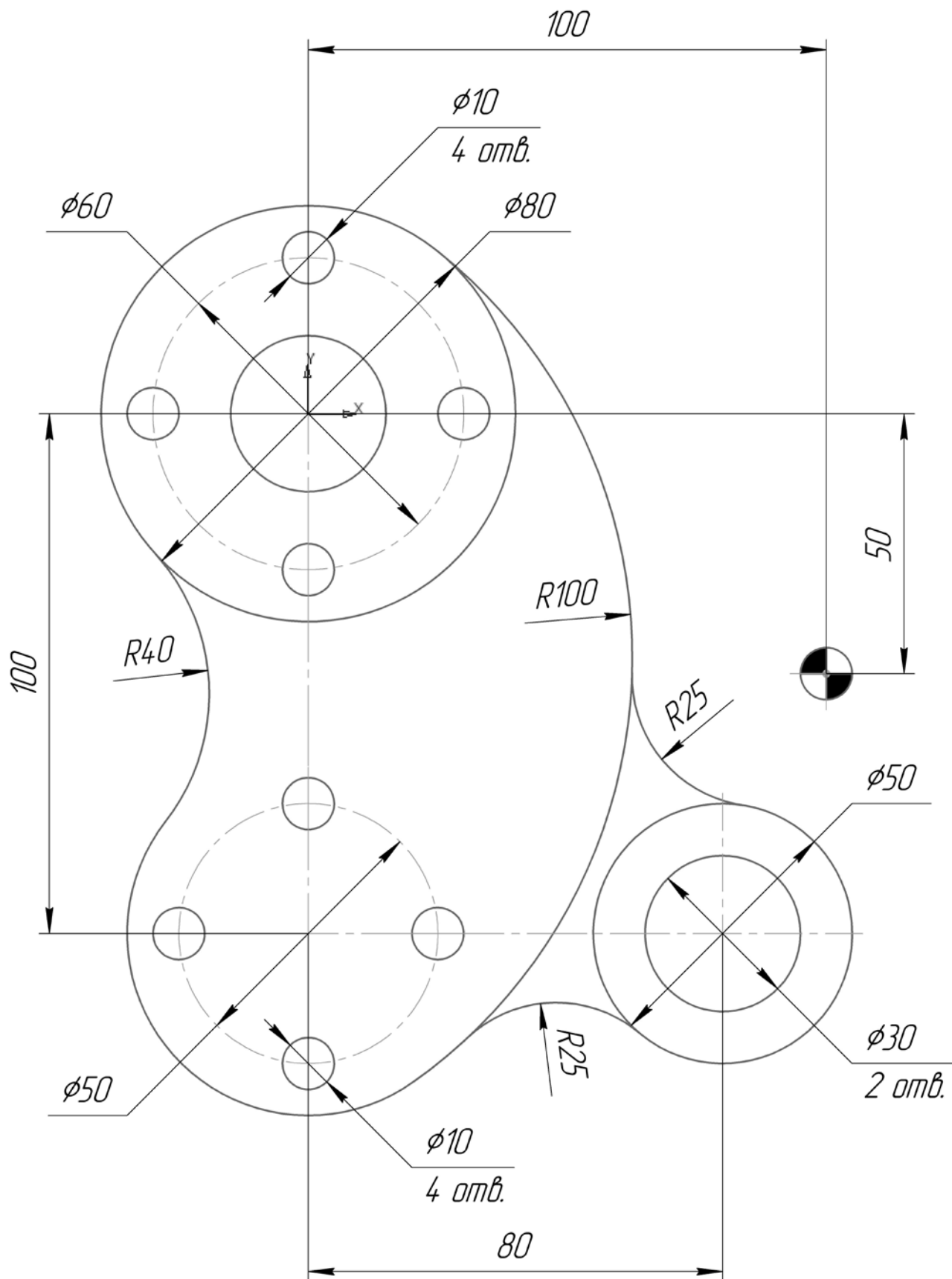




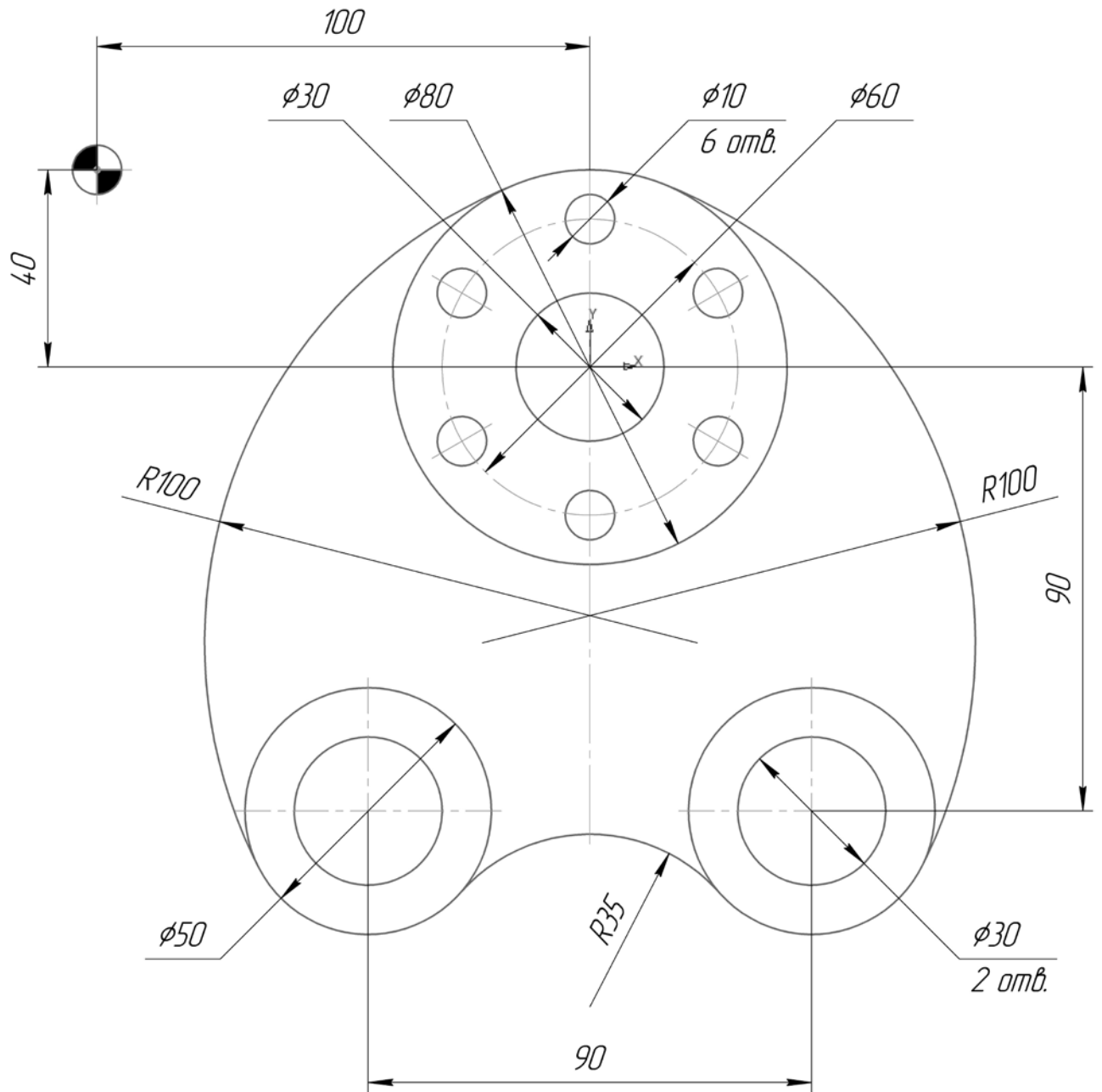
Вариант 3

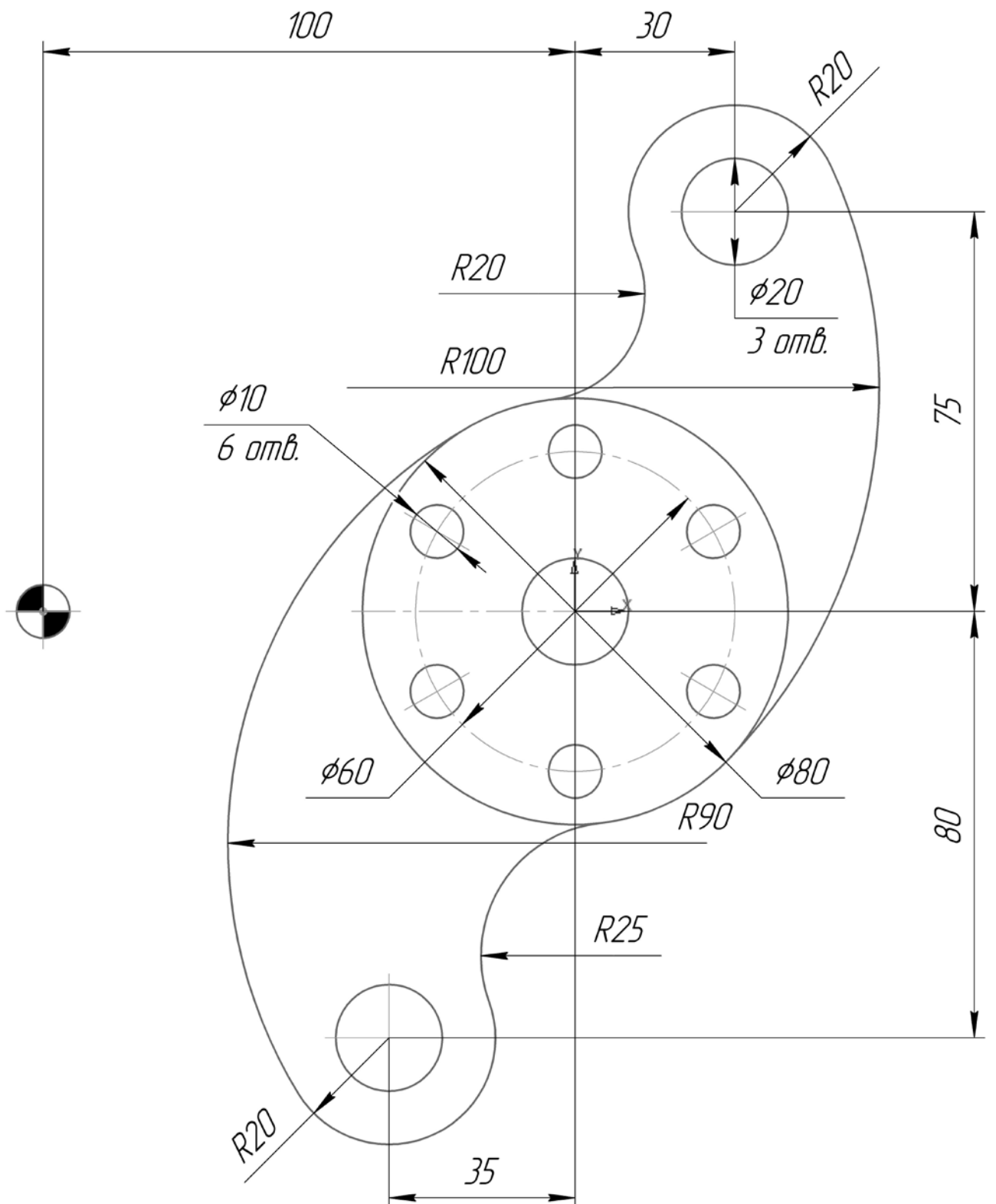




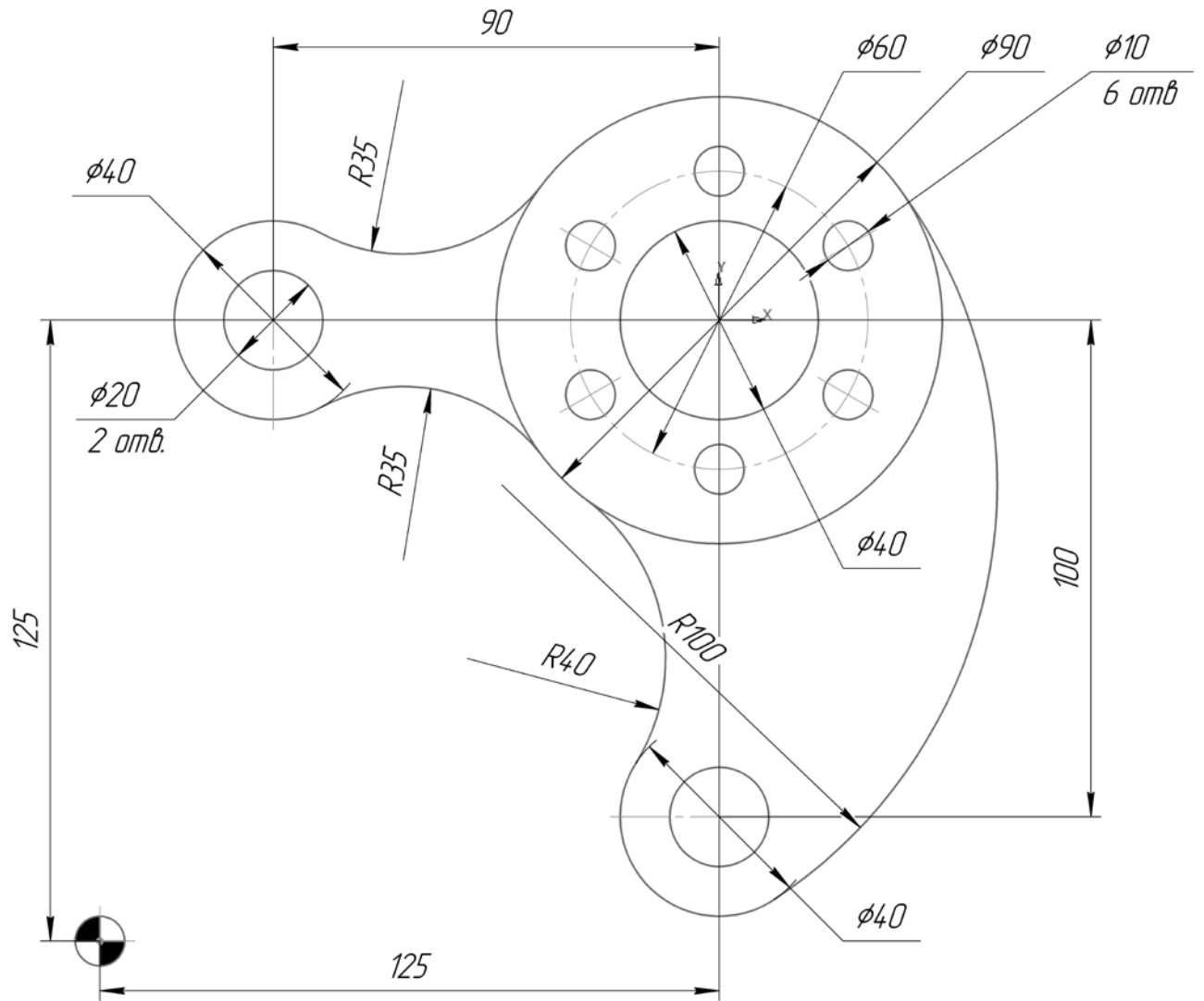


Вариант 6

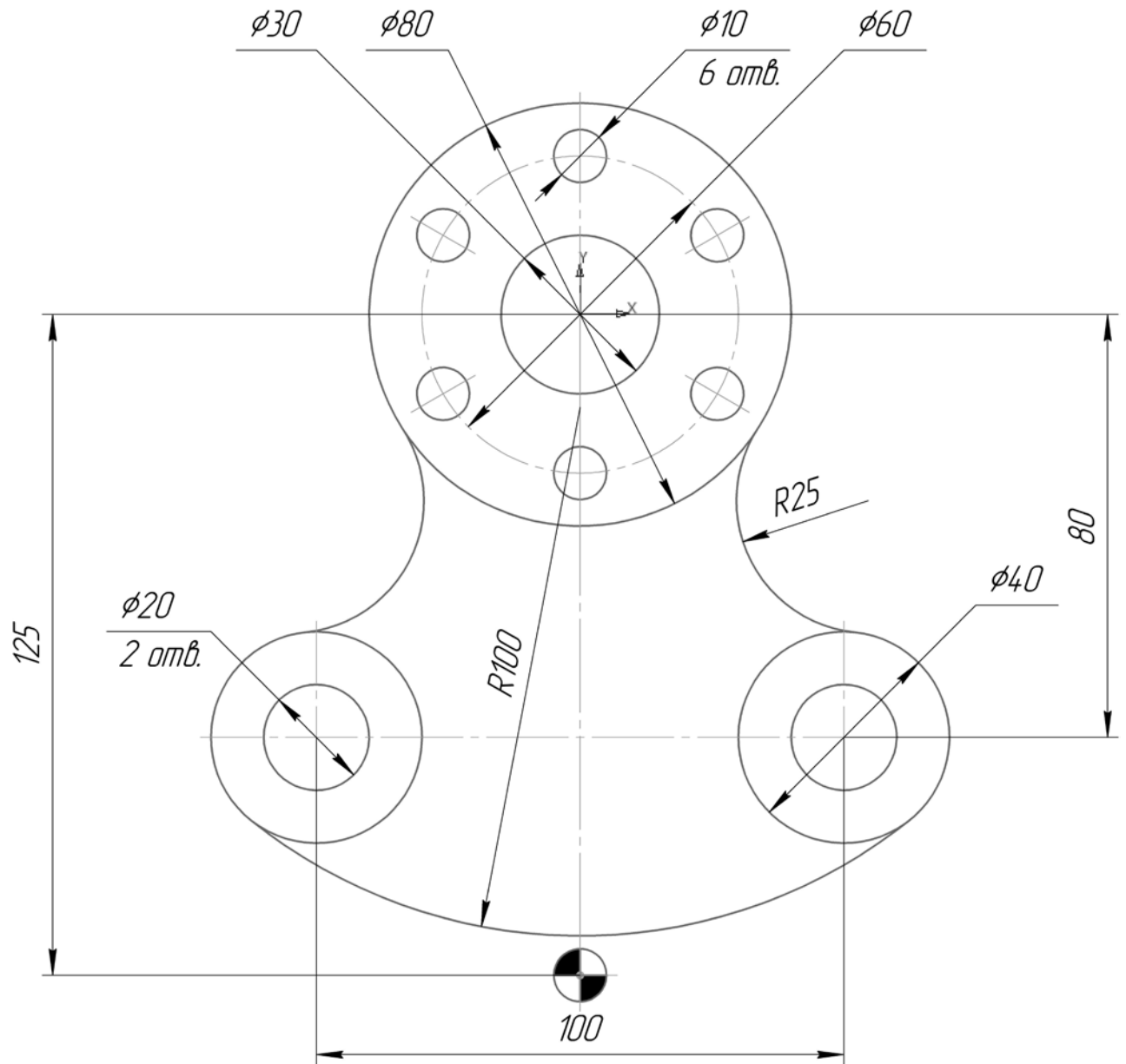




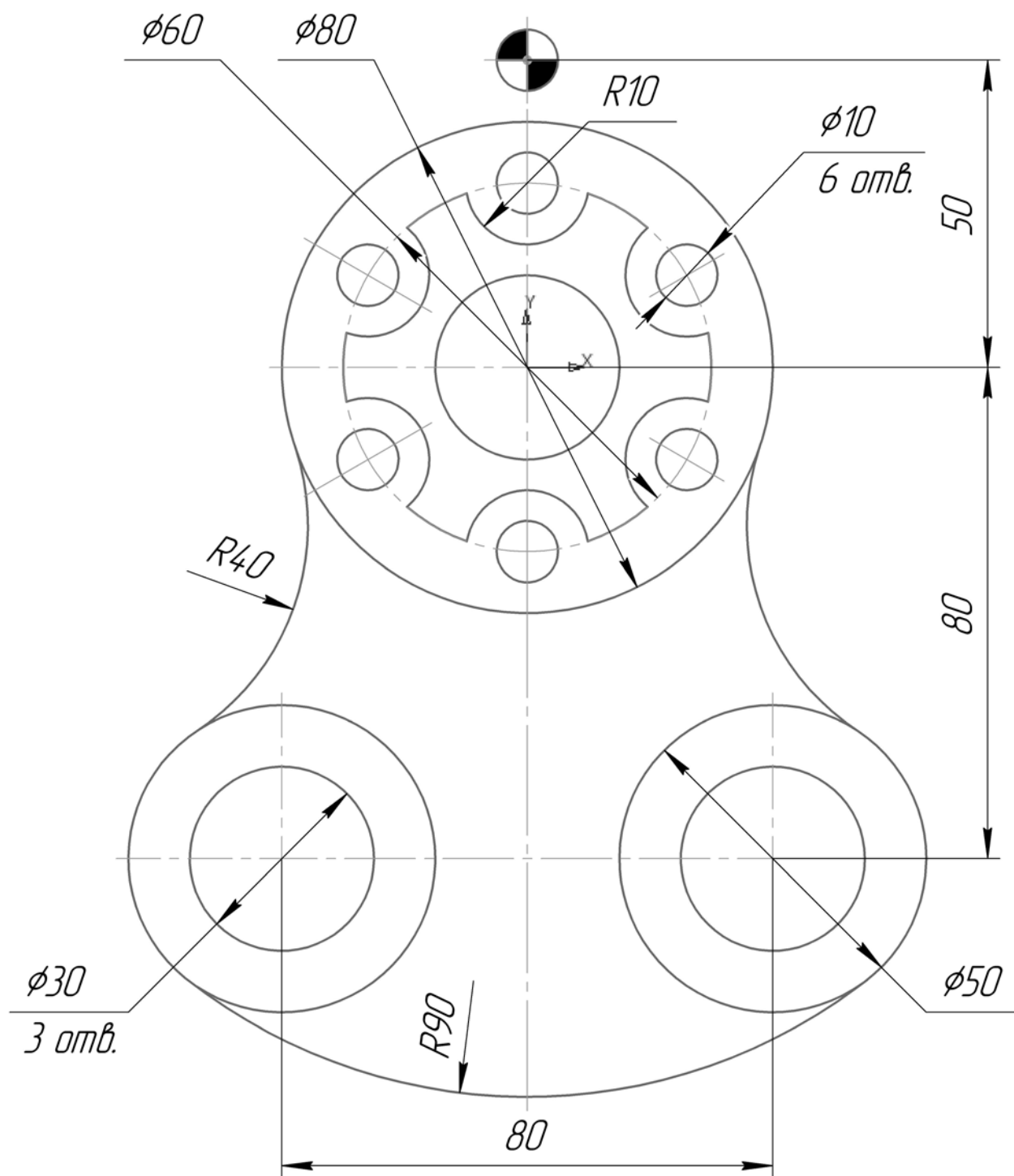
Вариант 8

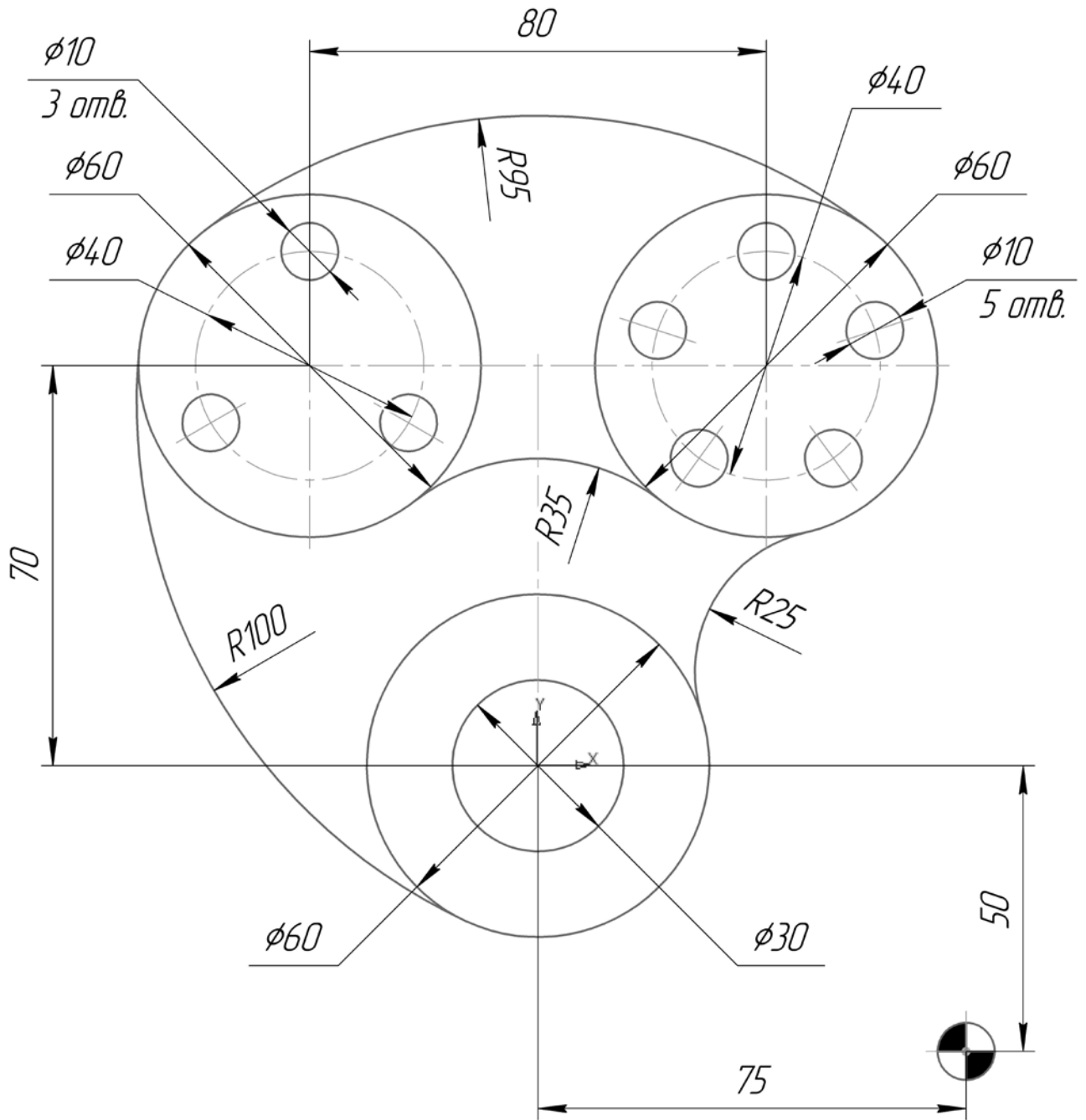


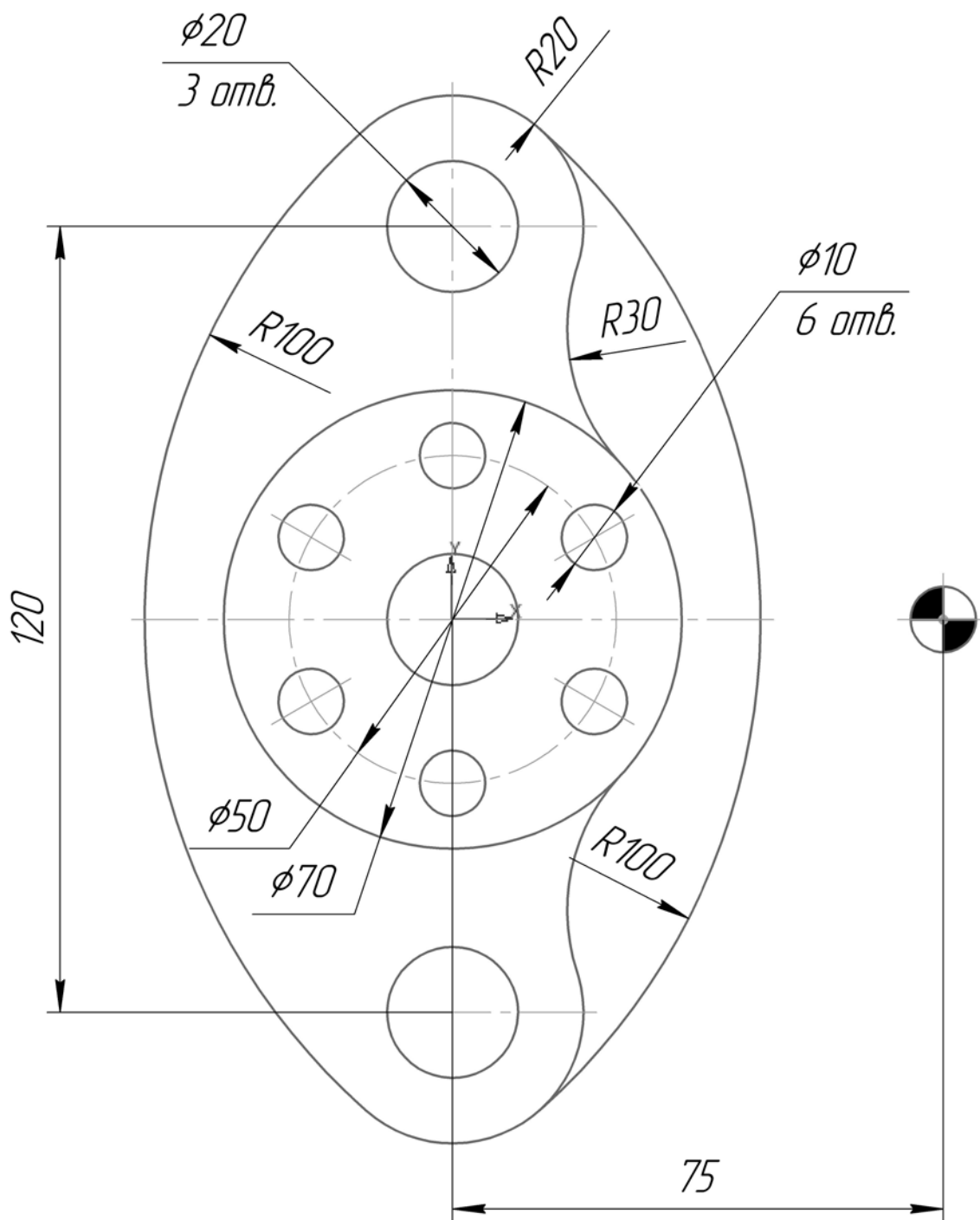
Вариант 9

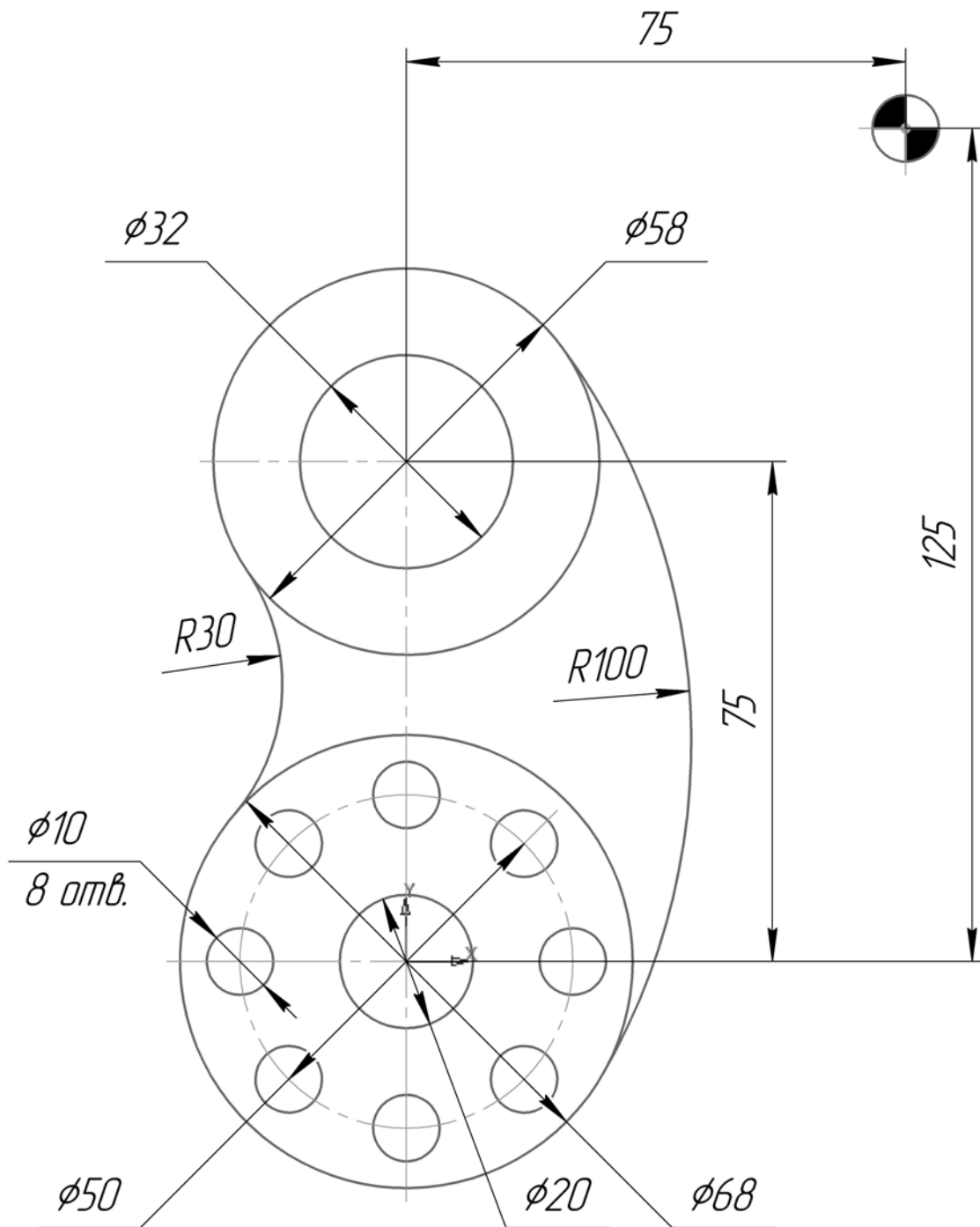


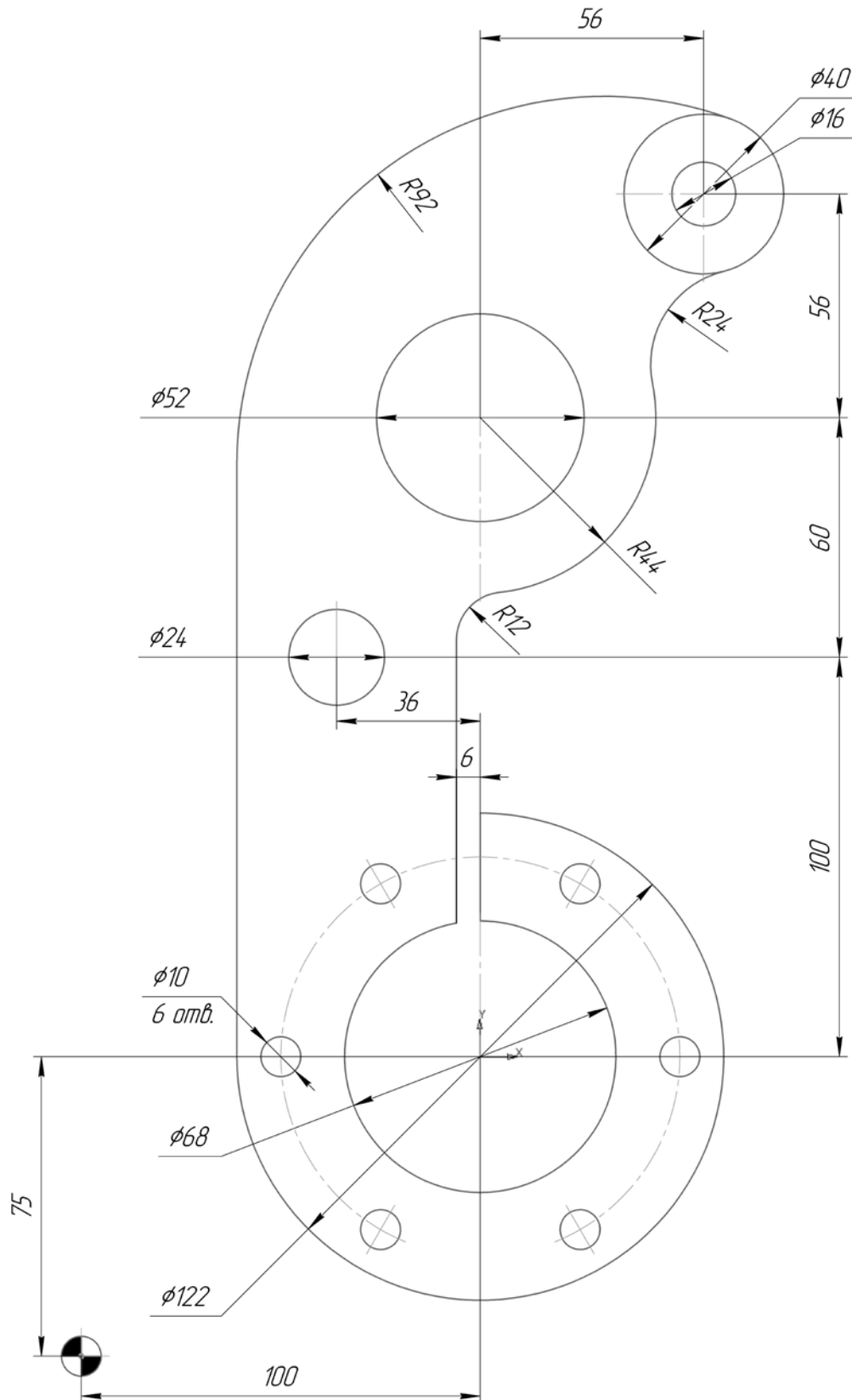


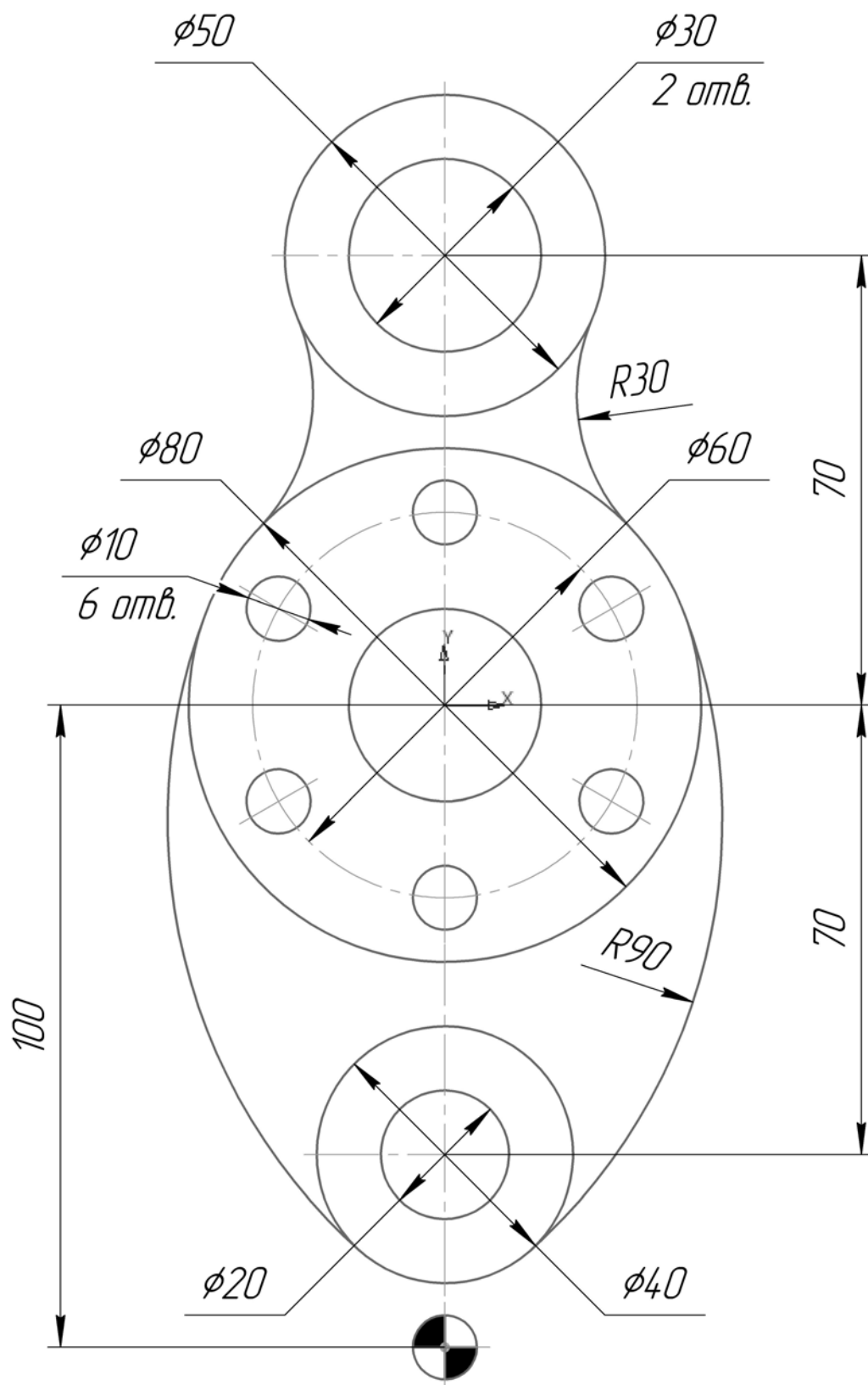


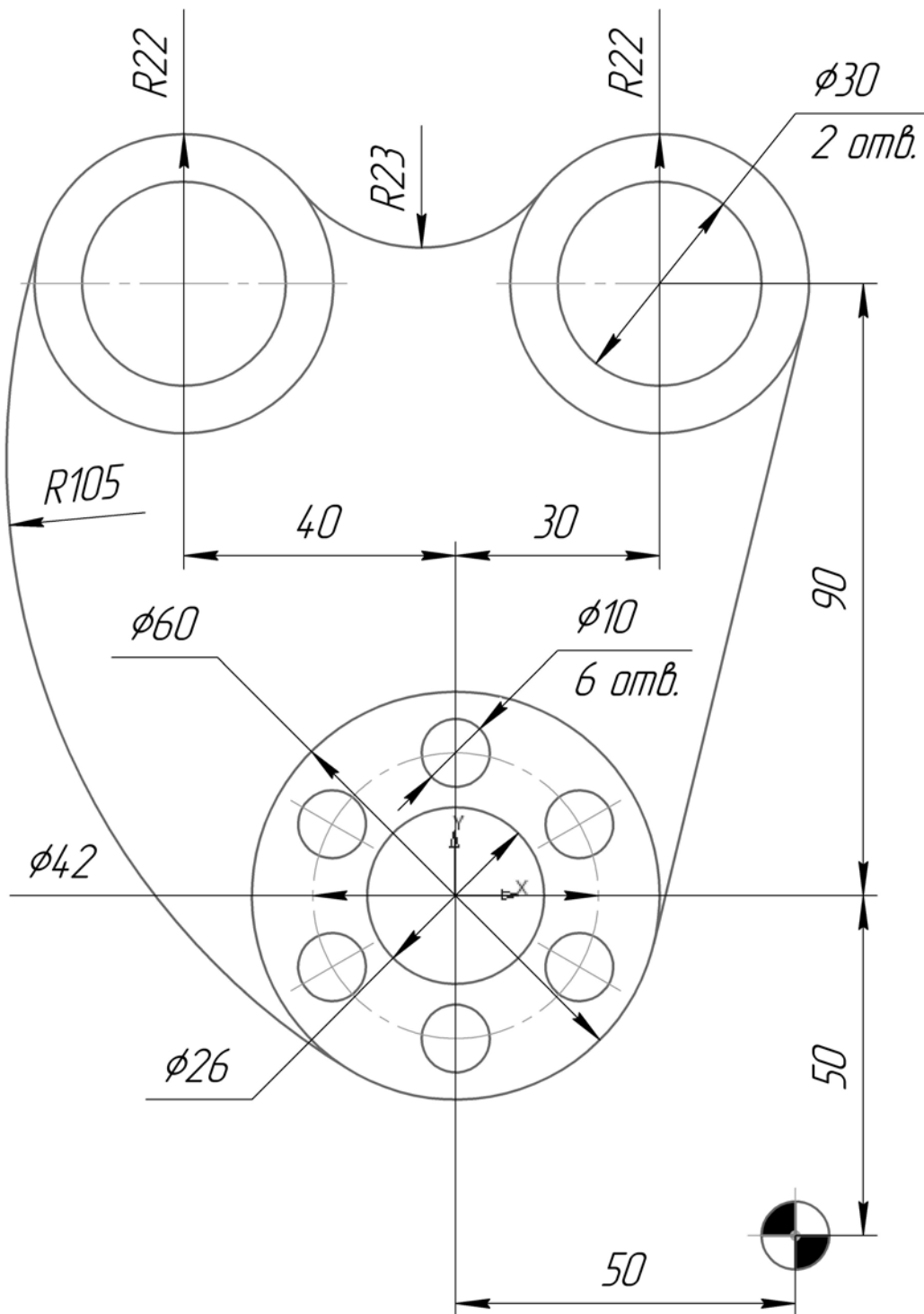


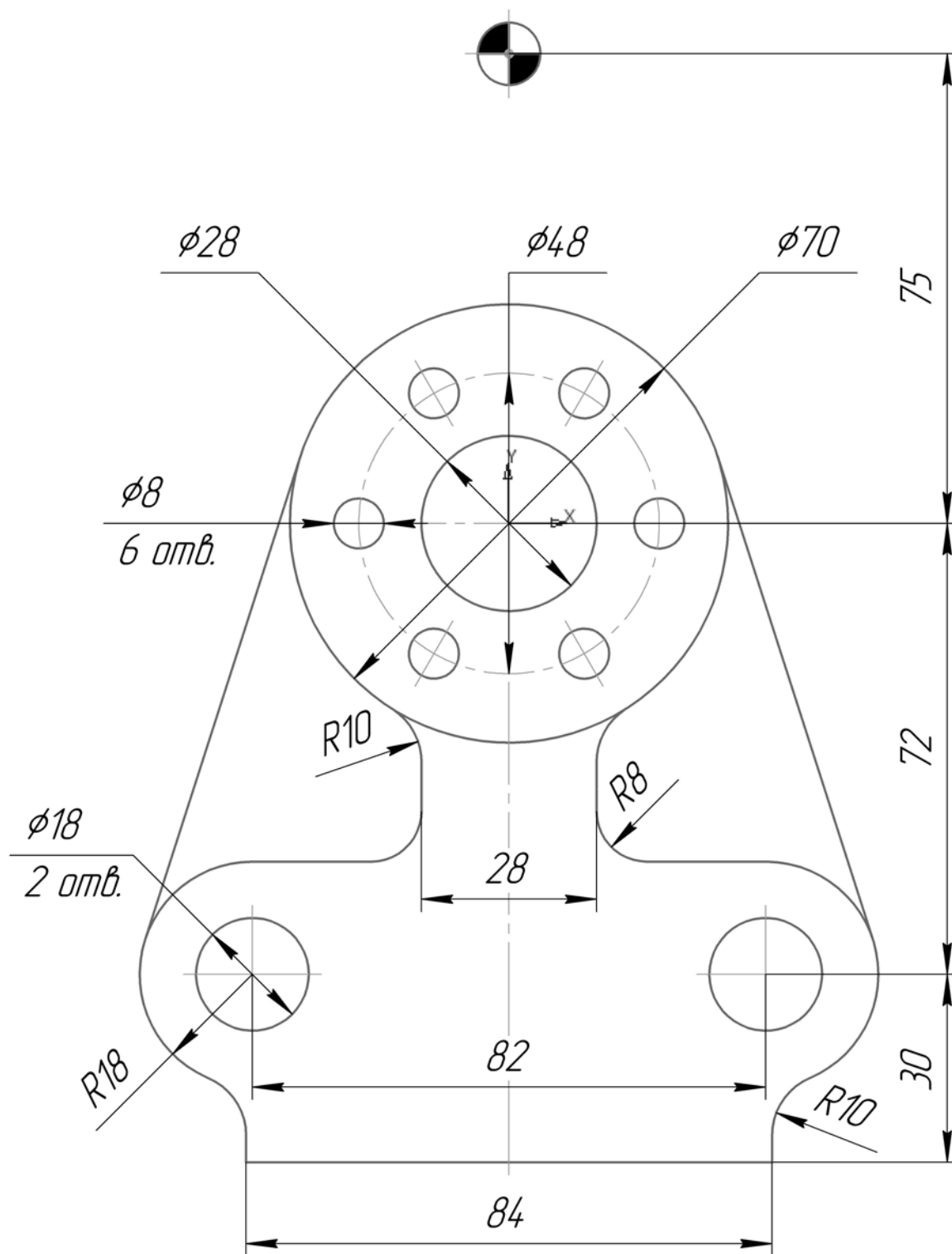




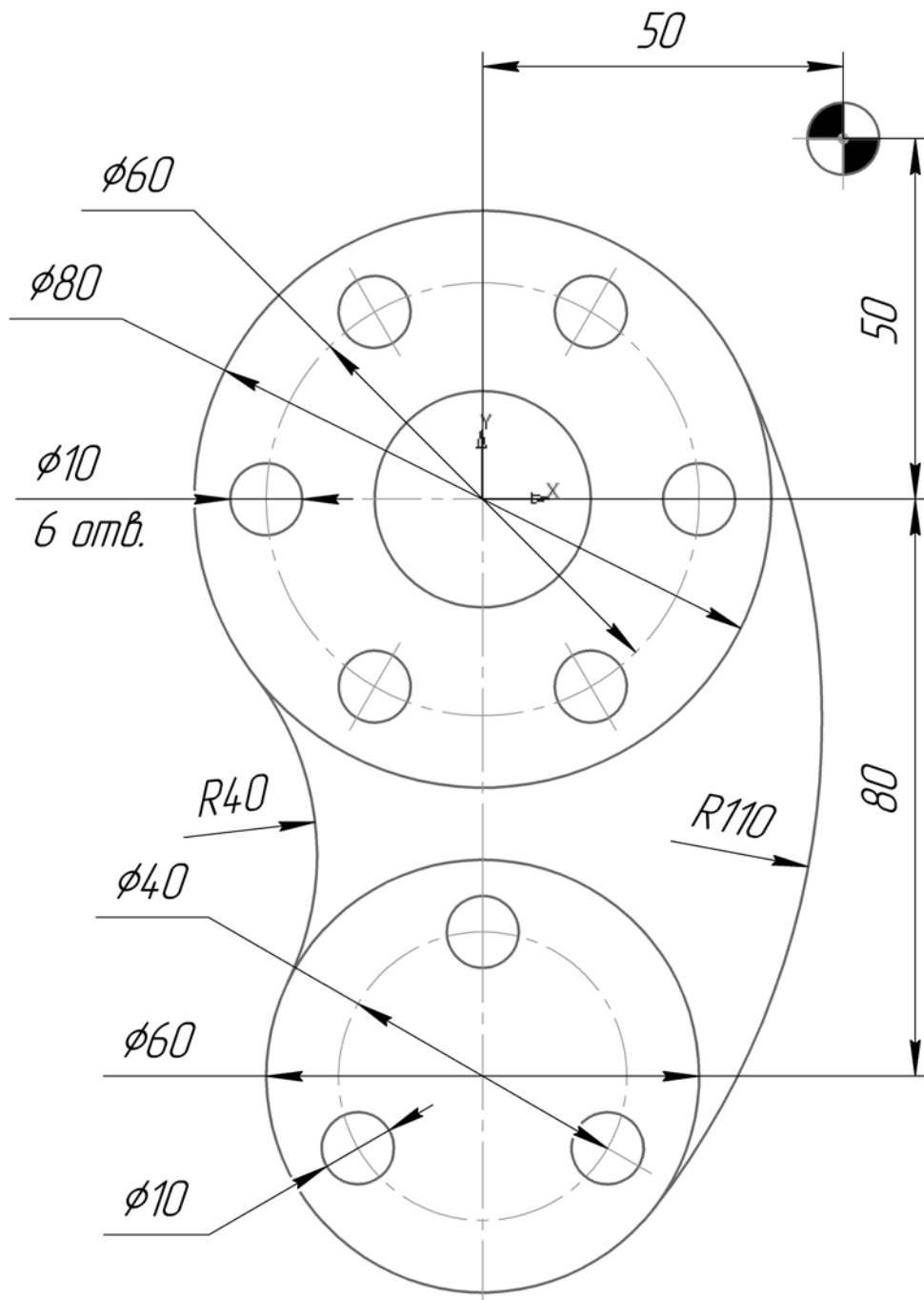


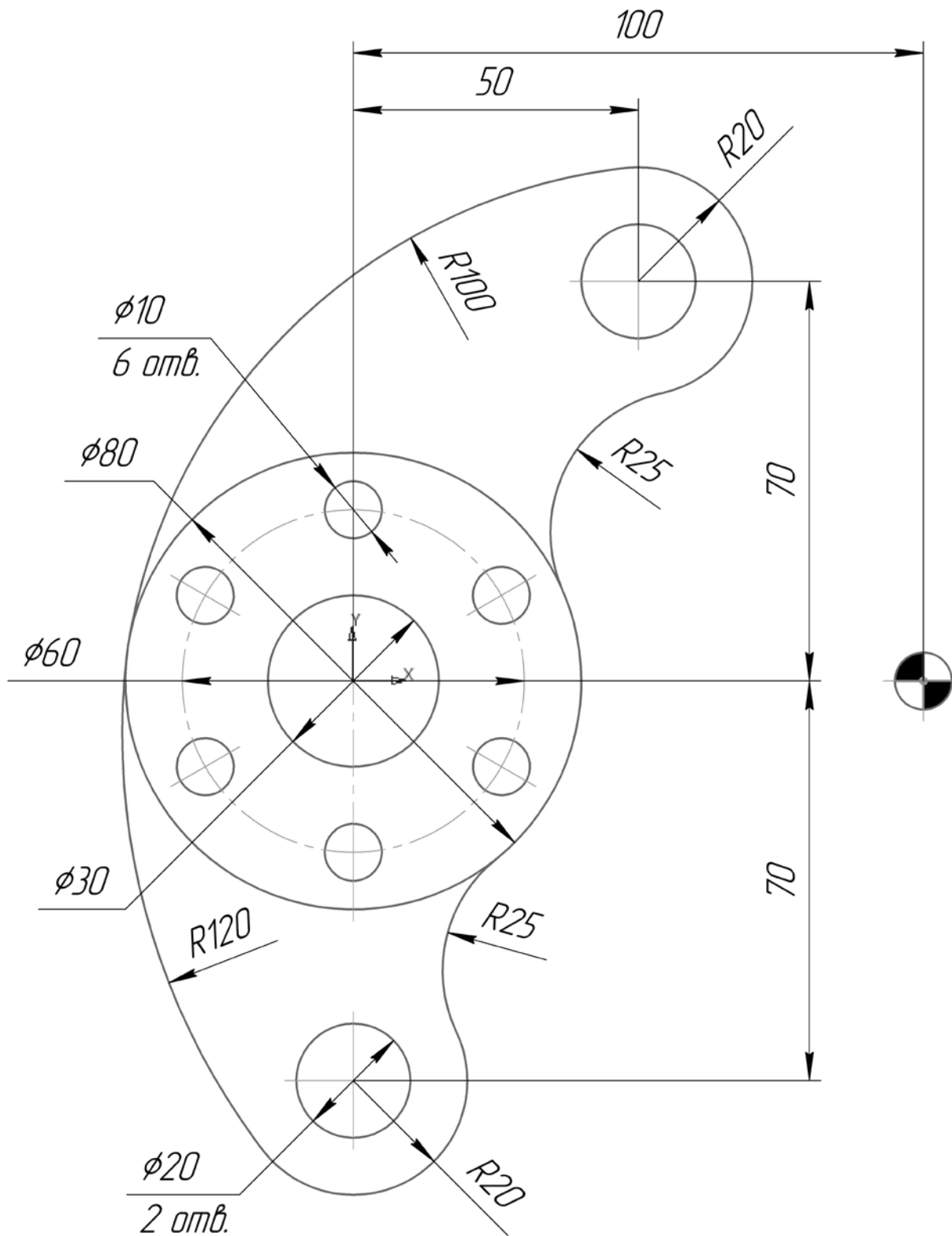


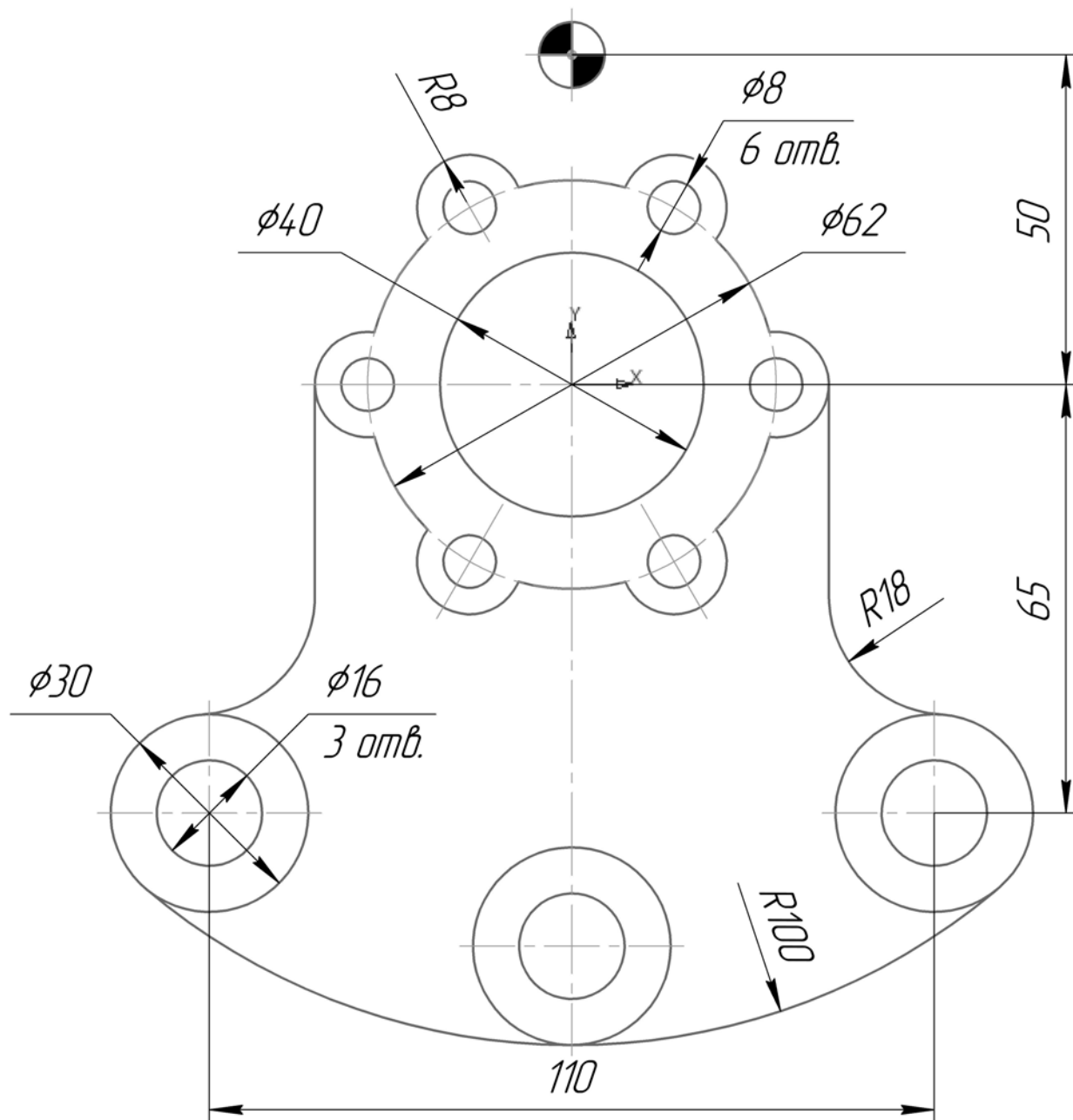


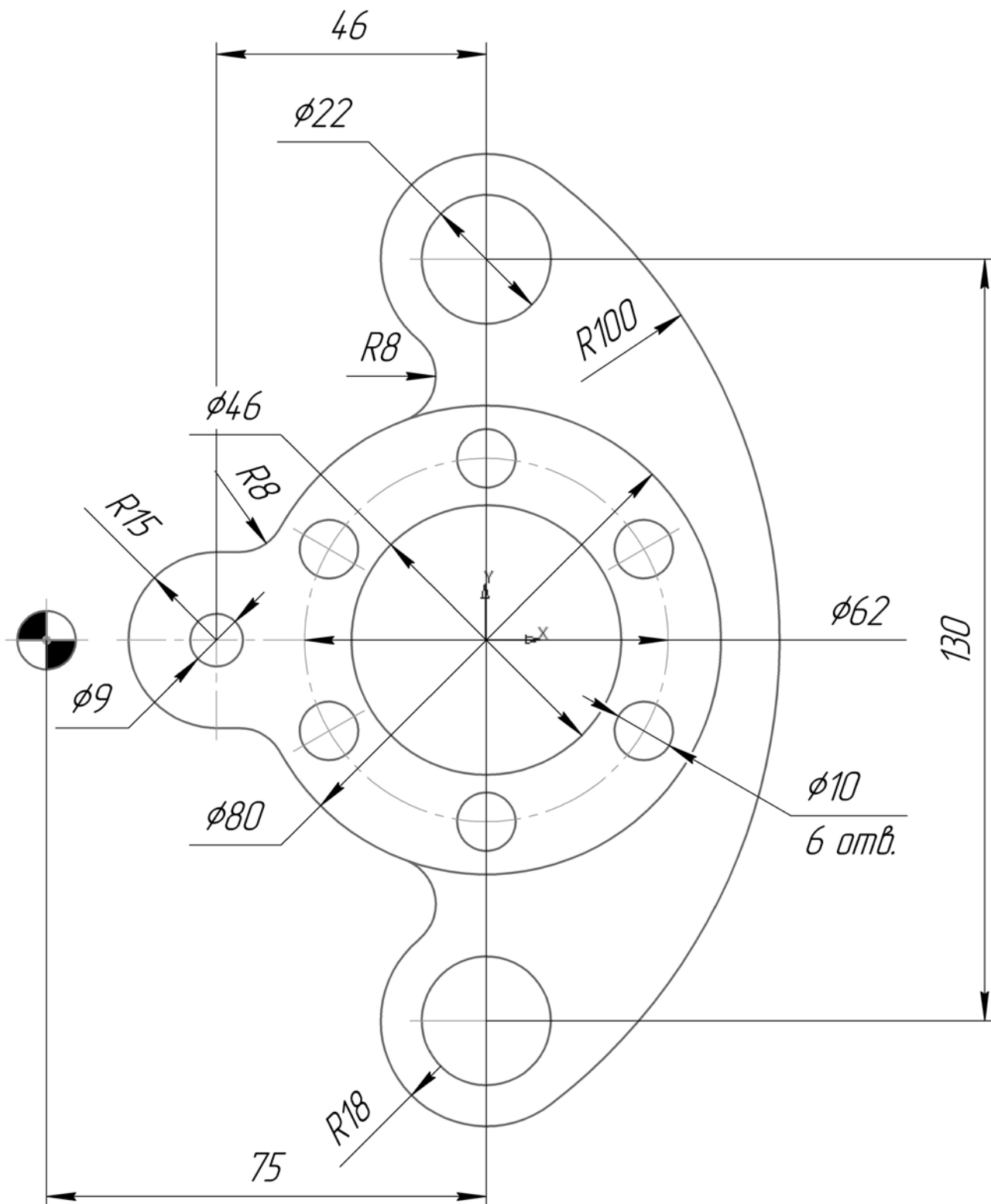


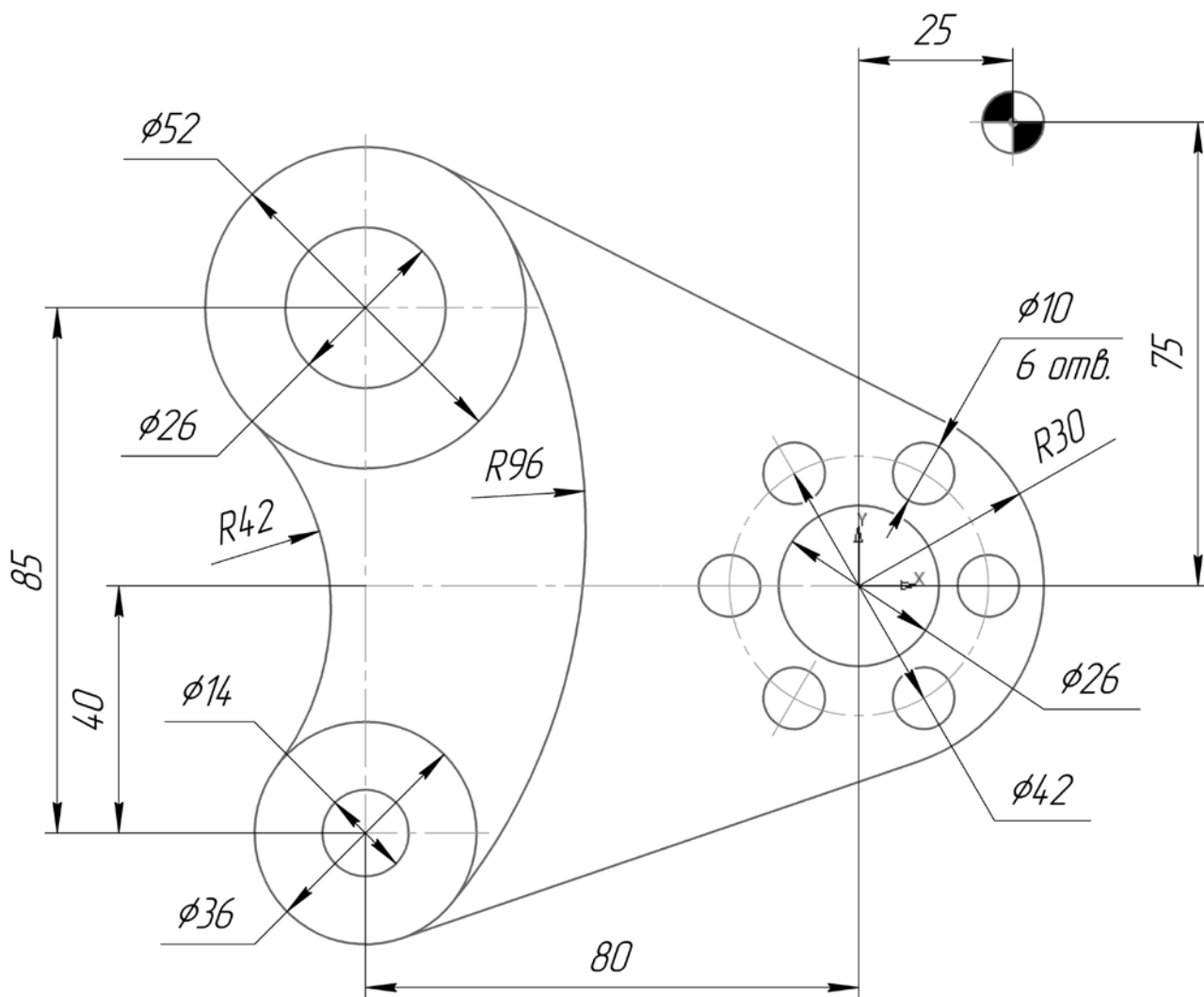


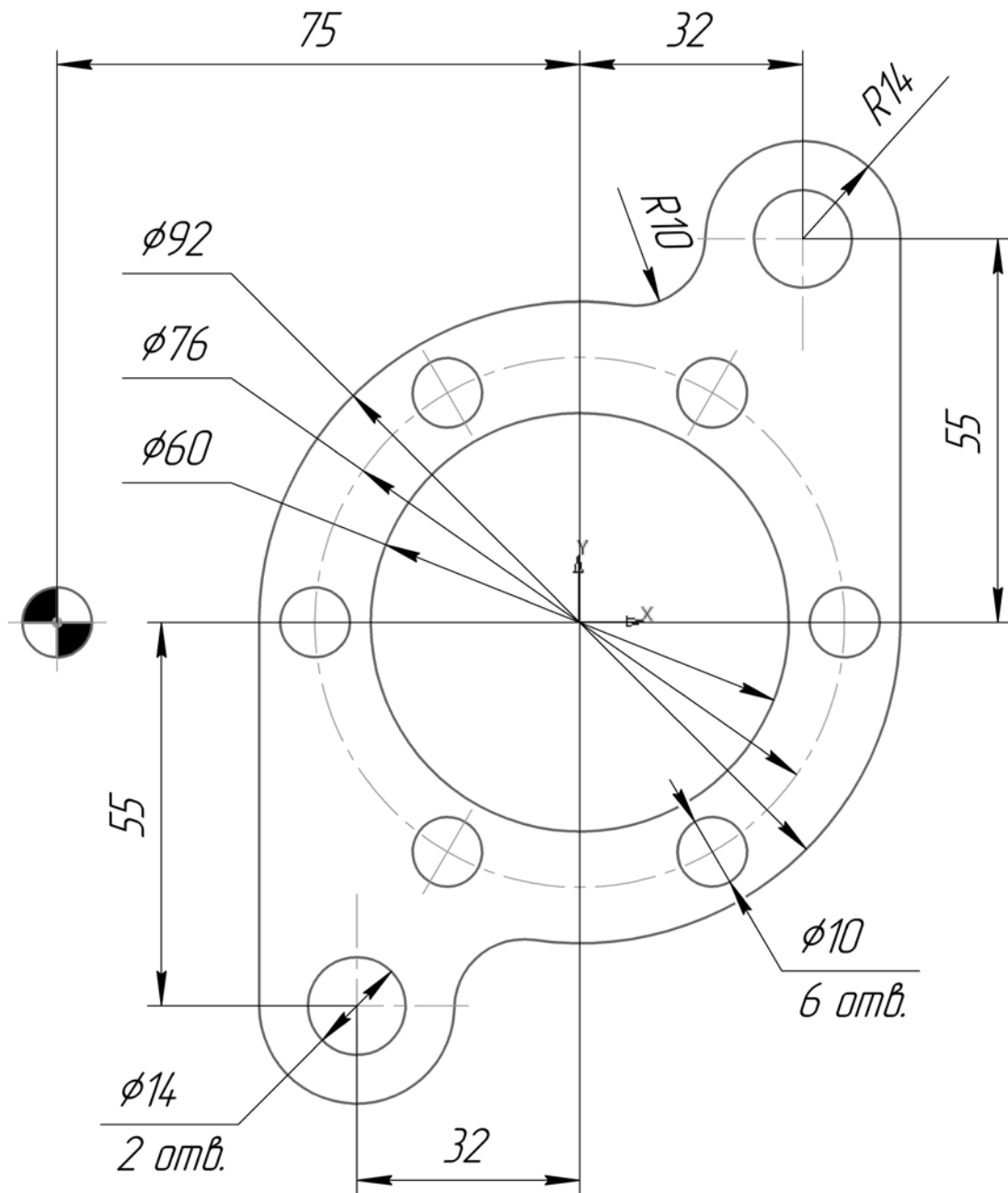


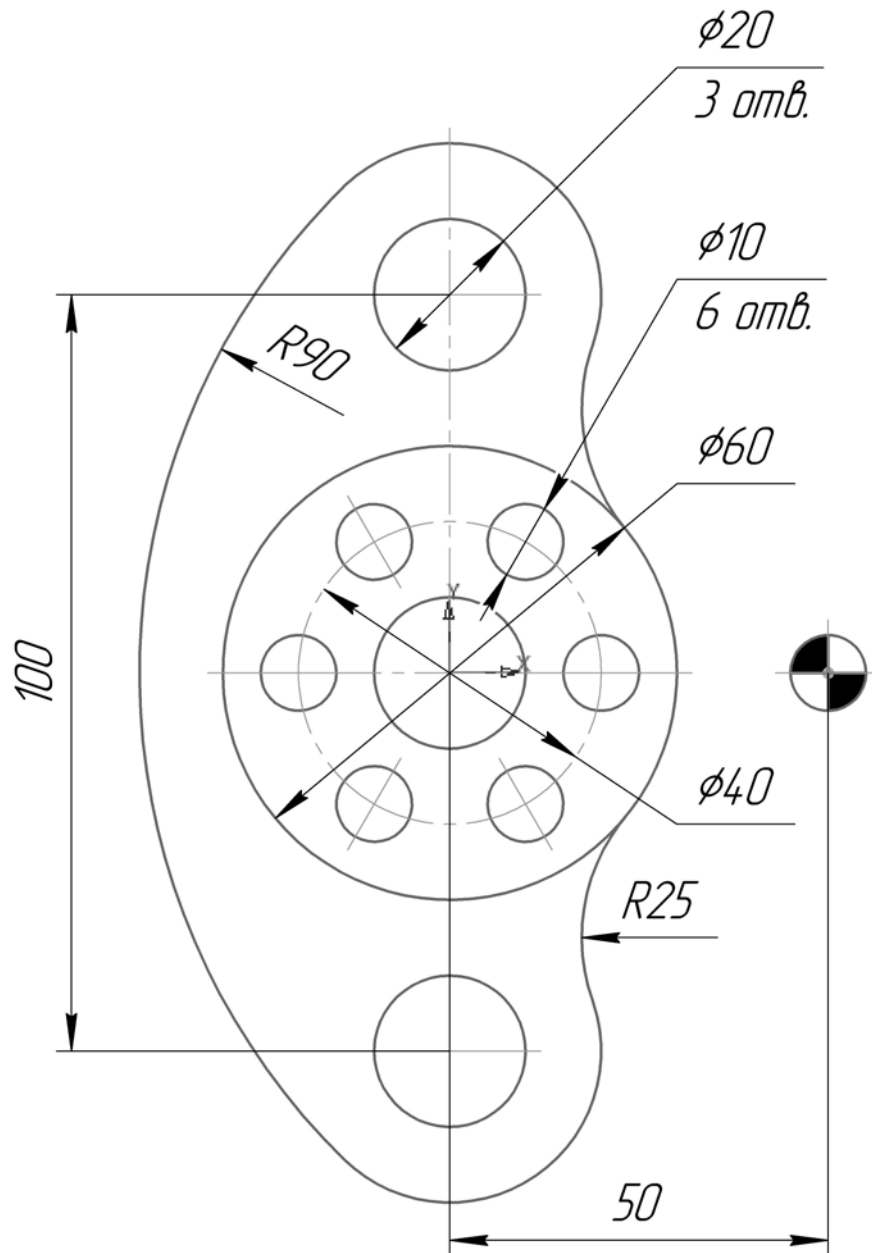


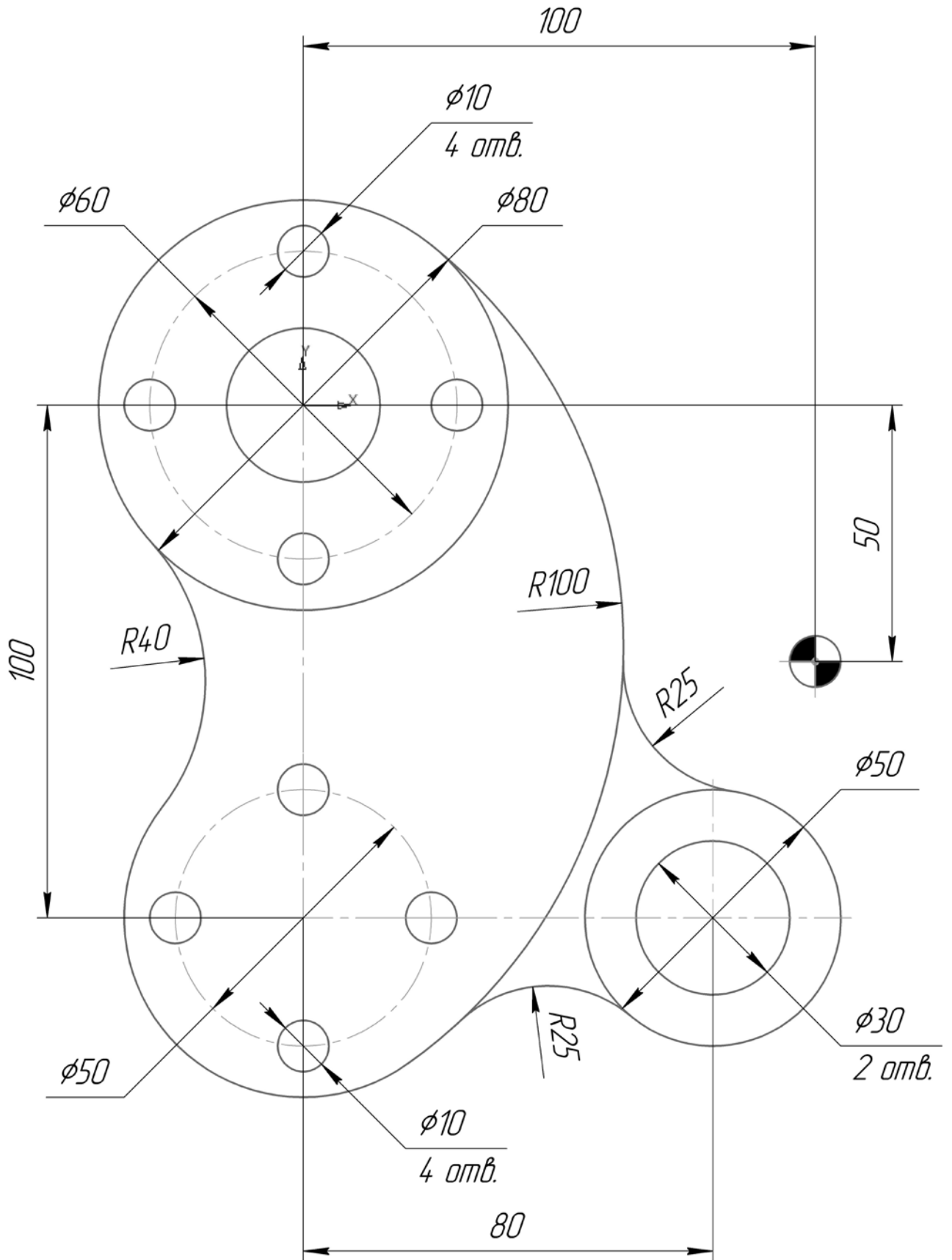














**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Леженинова  
« 14 » \_\_\_\_\_ 20\_\_\_\_ г.



**РЕШЕНИЕ ЗАДАЧИ ПОСТРОЕНИЯ МАРШРУТА ПЕРЕДВИЖЕНИЯ  
С ИСПОЛЬЗОВАНИЕМ ТЕОРИИ ГРАФОВ В СИСТЕМЕ MAPLE**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

УДК 519.6

Составитель Куц В.В.

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Решение задачи построения маршрута передвижения с использованием теории графов в системе Maple:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 20 с. :– Библиогр.: с. 5.

Содержат сведения по вопросам решения задач построения маршрута передвижения с использованием теории графов в системе Maple. Указываются порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 1,16 . Уч.-изд. л. 1,05.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

## 1 Цель работы

Получить практические навыки построения графовых моделей и реализации на их основе процедуры построения маршрута передвижения в системе Maple.

## 2 Задание

2.1 Изучить основные функции работы с графами в системе Maple.

2.2 Создать и сохранить новый документ в системе Maple.

2.3 В соответствии с заданием построить графовую модель и на её основе сформировать в созданном документе Maple граф и задать команды поиска пути на графе.

2.4 Построить два маршрута между заданными преподавателем вершинами.

2.5 Оформить отчёт

## 3 Краткие теоретические положения

Некоторые функции работы с графами:

*Graph(V,E)* – создает граф, заданный списком вершин V и ребер E;

*AddEdge(G, E)* - добавить в граф G ребра или дуги;

*AddVertex(seq(i,i=1..n),G)* - добавить к графу G вершины 1,2  
...n

*AdjacencyMatrix(G)* - матрица смежности графа G

*Arrivals(v,G)* - множество ребер, входящих в вершину v орг-  
рафа G

*ConnectedComponents(G)* - список вершин в связных компонентах графа

*ConnectedComponents (u,v,'weights'=L1,'names'=L2,'directed', G)* - соединить две вершины u, v. Можно указать вес, имя и выбрать оргграф (directed) или неогграф

*DeleteVertex(z,G)* - удаление вершины z из графа G;  
*DeleteEdge(z,G)* - удаление ребер z из графа G;

***Departures(v,G)*** - множество ребер, выходящих из вершины  $v$  орграфа  $G$

***DrawGraph(G)*** - нарисовать граф  $G$

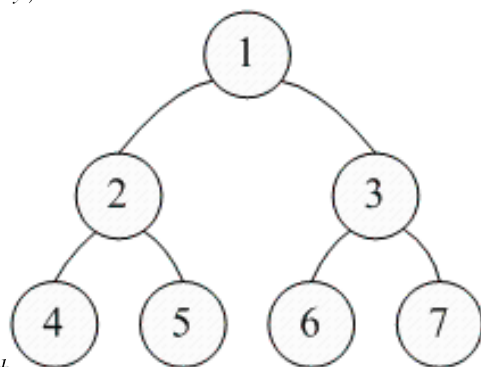
***Edges(G)*** - множество ребер графа  $G$

***IncidenceMatrix(G)*** - матрица инцидентности графа  $G$  (строки вершины, столбцы - ребра)

#### 4 Пример выполнения

> restart :

> with(GraphTheory) :



> #Создаём граф

>  $G := \text{Graph}(\{\{1, 2\}, \{1, 3\}, \{2, 4\}, \{2, 5\}, \{3, 6\}, \{3, 7\}\});$

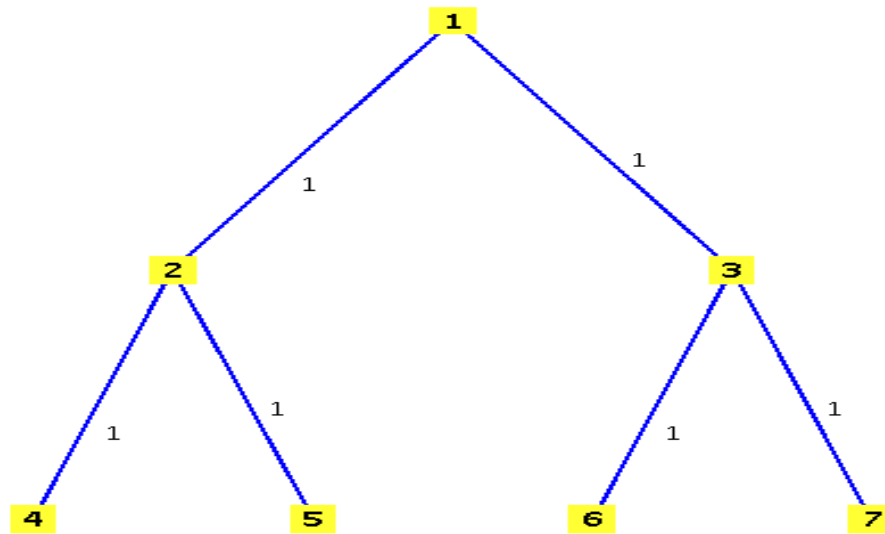
>  $G := \text{Graph } 5: \text{ an undirected unweighted graph with 7 vertices and 6 edge(s)}$

> #Посматриваем матрицу смежности

> AdjacencyMatrix(G);

```
[ 0 1 1 0 0 0 0
  1 0 0 1 1 0 0
  1 0 0 0 0 1 1
  0 1 0 0 0 0 0
  0 1 0 0 0 0 0
  0 0 1 0 0 0 0
  0 0 1 0 0 0 0]
```

- > #Отрисовываем граф
- > DrawGraph(G);



- > #Выполняем поиск пути
- > ShortestPath(G, 5, 7);

[5, 2, 1, 3, 7]

## 5 Требования к отчету

Отчет по лабораторной работе должен включать в себя:

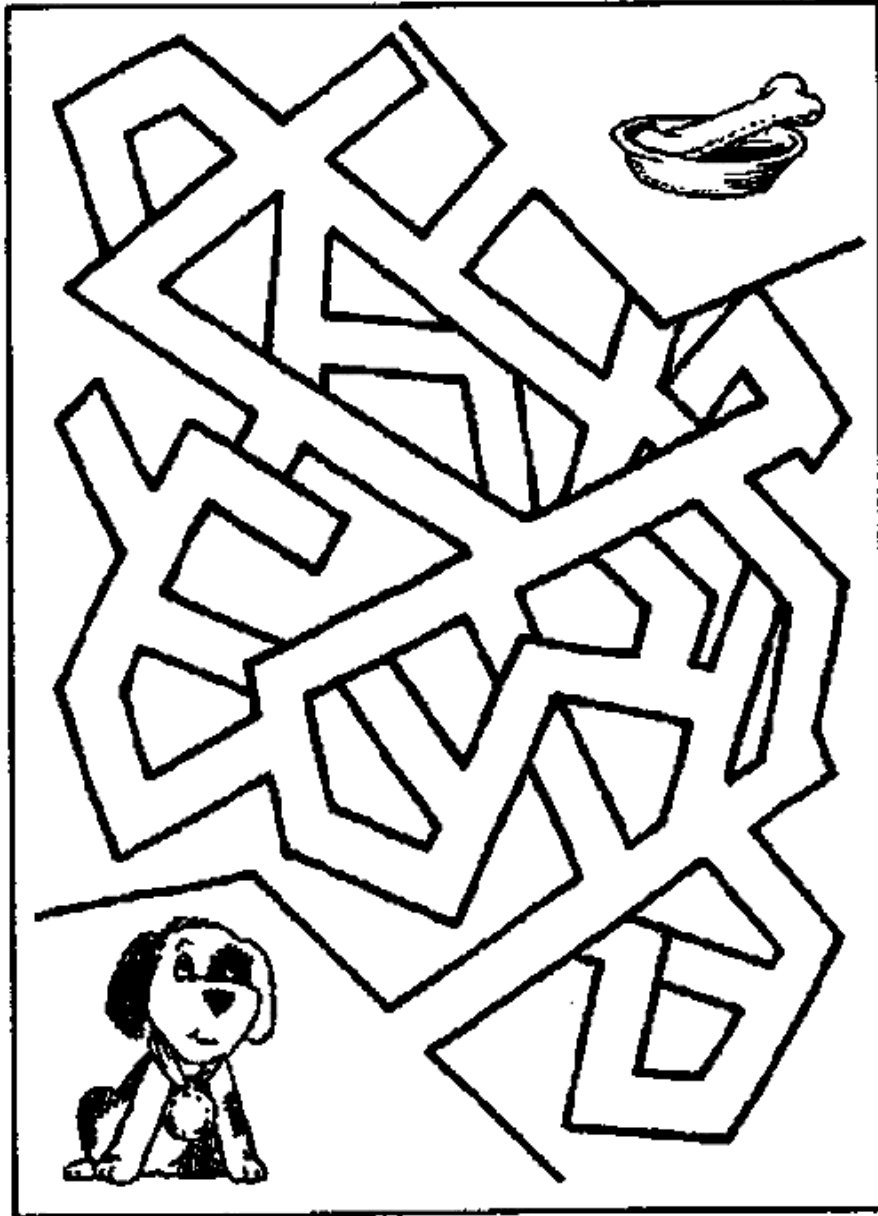
- тему лабораторной работы;
- цель лабораторной работы;
- задание к лабораторной работе;
- граф и матрицу смежности для вашего задания;
- написанный листинг программы в среде Maple

## 6 Библиографический список

- 1 <http://algotlib.narod.ru/Graph/Obhod.html>
- 2 <https://ru.wikipedia.org/wiki/%CF%E8%E1%E8%ED%E3%EB%E1%E8%ED%E3>
- 3 <http://hci.fenster.name/304y/practice/lab6/>

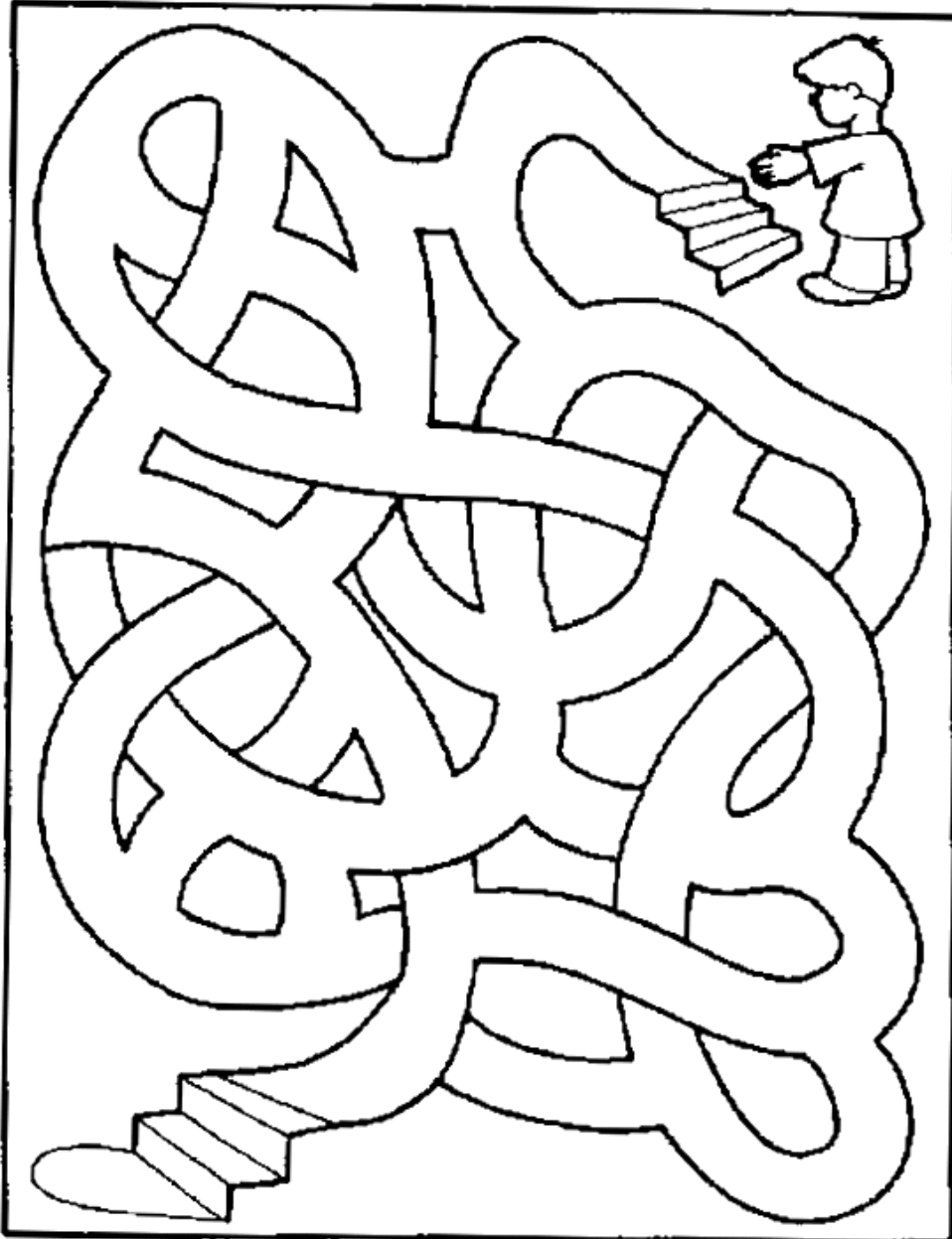
## 7. Задания к выполнению

Вариант 1



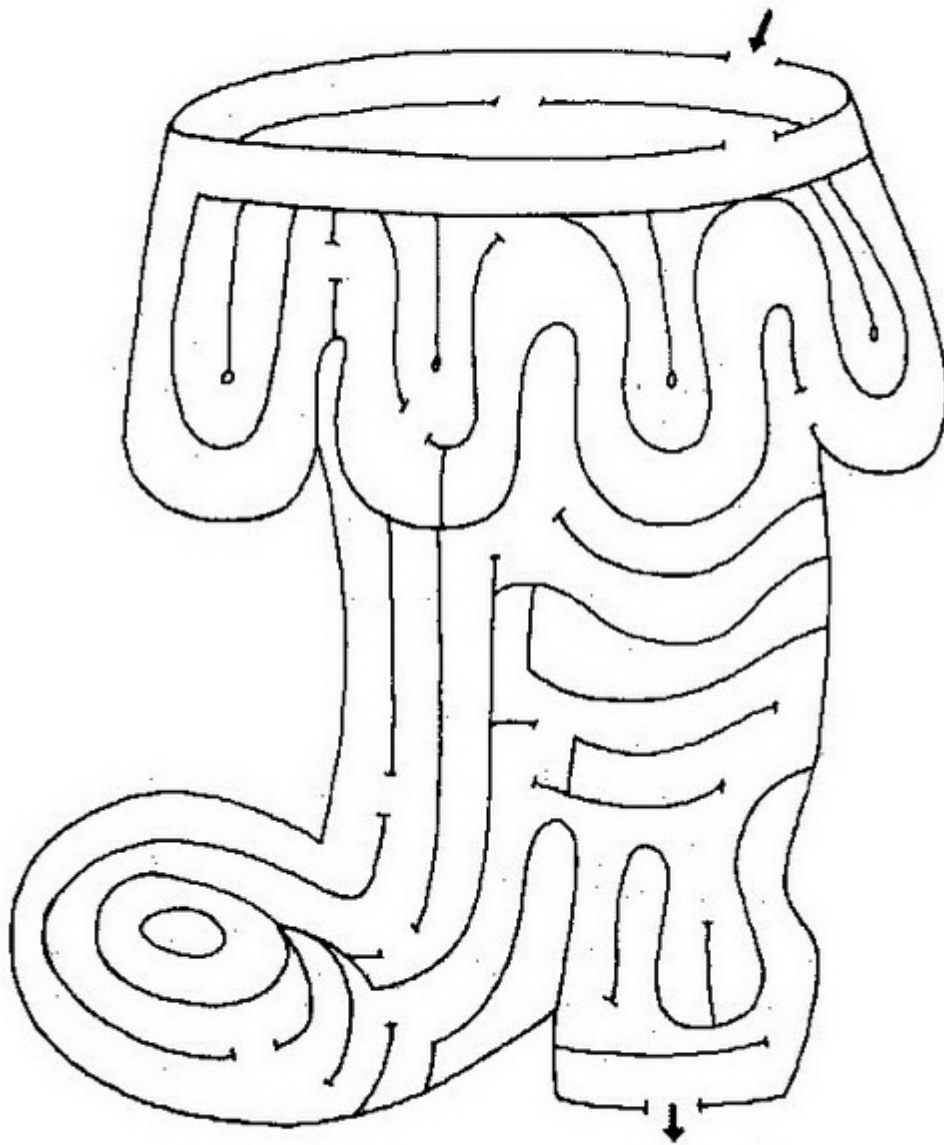
<http://www.vrac-coloriages.net>

Вариант 2



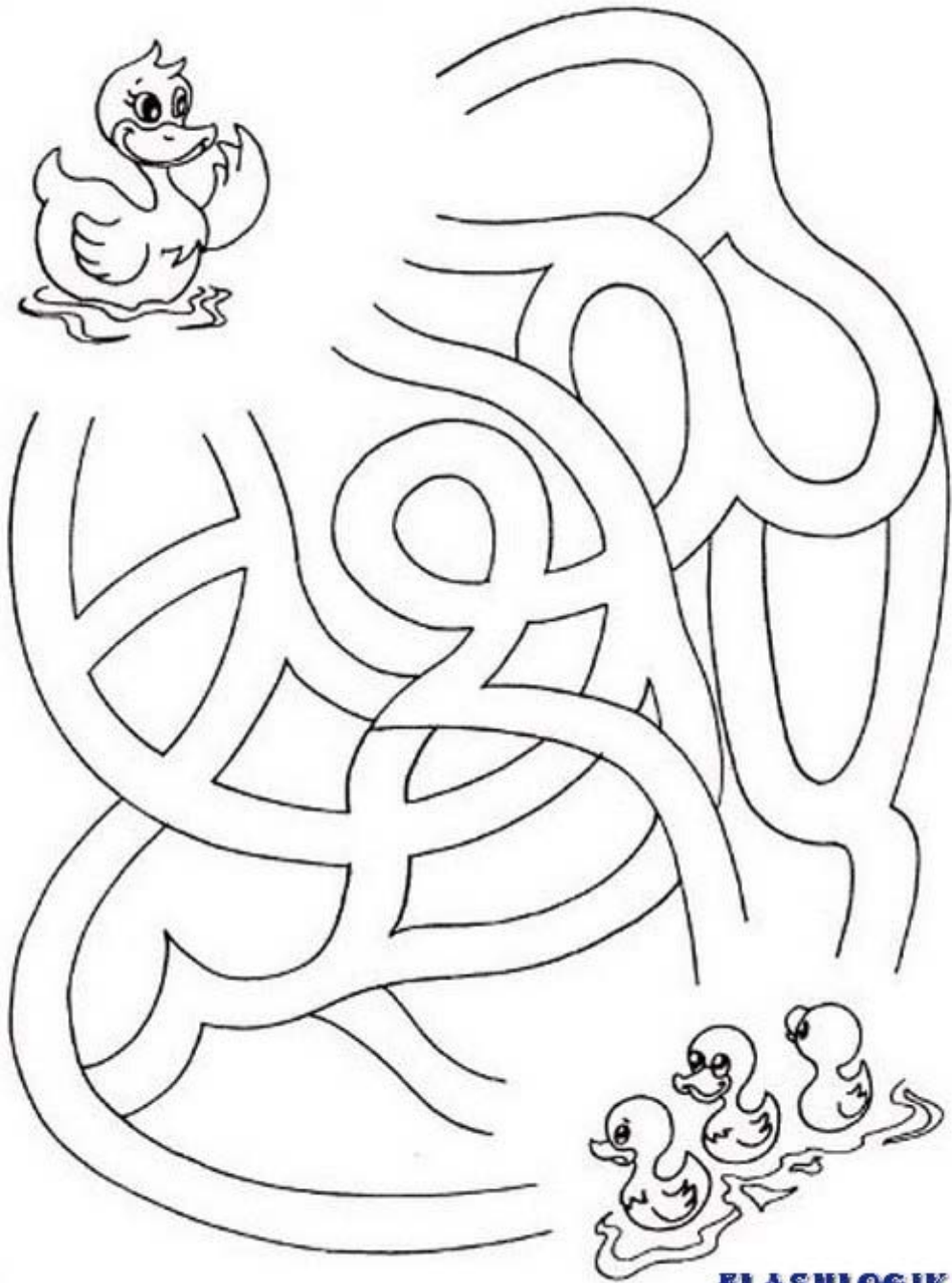
<http://www.vrac-coloriages.net>

Вариант 3

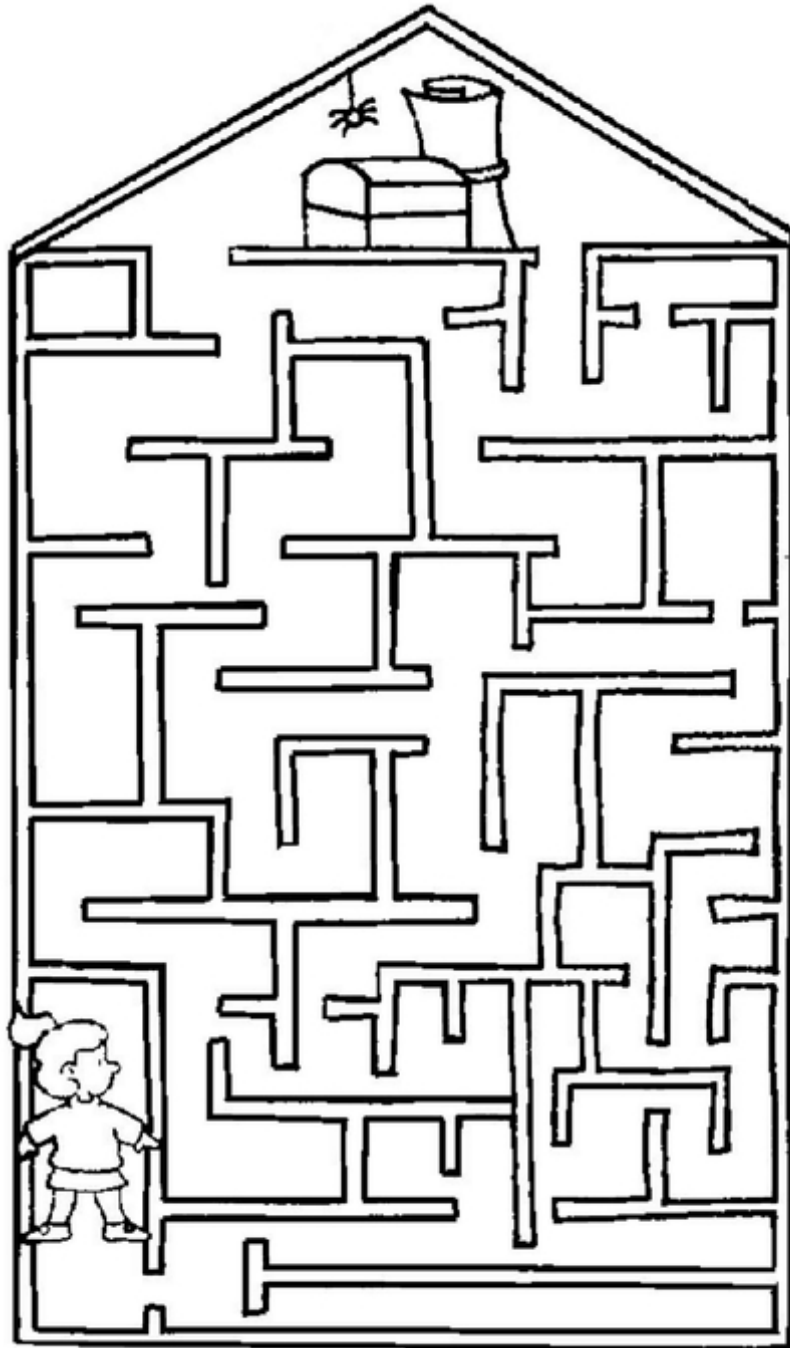




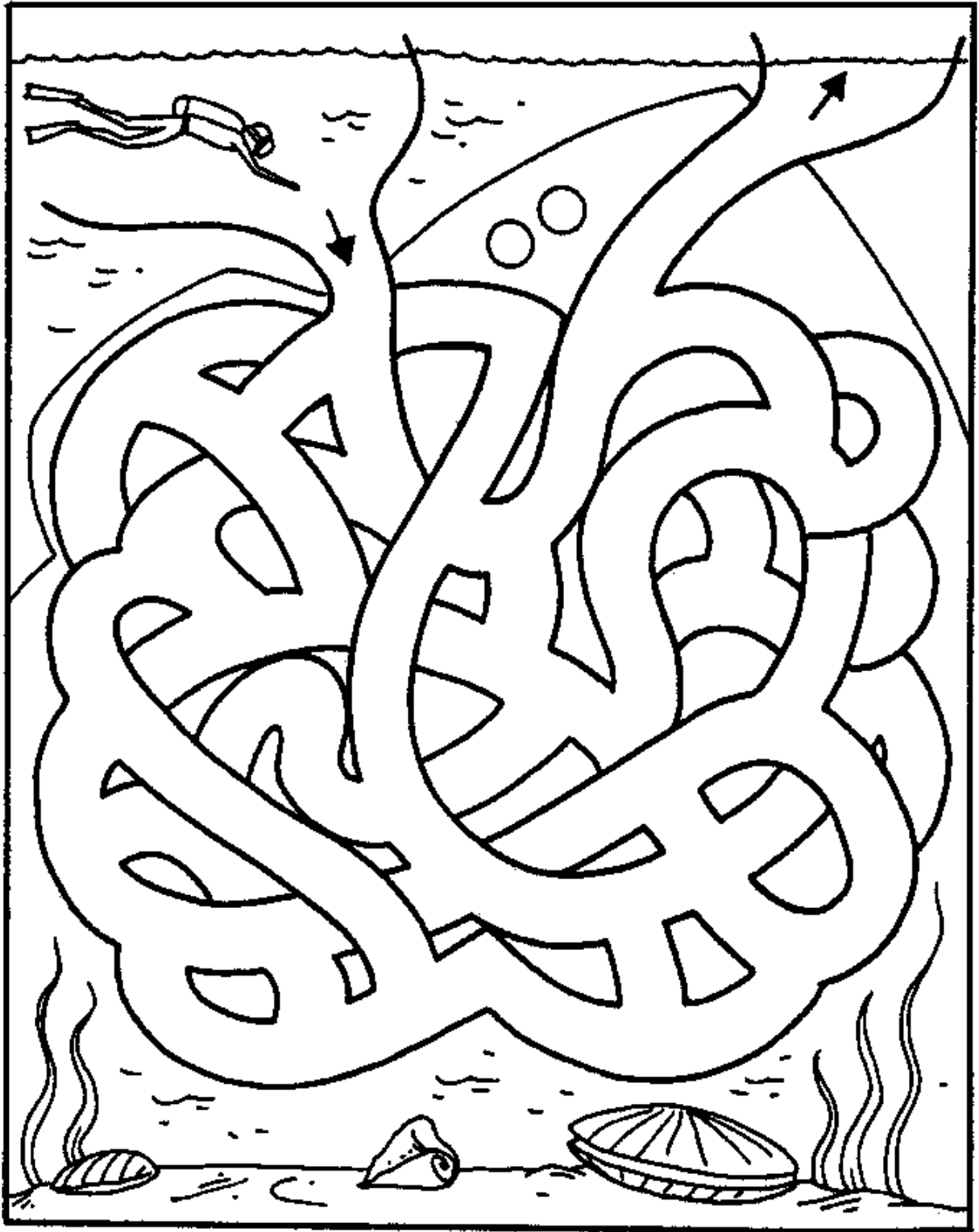
Вариант 4



Вариант 5



Вариант 6

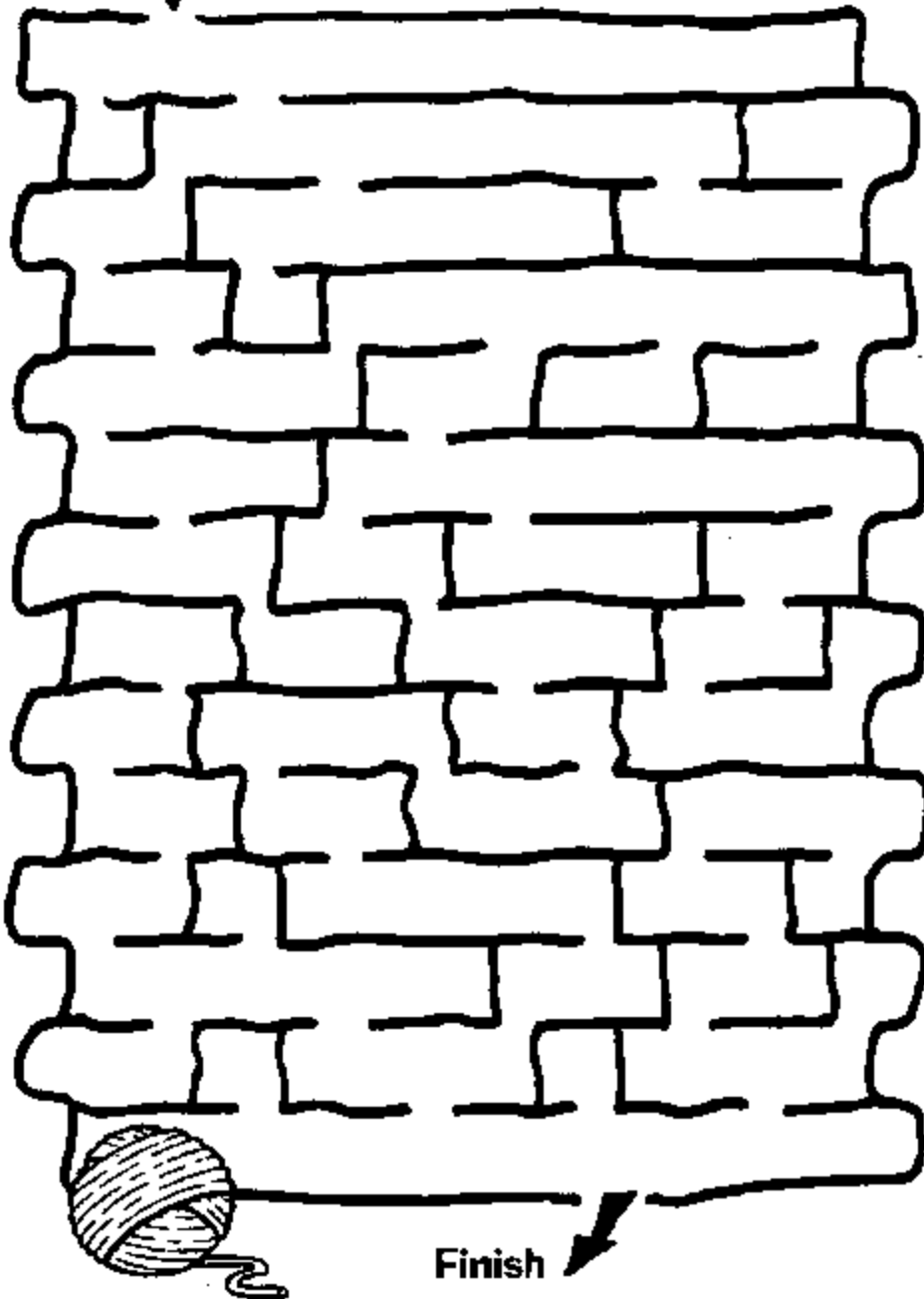


Вариант 7

**Help Kitty find her ball of yarn.**

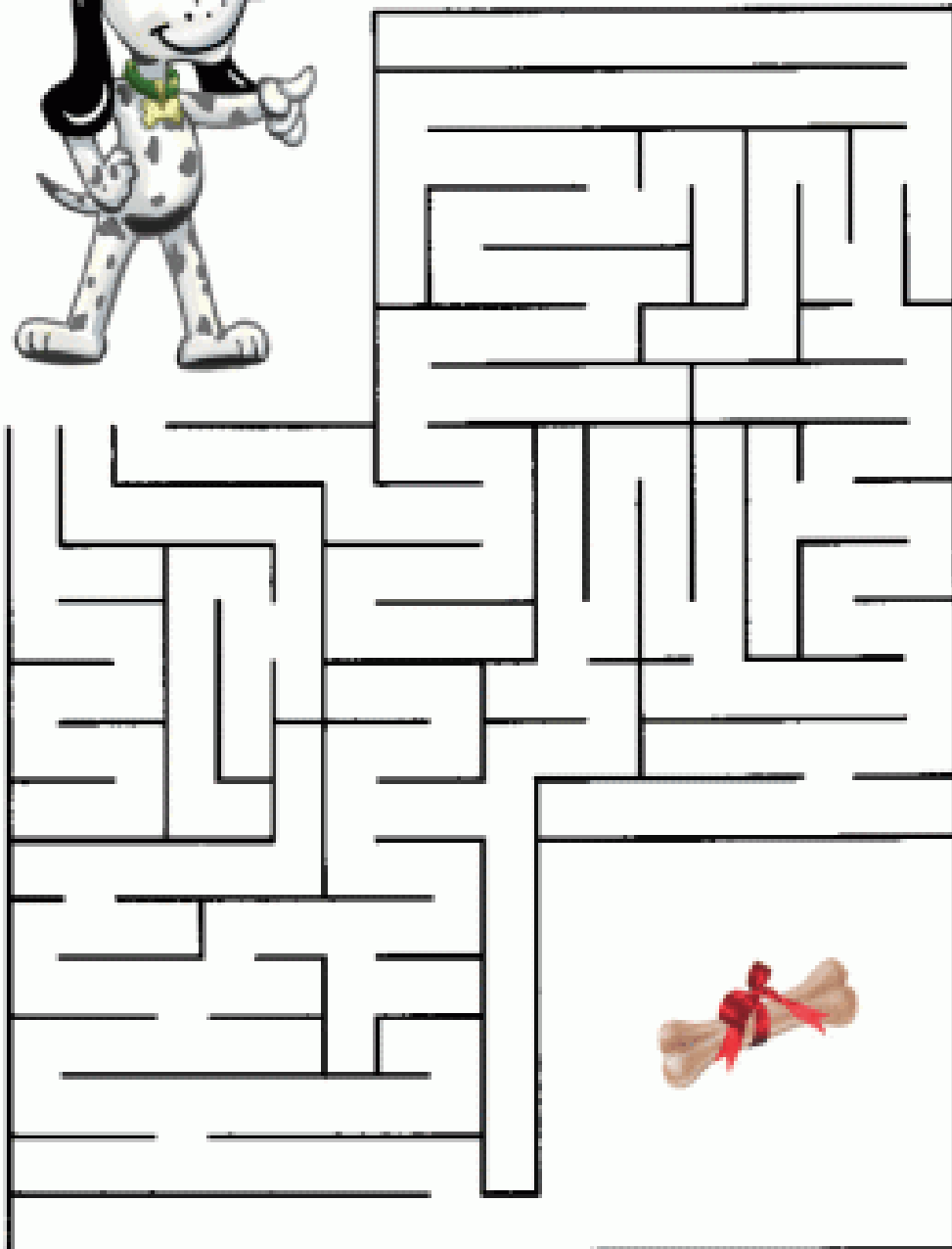


**Start**

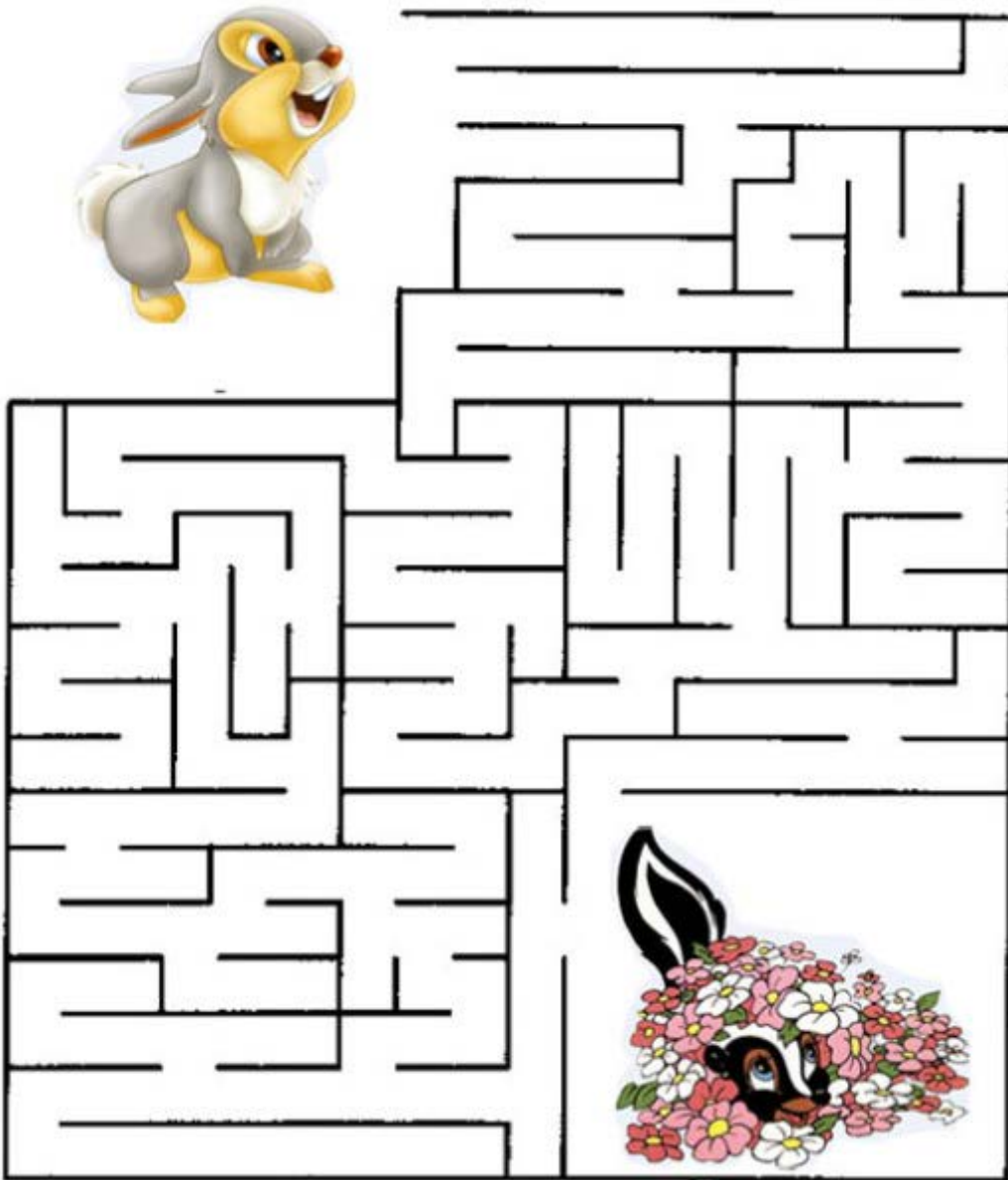


**Finish**

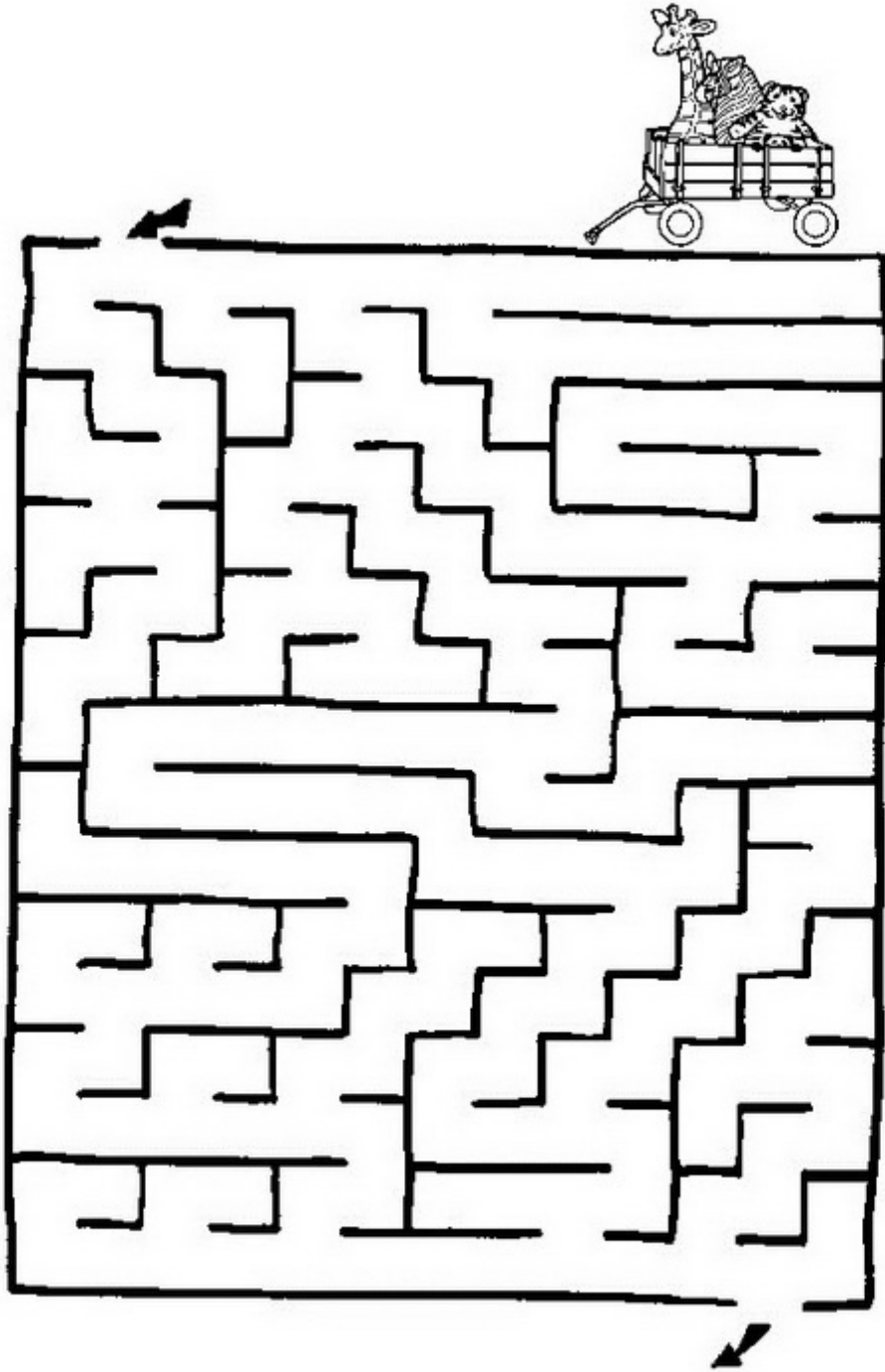
Вариант 8



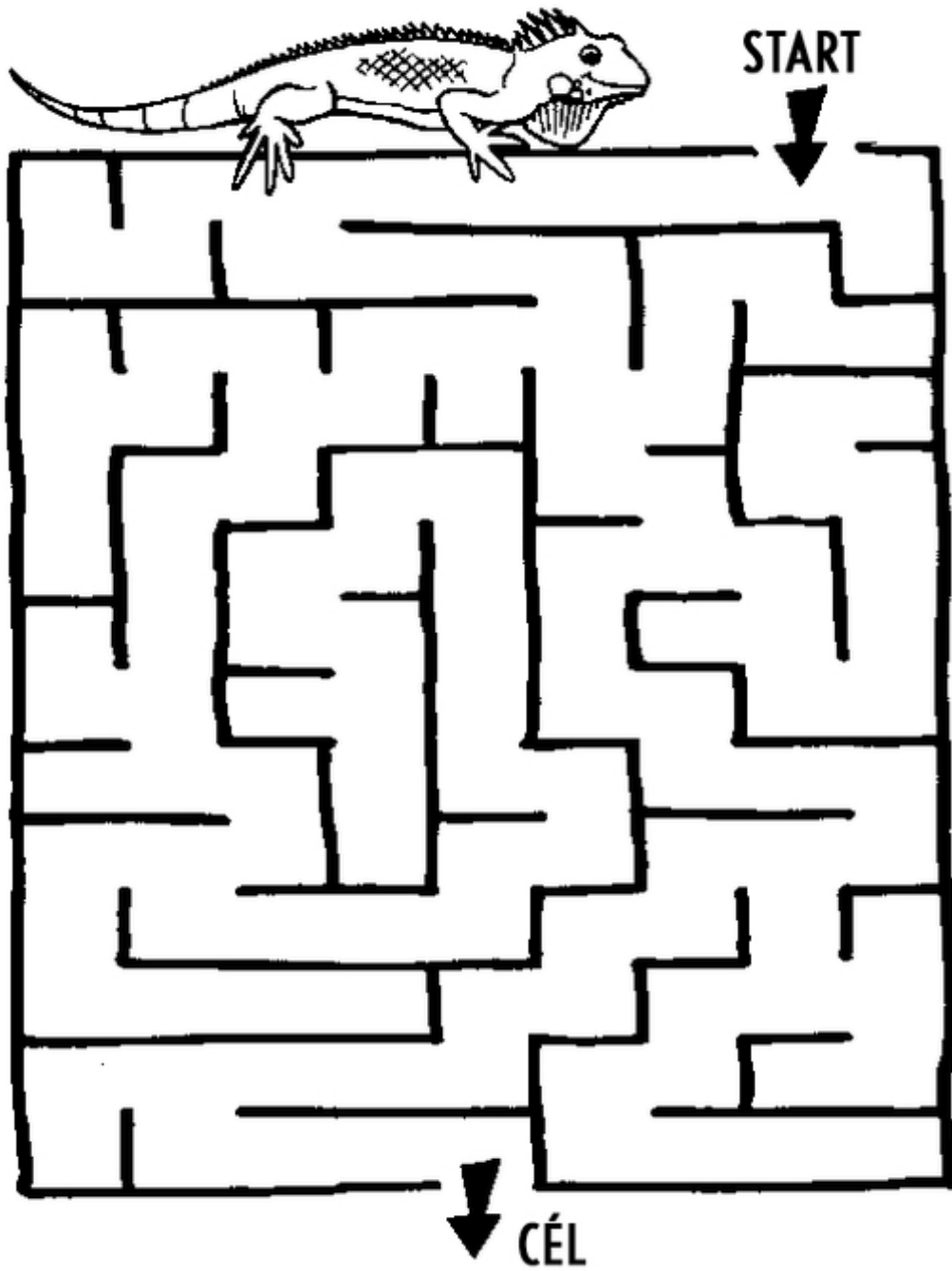
Вариант 9



Вариант 10

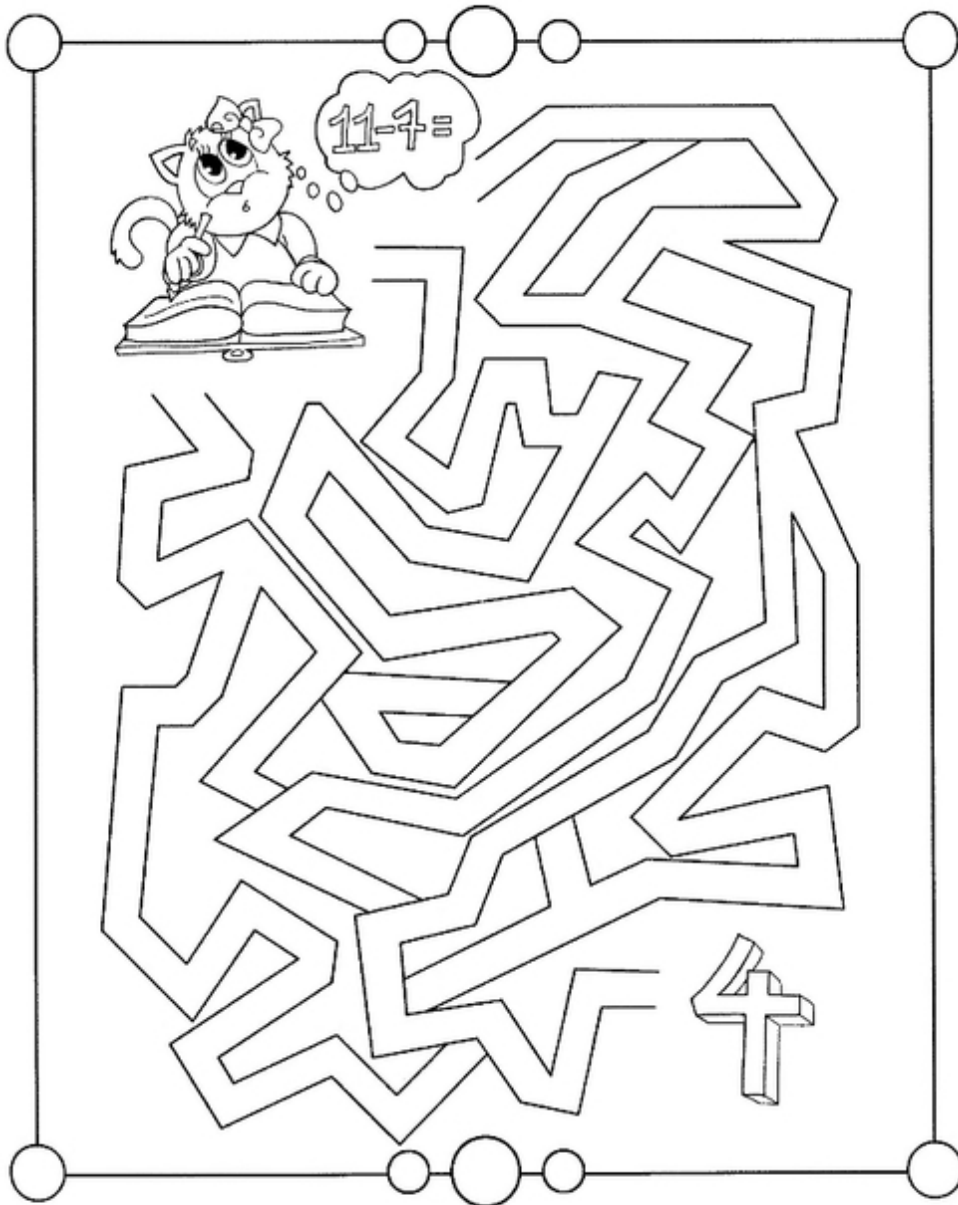


Вариант 11





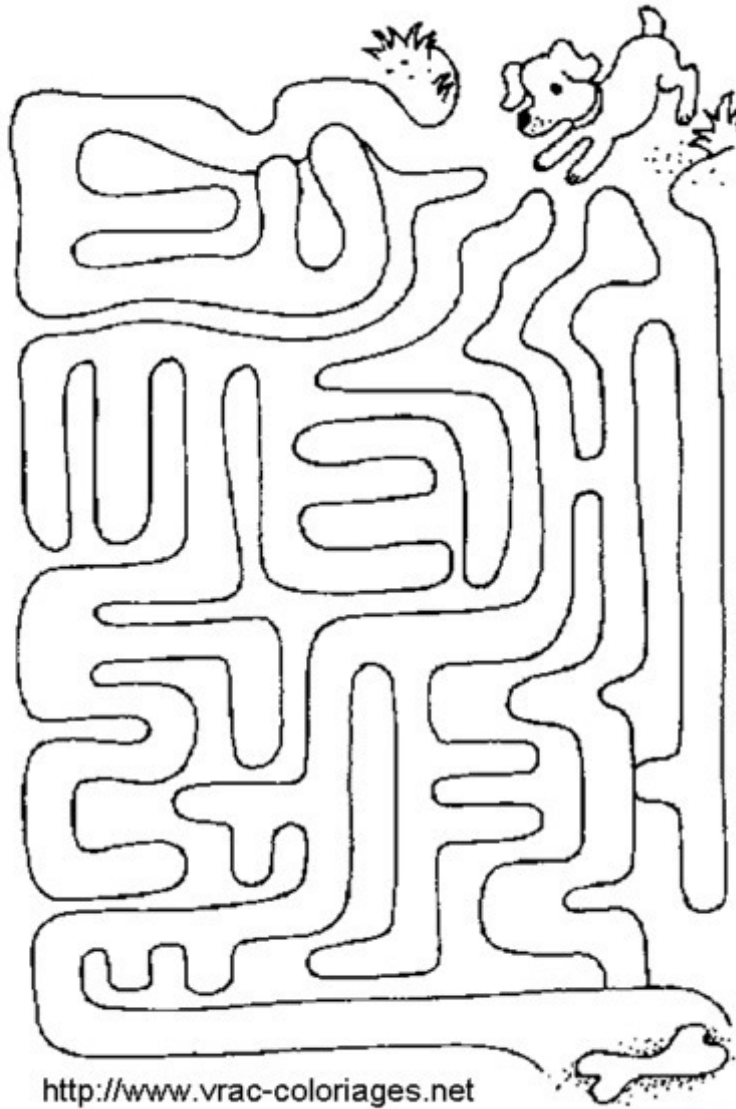
Вариант 12



Помоги котёнку найти ответ.  
Раскрась правильную дорожку.

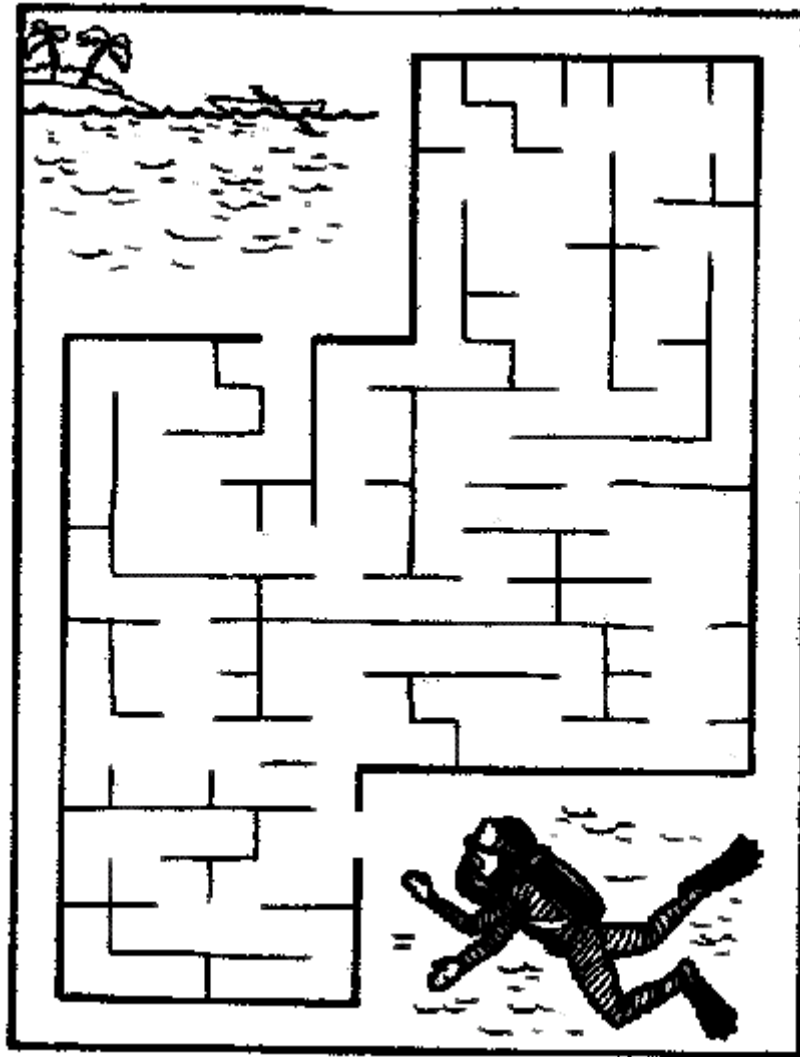
<http://www.liveinternet.ru/users/maknika/>

Вариант 13

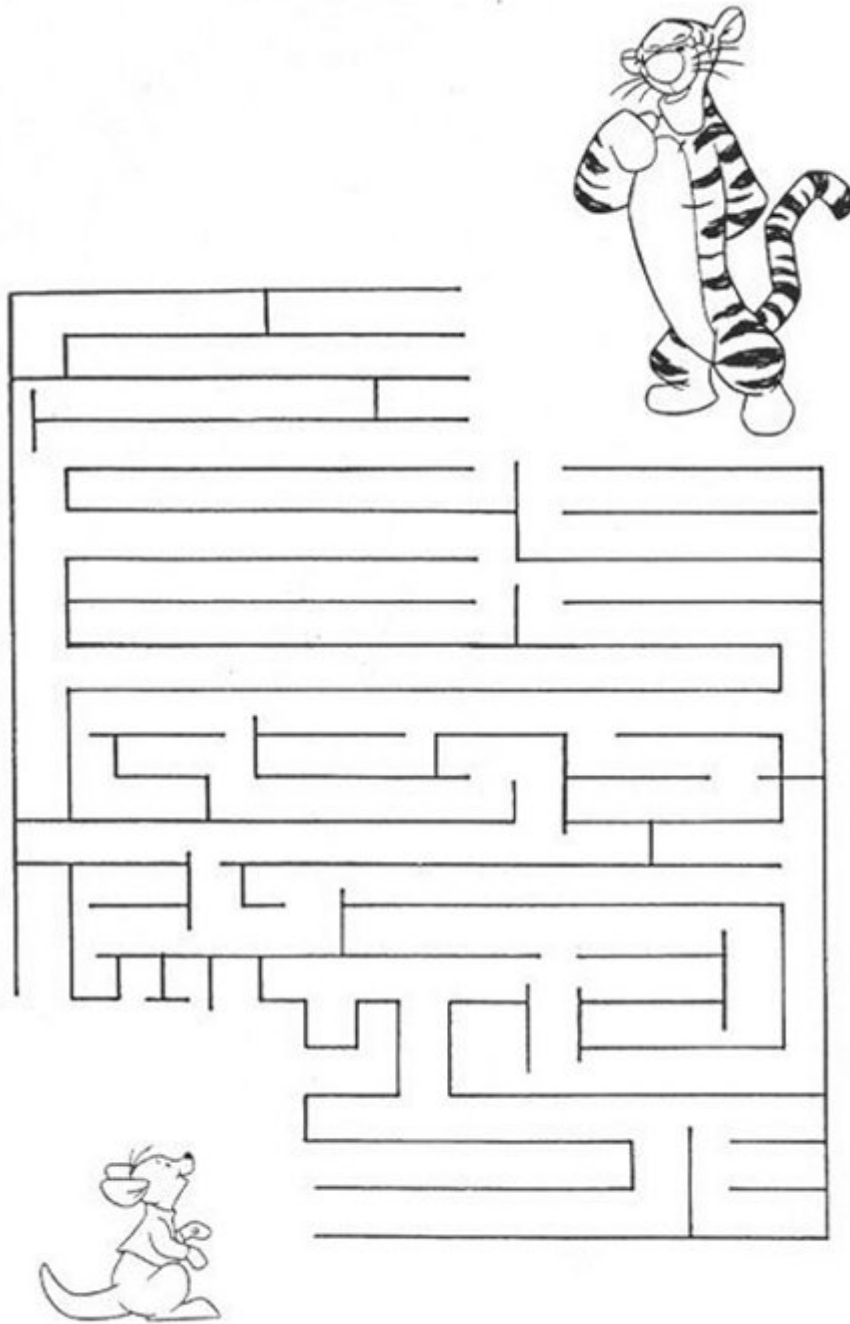


*Страна Мам*

Вариант 14



Вариант 15



**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Ложицкая  
« 14 » \_\_\_\_\_ г.



**РЕШЕНИЕ УРАВНЕНИЙ, НЕРАВЕНСТВ  
И ИХ СИСТЕМ В MAPLE**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023

УДК 519.6

Составители: В.В. Куц

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Решение уравнений, неравенств и их систем в Maple:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 19 с.:– Библиогр.: с. 19.

Содержат сведения о решении уравнений, неравенств и их систем в системе Maple. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 1,10 . Уч.-изд. л. 1,00.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

**1. ЦЕЛЬ РАБОТЫ:** приобрести практические навыки графического и численного решения алгебраических и тригонометрических уравнений, неравенств и их систем в СКМ MAPLE.

## 2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

### 2.1. Общие сведения

MAPLE – система компьютерной математики (СКМ), позволяющая решать сложные математические задачи без дополнительного программирования. Подробнее об этом см. в [4].

Работа Maple организована в диалоговом режиме: вопрос – ответ в отдельном блоке. Блок выделяется слева квадратной скобкой, длина которой зависит от размеров и количества исходных выражений (вопросов) и результатов вычислений (ответов). Строка ввода математических выражений (командная строка) имеет отличительный символ [ $\>$ ] (рис. 1).

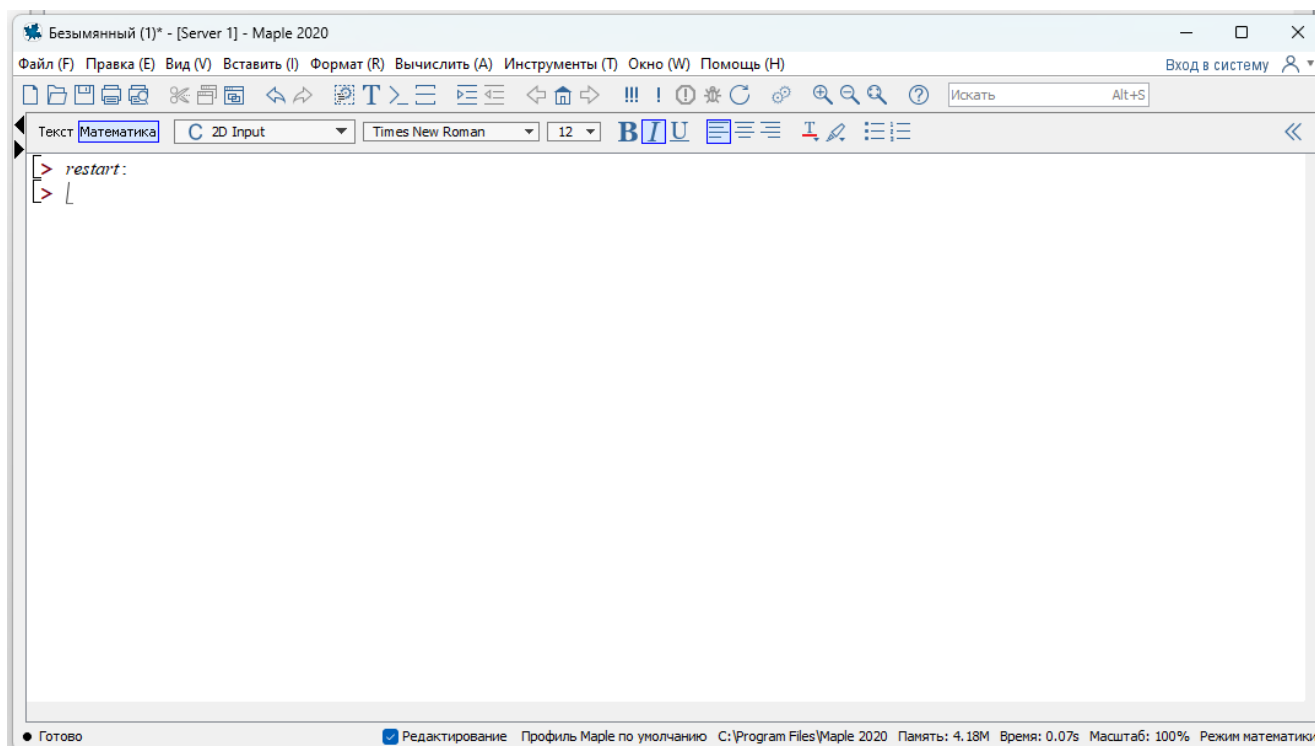


Рис. 1– Окно СКМ Maple

В командной строке записываются выражения, которые фор-

мируются из операторов и операндов. Результат вычислений (по умолчанию) возвращается в символьном виде, то есть в виде математических формул. Ввод выражения завершается символом фиксации конца выражения – *точкой с запятой*, если ответ выводится в строку вывода, или *двоеточием*, если ответ не выводится.

Выражение можно задавать, используя встроенные функции или создавая новые. Функция в выражениях может вводиться несколькими способами:

- с использованием оператора присваивания

```
[> fun1(x,y):=x^2+y^2;
```

$$fun2 := (x, y) \rightarrow y^2 + x^2$$

- с использованием функционального оператора  $\rightarrow$

```
[> fun2:=(x,y)->x^2+y^2;
```

$$fun2 := (x, y) \rightarrow y^2 + x^2$$

```
[> fun2(2,5);# вызов функции с параметрами 2, 5
```

29

- с использованием оператора «unapply

```
[>f:=unapply(x^2+y^2,x,y);
```

$$f:=unapply(x^2+y^2,x,y);$$

```
[>f(-7,5);
```

74

### 3.2. Типовые и расширенные средства графики

В само ядро Maple встроено ограниченное число функций графики. Для построения графиков более сложных типов командой *with* необходимо подключать пакеты расширений Maple. Подробнее об этом см. в [4].

Для построения **двумерных графиков** используется команда *plot*.

*Формат*

```
[> plot(fun, variable_x {,variable_y}{,option});
```

где *fun* – функция, график которой строится;

*variable\_x* – переменная, указывающая область изменения по горизонтали;

*variable\_y* – переменная, указывающая область изменения по вертикали;

*option* – набор параметров, задающих стиль построения графика



функции (см. таблицу 1).

Если в одних координатах нужно построить графики нескольких функций, эти функции берутся в квадратные скобки.

При построении графиков функцию можно определять через переменную.

*Формат*

`[> plot([fun1,... funN], variable_x {,variable_y}{,option});`

Таблица 1 - Параметры, задающие стиль построения графика функции в разделе *{,option}*

Параметр	Назначение параметра
<i>numpoints</i>	изменение количества точек графика (по умолчанию=49)
<i>color</i>	задание цвета кривой графика
<i>title</i>	добавление заголовка графика (например, <code>title="string"</code> )
<i>coords</i>	выбор системы координат, этот параметр задает 15 типов координатных систем. По умолчанию задана прямоугольная система координат
<i>axes</i>	задание типа осей координат ( <i>frame</i> - рамка, <i>boxed</i> - прямоугольник, <i>normal</i> - ортогональные, <i>none</i> – без осей)
<i>thickness</i>	толщина линии графика
<i>xtickmarks,</i> <i>ytickmarks</i>	управление числом меток на оси, т.е. задает минимальное число отметок по оси <i>x</i> и соответственно
<i>style</i>	стиль построения графика ( <i>line</i> – выводится интерполяционная кривая, <i>point</i> – выводятся точки)
<i>scalling</i>	масштаб графика ( <i>constrained</i> – сжатый, <i>unconstrained</i> - несжатый)
<i>size</i>	размер шрифта в пунктах
<i>symbol</i>	тип точки графика в виде символа ( <i>box</i> - прямоугольник, <i>cross</i> - крест, <i>circle</i> - окружность, <i>point</i> – точка, <i>diamond</i> - ромб)
<i>titlefont</i>	шрифт для заголовка
<i>labelfont</i>	шрифт для меток (labels) на осях координат

$view=[A,B]$	определение максимальной и минимальной координат, в пределах которых график будет отображаться на экране, где $A=[xmin..xmax]$ , $B=[ymin..ymax]$ .
--------------	---

**Трехмерными** называют графики, отображающие функции двух переменных  $z(x,y)$ . На деле трехмерные графики представляют собой объемные проекции в аксонометрии.

Для построения таких графиков Maple имеет встроенную в ядро функцию *plot3d*. Она может использоваться в следующих *формах*:

```
plot3d(expr1, x = a..b, y = c..d, p),
plot3d(f, a..b, c..d, p),
plot3d([exprf, exprg, exprh], s = a..b, t = c..d, p),
plot3d([f, g, h], a..b, c..d, p).
```

Здесь

$a,b,c,d$  – пределы изменения соответствующих переменных;  
 $p$  – параметры, с помощью которых можно в широких пределах управлять видом трехмерных графиков.

Существенно расширяет возможности графики системы Maple пакет *plots*, который содержит почти полсотни функций. Назначение всех функций можно посмотреть в справочной системе Maple.

Подключается этот пакет командой *with (plots)*;

### 2.3. Решение уравнений и неравенств

Для решения уравнений, неравенств и их систем в СКМ Maple используется функция *solve*, которая возвращает последовательность решений.

*Формат*

```
solve(eqn, var);
```

где *eqn* – уравнение, неравенство или процедура;

*var* – имя переменной, относительно которой решается уравнение.

Команда *solve*, примененная для решения тригонометрического уравнения, выдает только главные решения, то есть решения в интервале от 0 до  $2\pi$ .

*Например:*

```
[>solve(sin(x)=cos(x),x);
```

Для того, чтобы получить все решения, следует предварительно ввести дополнительную команду `_EnvAllSolutions:=true`.

*Например:*

```
>_EnvAllSolutions:=true;
      _EnvAllSolutions:=true;
>solve(sin(x)=cos(x),x);
       $\frac{1}{4}\pi + \pi\_Z1\sim$ 
```

В Maple символ `_Z1~` обозначает константу целого типа, поэтому решение данного уравнения в математической форме имеет вид

$$x := \frac{1}{4}\pi + \pi n,$$

где  $n$  – целые числа.

Команда `solve` применяется также для решения **неравенств**. Решение неравенства выдается в виде интервала изменения искомой переменной. В том случае, если решение неравенства полуось, то в поле вывода появляется конструкция вида `RealRange(-∞, Open(a))`, которая означает, что  $x \in (-\infty, a)$ ,  $a$  – некоторое число. Слово `Open` означает, что интервал с открытой границей. Если этого слова нет, то соответствующая граница интервала включена во множество решений.

*Например:*

```
[>s:=solve(sqrt(x+3)<=sqrt(x-1)+sqrt(x-2),x);
       $s := \left[ \frac{2\sqrt{21}}{3}, \infty \right)$ 
```

Если необходимо получить решение неравенства не в виде интервального множества типа  $x \in (a, b)$ , а в виде ограничений для искомой переменной типа  $a < x$ ,  $x < b$ , то переменную, относительно которой следует разрешить неравенство, следует указывать в фигурных скобках.

*Например:*

```
[> solve(1-1/2*ln(x)>2,{x});
```

$$\left\{ 0 < x, x < \frac{1}{e^2} \right\}$$

## 2.4. Решение систем линейных алгебраических уравнений

Системы линейных алгебраических уравнений можно решать также, используя команду *solve*. Такое решение в силу простоты записи может быть предпочтительным. Для решения система уравнений и перечень неизвестных задаются в виде множеств, то есть с использованием фигурных скобок.

*solve({eq1,eq2,...},{x1,x2,...})*

Если для дальнейших вычислений необходимо использовать полученные решения уравнений, то команде *solve* следует присвоить какое-нибудь имя *name*. Затем выполняется присвоения команда *assign(name)*. После этого над решениями можно будет производить математические операции.

Решение систем из трех линейных уравнений имеет наглядную геометрическую интерпретацию – в виде точки, в которой пересекаются три плоскости, каждая из которых описывается функцией двух переменных. Это позволяет сделать функция имплицитивной графики *implicitplot3d*.

*Формат:*

*implicitplot3d(expr1,x=a..b,y=c..d,z=p..q,<options>);*

*implicitplot3d(f,a..b,c..d,p..q,<options>);*

где *f, expr1* – уравнение поверхности, которая должна быть построена;

*a,b,c,d,p,q* – пределы изменения соответствующих переменных;

*options* – параметры, с помощью которых можно в широких пределах управлять видом трехмерных графиков.

С помощью команды *solve* можно также решить систему неравенств.

*Например:*

*[>solve({x+y>=2,x-2\*y<=1,x-y>=0,x-2\*y>=1},{x,y});*

$$\left\{ x = 2y + 1, \frac{1}{3} \leq y \right\}$$

## 3. ПРАКТИЧЕСКАЯ ЧАСТЬ

### Задание 1

Построить графики трех функций  $\sin(x)$ ,  $\sin(x)/x$ ,  $\sin(x^3/100)$  линиями трех цветов и трех типов.

*[>plot([sin(x),sin(x)/x,sin(x^3/100)],x=10..10,color=[black,blue,red],style=[line,line,point]);*

Результат представлен на рисунке 2.

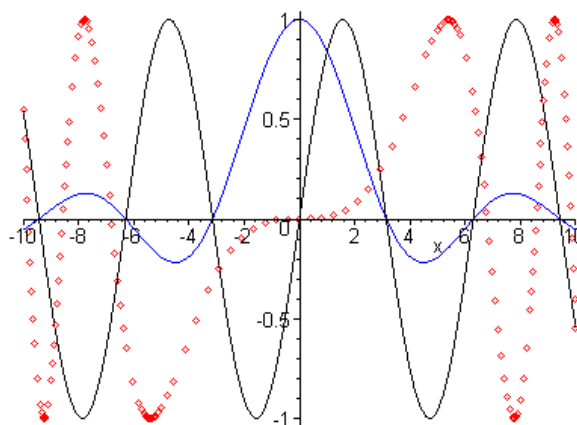


Рис.2 – Графики трех функций

### Задание 2

Построить поверхность  $h^2$  в цилиндрической системе координат.

`[>plot3d(h^2,a=-Pi..Pi,h=-5..5,coords=cylindrical,style=patch,color=sin(h));`

Результат представлен на рисунке 3

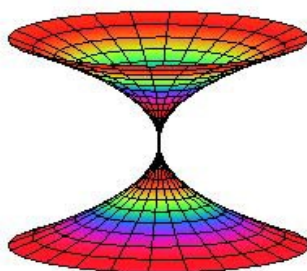


Рис.3 – Пример трехмерного графика

### Задание 3

Построить поверхности  $2*\sin(x*y)$ ,  $x^2+y^2-10$ ,  $-x^2-y^2+10$  в одной системе координат

`[>smartplot3d(2*sin(x*y),x^2+y^2-10,-x^2-y^2+10);`

Результат представлен на рисунке 4.

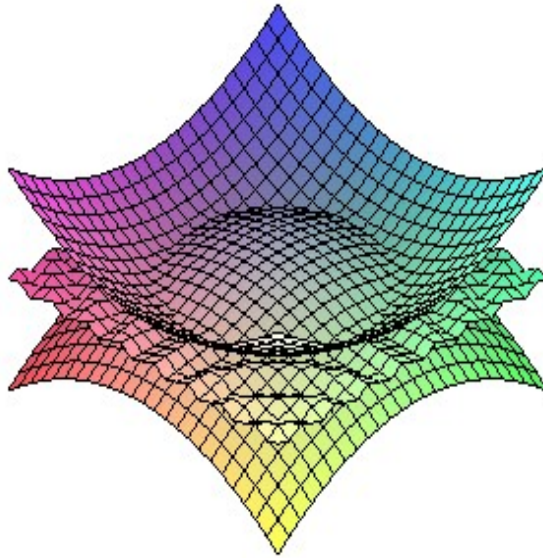


Рис.4 – Поверхности, построенные в одной системе координат

#### Задание 4

Решить нелинейное уравнение вида  $y=2x^2+1.25x-3.2$ . Выделить графически интервал изоляции корня уравнения и вычислить корень с точностью  $\varepsilon=10^{-5}$ .

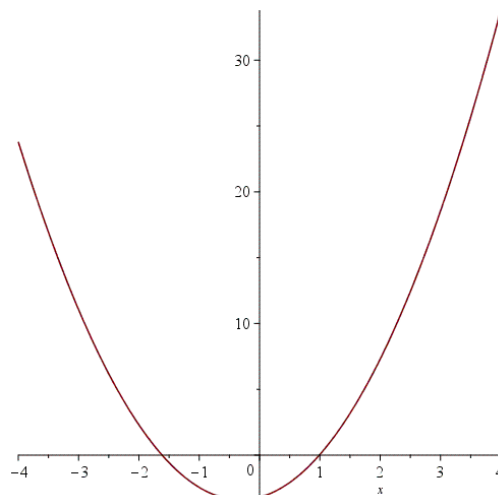
Сеанс работы с Maple:

> *restart* :

> *fn* :=  $2 * x^2 + 1.25 * x - 3.2$ ;

$$fn := 2x^2 + 1.25x - 3.2$$

> *plot*(*fn*, *x* = -4 ..4);



> *ans* := *evalf*(*solve*(*fn*, *x*), 5);

$$ans := 0.99044, -1.6154$$

> *x1* := *ans*[1];

```

> x1 := ans[2];
x1 := 0.99044
> x2 := ans[2];
x2 := -1.6154
> subs(x = ans[1], fn);
-7.213 10-6
> subs(x = ans[2], fn);
-0.00021568

```

## Задание 5

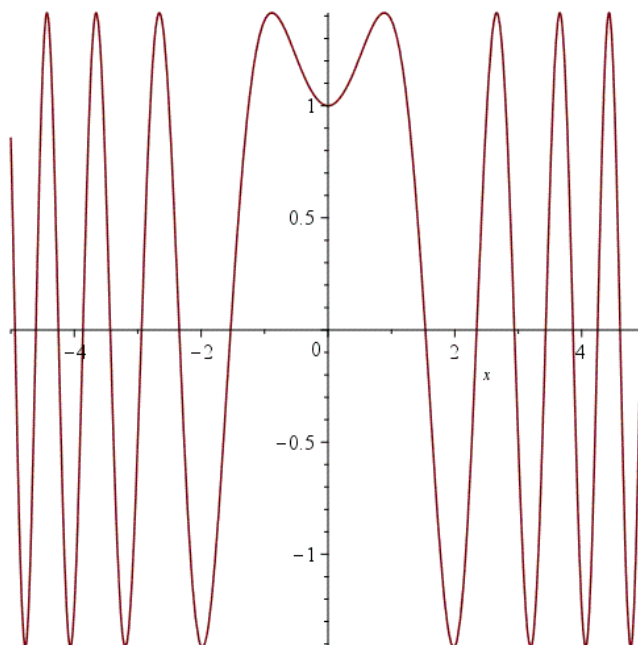
Решить тригонометрическое уравнение вида  $y = \sin(x^2) + \cos(x^2)$ . Выделить графически интервал изоляции корней уравнения на отрезке  $[-5..5]$  и вычислить все корни на этом отрезке.

Сеанс работы с Maple:

```

> restart;
> fn := sin(x^2) + cos(x^2);
fn := sin(x2) + cos(x2)
> plot(fn, x = -5 .. 5);

```



```

> ans := solve(fn, x);
ans :=  $\frac{1}{2} \sqrt{\pi}, -\frac{1}{2} \sqrt{\pi}$ 
> _EnvAllSolutions := true;
_EnvAllSolutions := true
> ans := solve(fn, x);
ans :=  $\frac{\sqrt{4 \pi \_Z1 \sim - \pi}}{2}, -\frac{\sqrt{4 \pi \_Z1 \sim - \pi}}{2}$ 

```

> x1 := ans[1];

$$x1 := \frac{\sqrt{4\pi Z1 - \pi}}{2}$$

> x2 := ans[2];

$$x2 := -\frac{\sqrt{4\pi Z1 - \pi}}{2}$$

## Задание 6

Рассмотрим возможности использования функции *solve* на примере расчета производственной программы.

Фирма производит четыре вида продукции (*A, B, C, D*), причем для выпуска каждого используется сырье 4 видов. Известно, что для выполнения плана было израсходовано сырья первого вида в количестве 65 кг, сырья второго вида – 122 кг, третьего вида – 80 кг, четвертого вида – 31 кг. Нормы расхода на изделие по каждому виду сырья представлены в таблице 2.

Таблица 2. – Нормы расхода на изделие по каждому виду сырья, кг.

Изделие	1	2	3	4	Общий расход сырья, кг
Сырье вида 1	1	1	3	5	65
Сырье вида 2	2	18	0	5	122
Сырье вида 3	1	5	2	6	80
Сырье вида 4	0	3	1	2	31
Количество изделий	$x_1$	$x_2$	$x_3$	$x_4$	

Требуется рассчитать план производства, обеспечивающий полный расход сырья.

Решение этой задачи сводится к решению системы четырех линейных уравнений с четырьмя неизвестными вида

$$\begin{cases} x_1 + x_2 + 3x_3 + 5x_4 = 65 \\ 2x_1 + 18x_2 + 5x_4 = 122 \\ x_1 + 5x_2 + 2x_3 + 6x_4 = 80 \\ 3x_2 + x_3 + 2x_4 = 31 \end{cases}$$

Решим эту систему, пользуясь функцией *solve* СКМ MAPLE.



- >  $eq1 := x1 + x2 + 3 * x3 + 5 * x4 = 65;$   
 $eq1 := x1 + x2 + 3 x3 + 5 x4 = 65$
- >  $eq2 := 2 * x1 + 18 * x2 + 5 * x4 = 122;$   
 $eq2 := 2 x1 + 18 x2 + 5 x4 = 122$
- >  $eq3 := x1 + 5 * x2 + 2 * x3 + 6 * x4 = 80;$   
 $eq3 := x1 + 5 x2 + 2 x3 + 6 x4 = 80$
- >  $eq4 := 3 * x2 + x3 + 2 * x4 = 31;$   
 $eq4 := 3 x2 + x3 + 2 x4 = 31$

- найдем корни:

- >  $rez := solve(\{eq1, eq2, eq3, eq4\}, \{x1, x2, x3, x4\});$   
 $rez := \{x1 = 10, x2 = 4, x3 = 7, x4 = 6\}$

#### 4. Задания для самостоятельной работы

**Задание 1.** Решить нелинейное уравнение (см. таблицу 3). Выделить графически интервал изоляции корня уравнения и вычислить корень с точностью  $\epsilon = 10^{-4}$ .

Таблица 3

Вариант	Уравнение	Вариант	Уравнение
1	$2 \sin \sqrt{x} + 0,28x - 1,2 = 0$	11	$2 \ln x^2 + 4,3 \ln x - 2,9 = 0$
2	$2e^{-x} - 2e^x - 1 = 0$	12	$x + \sin(x^{0,31} + 5,4) = 0$
3	$x - 2\sqrt{x} + 3 \cdot \sqrt[3]{x} - 4 = 0$	13	$\sin^2 \frac{x}{2} + \cos^2 \frac{x}{4} - 6x = 0$
4	$2,7 + \sin \sqrt{x} - x = 0$	14	$x - \sin(1 - 0,3x^3) = 0$
5	$\cos x - \frac{1}{2} \cos^2 x + \frac{1}{4} \cos^4 x - \frac{1}{8} = 0$	15	$2x \ln x - \sin x + \cos x = 0$
6	$2 - x + \cos x - \ln(x + 3) = 0$	16	$0,4 \cdot 2^x - 6,7x + 3 = 0$
7	$e^x + \ln 2x - 4,2x = 0$	17	$\ln x + x - 3,4 = 0$
8	$\sin \frac{2}{x} + \frac{1}{3} \cos \frac{4}{x} - \frac{3}{x} + \frac{1}{5} = 0$	18	$0,3x^2 - 2,1x \cdot \ln x = 0$
9	$2x + 14 - e^x + e^{-x} = 0$	19	$\sqrt{2-x} - \sqrt[4]{2+x} + 1 = 0$
10	$0,5x - \frac{2}{5 + \cos(3,2x)} = 0$	20	$2,8x - 1,3 \ln x - e^x = 0$

**Задание 2.** Решить неравенство (см. таблицу 4).

Таблица 4

Вар.	Неравенство	Вар.	Неравенство
1	$\log_x \frac{3x-1}{x^2+1} > 0$	11	$\sqrt{2x^2-3x-5} < x-1$
2	$\log_2^2(x-1) \log_{1/2}(x-1) > 5$	12	$4^{-x+1/2} - 7 \cdot 2^{-x} < 0$
3	$\log_{1/3} x < \log_3 x - \frac{5}{2} = 0$	13	$(x-1)\sqrt{x^2-x-2} \geq 0$
4	$\frac{x+4}{7} - \frac{x+7}{4} > -3$	14	$\sqrt{x+4} > \sqrt{2-\sqrt{3+x}}$
5	$\frac{x^2-2x+4}{5x^2+10x+4} < 0$	15	$  2x+1 -5  < 2$
6	$\left(\frac{15}{14}\right)^{ x+7 } < \left(\frac{15}{14}\right)^{ x^2-3x+2 }$	16	$\frac{81^x + 2 \cdot 25^{x \log_5 3} - 5}{(4x-1)^2} \geq 0$
7	$\log_{\frac{x^2-18x+91}{90}} \left(5x - \frac{3}{10}\right) \leq 0$	17	$x\sqrt{x} + 2\sqrt{x} + 3 \leq \frac{6}{2-\sqrt{x}}$
8	$\log_2(5-x) \cdot \log_{x+1} \frac{1}{8} \geq -6$	18	$\frac{2x^2}{x+3} + \frac{x+3}{x^2} \leq 3$
9	$\frac{\sqrt{2x-1} + \sqrt{x-3} - 3x + 10}{\sqrt{2x^2-7x+3}} > 2$	19	$8^{x+1} - 20 \cdot 2^x + 12 > 0$
10	$ 3^{1+\sqrt{3-x}}  \leq 4^{\sqrt{3-x}}$	20	$\sqrt{1+x^2} - x \leq \frac{5}{2\sqrt{1+x^2}}$

**Задание 3.** Решить систему уравнений (см. таблицу 5).

Таблица 5

Вар.	СЛАУ	Вар.	СЛАУ
1	$\begin{cases} x_1 + x_2 + 2x_3 + 3x_4 = 1; \\ 3x_1 - x_2 - x_3 - 2x_4 = -4; \\ 2x_1 + 3x_2 - x_3 - x_4 = -6; \\ x_1 + 2x_2 + 3x_3 - x_4 = -4. \end{cases}$	11	$\begin{cases} 5x + 8y - z = -7; \\ x + 2y + 3z = 1; \\ 2x - 3y + 2z = 9. \end{cases}$
2	$\begin{cases} x_1 + 2x_2 + 3x_3 - 2x_4 = 6; \\ x_1 - x_2 - 2x_3 - 3x_4 = 8; \\ 3x_1 + 2x_2 - x_3 + 2x_4 = 4; \\ 2x_1 - 3x_2 + 2x_3 + x_4 = -8. \end{cases}$	12	$\begin{cases} x + 2y + z = 4; \\ 3x - 5y + 3z = 1; \\ 2x + 7y - z = 8. \end{cases}$
3	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 5; \\ 2x_1 + x_2 + 2x_3 + 3x_4 = 1; \\ 3x_1 + 2x_2 + x_3 + 2x_4 = 1; \\ 4x_1 + 3x_2 + 2x_3 + x_4 = -5. \end{cases}$	13	$\begin{cases} 3x + 2y + z = 5; \\ 2x + 3y + z = 1; \\ 2x + y + 3z = 11. \end{cases}$
4	$\begin{cases} x_2 - 3x_3 + 4x_4 = -5; \\ x_1 - 2x_3 + 3x_4 = -4; \\ 3x_1 + 2x_2 - 5x_4 = 12; \\ 4x_1 + 3x_2 - 5x_3 = 5. \end{cases}$	14	$\begin{cases} x + 2y + 4z = 31; \\ 5x + y + 2z = 29; \\ 3x - y + z = 10. \end{cases}$
5	$\begin{cases} x_1 + 3x_2 + 5x_3 + 7x_4 = 12; \\ 3x_1 + 5x_2 + 7x_3 + x_4 = 0; \\ 5x_1 + 7x_2 + x_3 + 3x_4 = 4; \\ 7x_1 + x_2 + 3x_3 + 5x_4 = 16. \end{cases}$	15	$\begin{cases} 4x - 3y + 2z = 9; \\ 2x + 5y - 3z = 4; \\ 5x + 6y - 2z = 18. \end{cases}$
6	$\begin{cases} x_1 + 5x_2 + 3x_3 - 4x_4 = 20; \\ 3x_1 + x_2 - 2x_3 = 9; \\ 5x_1 - 7x_2 + 10x_4 = -9; \\ 3x_2 - 5x_3 = 1. \end{cases}$	16	$\begin{cases} 2x - y - z = 4; \\ 3x + 4y - 2z = 11; \\ 3x - 2y + 4z = 11. \end{cases}$
7	$\begin{cases} 2x_1 + x_2 - 5x_3 + x_4 = 8; \\ x_1 - 3x_2 - 6x_4 = 9; \\ 2x_2 - x_3 + 2x_4 = -5; \\ x_1 + 4x_2 - 7x_3 + 6x_4 = 0. \end{cases}$	17	$\begin{cases} x + y + 2z = -1; \\ 2x - y + 2z = -4; \\ 4x + y + 4z = -2. \end{cases}$
8	$\begin{cases} x_1 - x_2 + 3x_3 + 2x_4 = 4; \\ 3x_1 + 3x_2 + 3x_3 + 2x_4 = 6; \\ 3x_1 - x_2 - x_3 + 2x_4 = 6; \\ 3x_1 - x_2 + 3x_3 - x_4 = 6. \end{cases}$	18	$\begin{cases} 3x - y = 5; \\ -2x + y + z = 0; \\ 2x - y + 4z = 15. \end{cases}$

9	$\begin{cases} x_1 + 2x_2 - x_3 + x_4 = 8; \\ 2x_1 + x_2 + x_3 + x_4 = 5; \\ x_1 - x_2 + 2x_3 + x_4 = -1; \\ x_1 + x_2 - x_3 + 3x_4 = 10. \end{cases}$	19	$\begin{cases} 3x - y + z = 4; \\ 2x - 5y - 3z = -17; \\ x + y - z = 0. \end{cases}$
10	$\begin{cases} 4x_1 + x_2 - x_4 = -9; \\ x_1 - 3x_2 + 4x_3 = -7; \\ 3x_2 - 2x_3 + 4x_4 = 12; \\ x_1 + 2x_2 - x_3 - 3x_4 = 0. \end{cases}$	20	$\begin{cases} x + y + z = 2; \\ 2x - y - 6z = -1; \\ 3x - 2y = 8. \end{cases}$

**Задание 4.** Решить систему неравенств (см. таблицу 6).

Таблица 6

<b>Вариант</b>	<b>Условие</b>	<b>Вариант</b>	<b>Условие</b>
1	$\begin{cases} 3-x > 0, \\ 9-x^2 > 0 \end{cases}$	6	$\begin{cases} 2x^2 - x > 0, \\ 2x + 2 > 0, \\ \log_4(2x + 2) \neq 0; \end{cases}$
2	$\begin{cases} 3-x > 0, \\ 9-2^x > 0; \end{cases}$	7	$\begin{cases}  x^2 + 5x  < 6, \\  x+1  < 2; \end{cases}$
3	$\begin{cases} \frac{x}{x-3} + \frac{x-5}{x} < \frac{2x}{3-x} \\ \log_{\sqrt{2}}(x-1) < 4 \end{cases}$	8	$\begin{cases}  x^2 - 4x  < 5 \\  x+1  < 3 \end{cases}$
4	$\begin{cases} \frac{x+8}{x+2} > 2 \\ \lg(x+1) < 1 \end{cases}$	9	$\begin{cases} \sqrt{4x-7} < x \\ \sqrt{x+5} + \sqrt{5-x} > 4; \end{cases}$
5	$\begin{cases} \sqrt{x^2 - 9x + 20} \leq \sqrt{x-1} \\ \sqrt{x-1} \leq \sqrt{x^2 - 13} \end{cases}$	10	$\begin{cases} x^2 - 3x < 2 - x, \\ x^2 - 3x > x - 2; \end{cases}$

**Задание 5. Построить поверхность (см. таблицу 7)**

*Таблица 7*

<b>Вариант</b>	<b>Условие</b>	<b>Вариант</b>	<b>Условие</b>
1	$f = 2 \cdot \sin(xy)$ , при $x = -\pi.. \pi$ , $y = -\pi.. \pi$	6	$f = \sin(x^2 + (y - 1)^2)$ при $x = -2..2$ , $y = -1..3$
2	$f = \cos(x+y)^{-1}$ при $x = -4..4$ , $y = -4..4$	7	$f = (e^{xy})^{xy}$ при $x = -1..1$ , $y = -1..1$
3	$f = \cos(x) \cdot \sin(2y)$ при $x = -\pi.. \pi$ , $y = -\pi.. \pi$	8	$f = \sin((x+2)t)$ при $x = -10..10$ , $t = 1..20$
4	$f = \sin(xy)$ при $x = -\pi.. \pi$ , $y = -\pi.. \pi$	9	$f = \sin(x) \cdot \cos(x) \cdot \tan(xy)$ при $x = -4..4$ , $y = -4..4$
5	$f = \frac{\sin(x + t \cdot \pi)}{x + 11}$ при $x = -10..10$ , $t = 1..4$	10	$f = \frac{\ln(x + y)}{\cos(x + y)}$ при $x = -4..4$ , $y = -4..4$

## Библиографический список

1. Intranet – ресурс. Электронные учебные пособия кафедры «Информатика» УО ВГТУ. <http://192.168.40.69>
2. Вардомацкая, Е. Информатика. В двух частях. Часть I.: учебное пособие / Е.Ю. Вардомацкая, Т.Н. Окишева. – Витебск: УО «ВГТУ», 2007. – 220 с.
3. Вардомацкая Е.Ю. Электронный учебно-методический комплекс по дисциплине «Основы информатики и вычислительной техники» (CD-RW, <http://test>), 2008 г.
4. Дьяконов, В. Maple 6 : учебный курс – Санкт-Петербург : Питер, 2001. – 608 с. : ил
5. Информатика для юристов и экономистов: Учебник для вузов / Под ред. С.В. Симоновича. – СПб: Питер, 2006.
6. Морозевич А.Н. и др. Прикладная информатика: Учебное пособие – Мн.: Выш. школа, 2003. – 335 с.: ил
7. Шарстнев В.Л., Вардомацкая Е.Ю. Компьютерные информационные технологии. Пакеты прикладных программ для моделирования а анализа задач экономики : пособие / В.Л. Шарстнев, Е.Ю. Вардомацкая – Витебск: УО «ВГТУ», 2007. – 138 с.
8. Шарстнев В.Л., Вардомацкая Е.Ю., Компьютерные информационные технологии: лабораторный практикум : пособие – Витебск: УО «ВГТУ», 2008. – 170 с.

**МИНОБРНАУКИ РОССИИ**  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра машиностроительных технологий и оборудования

УТВЕРЖДАЮ  
Проректор по учебной работе  
О.Г. Жолдинова  
« 14 » \_\_\_\_\_ 20\_\_ г.



**ИНТЕРПОЛЯЦИЯ ВСТРОЕННЫМИ  
ПРОЦЕДУРАМИ MAPLE**

Методические указания к выполнению лабораторной  
и самостоятельной работы для студентов по направлению  
подготовки 15.03.01

Курск 2023



УДК 519.6

Составитель Куц В.В.

Рецензент

Кандидат технических наук, доцент А.Н. Гречухин

**Интерполяция встроенными процедурами Maple:** методические указания к выполнению лабораторной и самостоятельной работы для студентов по направлению подготовки 15.03.01 / Минобрнауки России, Юго-Зап. гос. ун-т; сост.: В.В. Куц. – Курск, 2023. – 11 с.

Содержат сведения по вопросам выполнения процедуры интерполяции встроенными процедурами Maple. Указывается порядок выполнения лабораторной работы и правила оформления отчета.

Предназначены для студентов по направлению подготовки 15.03.01 всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.

Усл.печ. л. 0,64 . Уч.-изд. л. 0,58.

Тираж 100 экз. Заказ . Бесплатно.

Юго-Западный государственный университет.

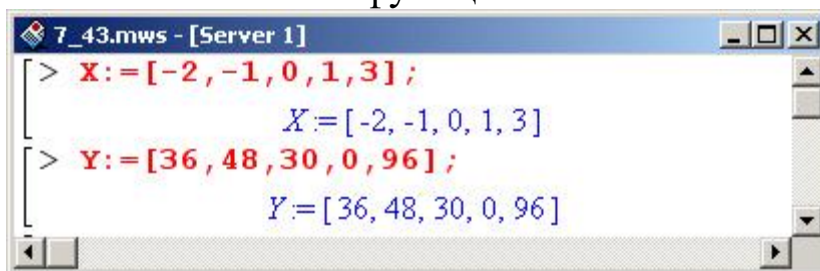
305040, г. Курск, ул. 50 лет Октября, 94.

## 1. Краткие теоретические сведения

### 1.2. Встроенные процедуры Maple

Было бы удивительно, если разработчики Maple обошли стороной вопросы, связанные с проблемой интерполирования функций. В частности, для построения интерполяционного полинома в Maple предусмотрена процедура `interp()`. Процедура вызывается с тремя параметрами: списком значений узловых точек, списком значений интерполируемой функции в этих точках, а также названием переменной, которую следует использовать при построении интерполяционного полинома. Например, построим полином по значениям, приведенным в табл. 1.

В соответствии со значениями, приведенными в таблице, создаем два списка: сначала список  $X$  с узловыми точками, а затем список  $Y$  со значениями функции в этих точках.

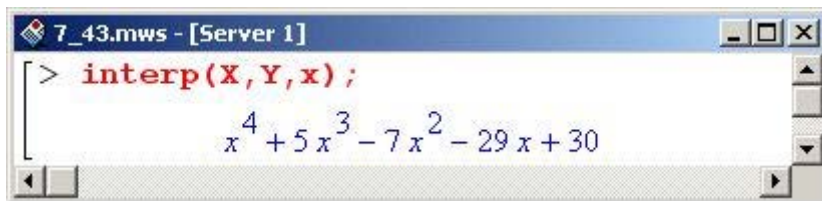


```
> X:=[-2, -1, 0, 1, 3];
X:=[-2, -1, 0, 1, 3]
> Y:=[36, 48, 30, 0, 96];
Y:=[36, 48, 30, 0, 96]
```

Таблица 1. Табулированные значения для интерполируемой функции.

Узловая точка	Значение функции
-2	36
-1	48
0	30
1	0
3	96

После этого можно вызвать процедуру интерполирования, указав описанные выше списки ее параметрами и, кроме того, приняв  $x$  за переменную интерполирования.



```
> interp(X, Y, x);  
x4 + 5x3 - 7x2 - 29x + 30
```

Стоит заметить, что в качестве результата процедурой `interp()` возвращается выражение, а не оператор, как это было выше при разработке процедуры построения интерполяционного полинома Лагранжа.

Еще один достаточно популярный способ интерполяции состоит в следующем. На каждом интервале между соседними узловыми точками интерполируемая функция представляется в виде полинома. Но в отличие от, скажем, интерполяции Лагранжа, где один и тот же полином используется для всех точек, в данном случае на каждом интервале полином свой. Кроме равенства интерполяционной функции в узлах табличным значениям функции интерполируемой, на первую накладывается еще и условие непрерывности производных до порядка, на единицу меньшего, чем степень интерполяционных полиномов. Подобный тип интерполяции называется интерполяцией сплайнами, или сплайн-интерполяцией. Наибольшей популярностью пользуется интерполяция кубическими сплайнами.

В Maple для выполнения интерполяции сплайнами может быть использована процедура `spline ()`. Процедура имеет три обязательных параметра. Первые два – это списки с узловыми точками и значениями функции соответственно. Третьим параметром указывается переменная интерполирования. Если четвертый необязательный параметр не указан, то интерполяция будет выполняться кубическими сплайнами, т.е. для "сшивки" узловых точек будут использоваться полиномы третьей степени. Например, если параметрами процедуры указать ранее рассмотренные списки, табулирующие значения функции в узловых точках, получим такой результат.

```

7_43.mws - [Server 1]
> spline(X,Y,x);

$$\left[ \begin{array}{l} \frac{960}{43} - \frac{2454}{43}x - \frac{1620}{43}x^2 - \frac{270}{43}x^3 \\ 30 - \frac{1464}{43}x - \frac{630}{43}x^2 + \frac{60}{43}x^3 \\ 30 - \frac{1464}{43}x - \frac{630}{43}x^2 + \frac{804}{43}x^3 \\ \frac{2391}{43} - \frac{4767}{43}x + \frac{2673}{43}x^2 - \frac{297}{43}x^3 \end{array} \right. \begin{array}{l} x < -1 \\ x < 0 \\ x < 1 \\ otherwise \end{array}$$


```

Четвертым параметром процедуры spline () может быть либо одно из зарезервированных ключевых слов из набора linear (интерполяция линейными зависимостями), quadratic (интерполяция параболой), cubic (кубический сплайн), quartic (интерполяция полиномами четвертой степени), либо целое неотрицательное число, определяющее степень интерполяционного полинома. Причем указание числа от 1 до 4 эквивалентно использованию перечисленных текстовых опций, согласно того порядка, как они были представлены выше.

### Внимание!

В качестве четвертого параметра может быть указано любое целое положительное число, не только в диапазоне от 1 до 4. Просто для этих чисел существует альтернативный вызов через текстовую опцию. Например, командой spline(X,Y,x,5) можно выполнить интерполяцию рассмотренной выше функции полиномами пятой степени и т.д.

Интерполяция полиномами четвертой степени все той же функции выглядит следующим образом.

```

43.mws - [Server 1]
spline(X, Y, x, 4);
[
  - 6424996 / 691641 - 96669332 / 691641 x - 27002432 / 230547 x^2 - 27002432 / 691641 x^3 - 3375304 / 691641 x^4   x < -3/2
  20677604 / 691641 - 24395732 / 691641 x - 2911232 / 230547 x^2 + 5119168 / 691641 x^3 + 21272 / 7437 x^4   x < -1/2
  30 - 7940908 / 230547 x - 779408 / 76849 x^2 + 2470400 / 230547 x^3 + 3124312 / 691641 x^4   x < 1/2
  20301884 / 691641 - 20243956 / 691641 x - 5916992 / 230547 x^2 + 21726272 / 691641 x^3 - 14456 / 2479 x^4   x < 2
  381356 / 2077 - 77874428 / 230547 x + 15809280 / 76849 x^2 - 10539520 / 230547 x^3 + 2634880 / 691641 x^4   otherwise
]

```

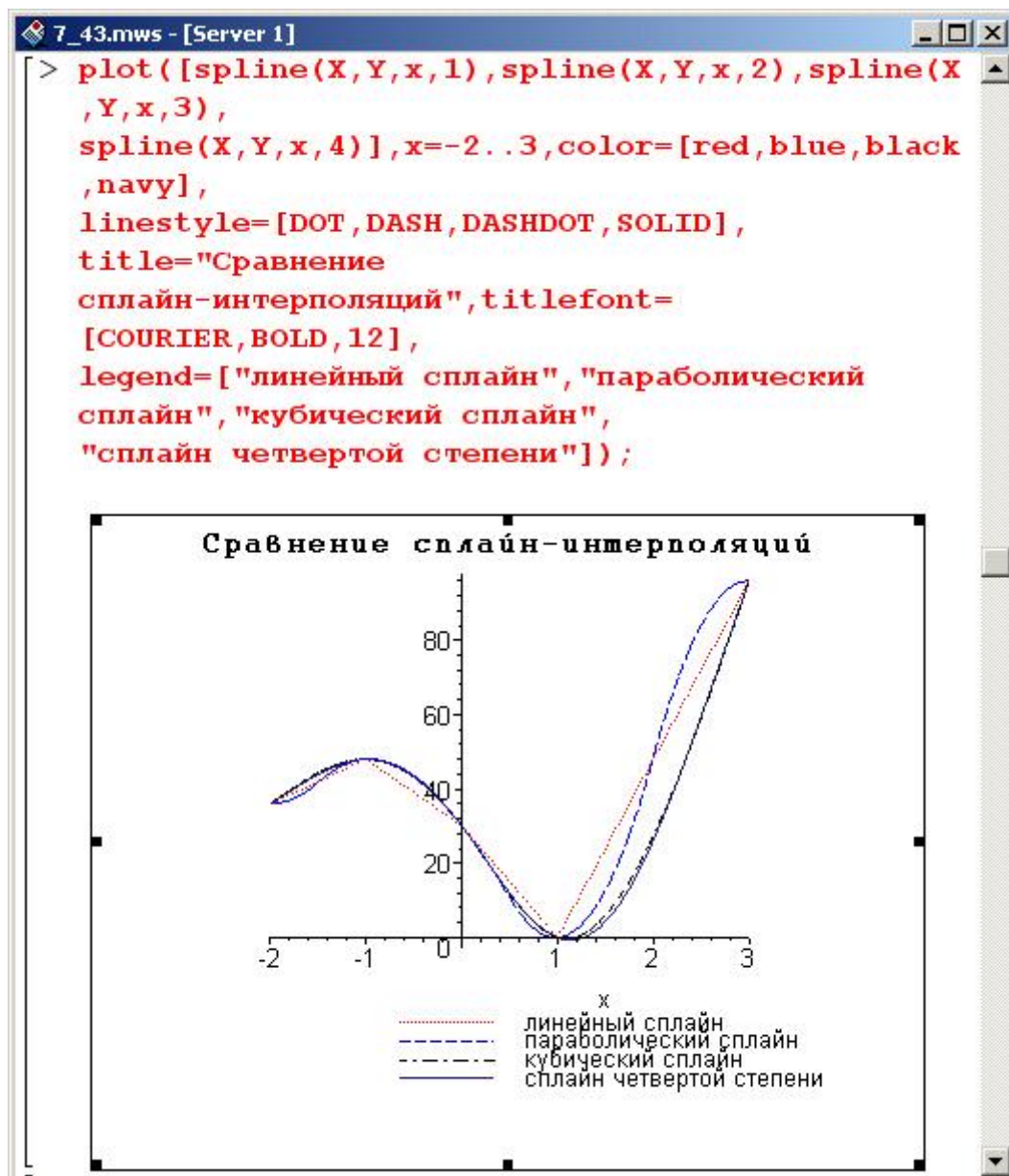
Выражения достаточно громоздки. Посмотрим, что будет, если использовать в качестве сплайн-полиномов линейные зависимости.

```

7_43.mws - [Server 1]
> spline(X, Y, x, linear);
[
  60 + 12 x   x < -1
  30 - 18 x   x < 0
  30 - 30 x   x < 1
  -48 + 48 x   otherwise
]

```

По сравнению с предыдущим случаем, здесь получен более простой результат. Однако простота – далеко не всегда значит эффективность. В этом несложно убедиться, если построить графики для интерполяционных функций, получаемых при сплайн-интерполяции полиномами разных степеней.

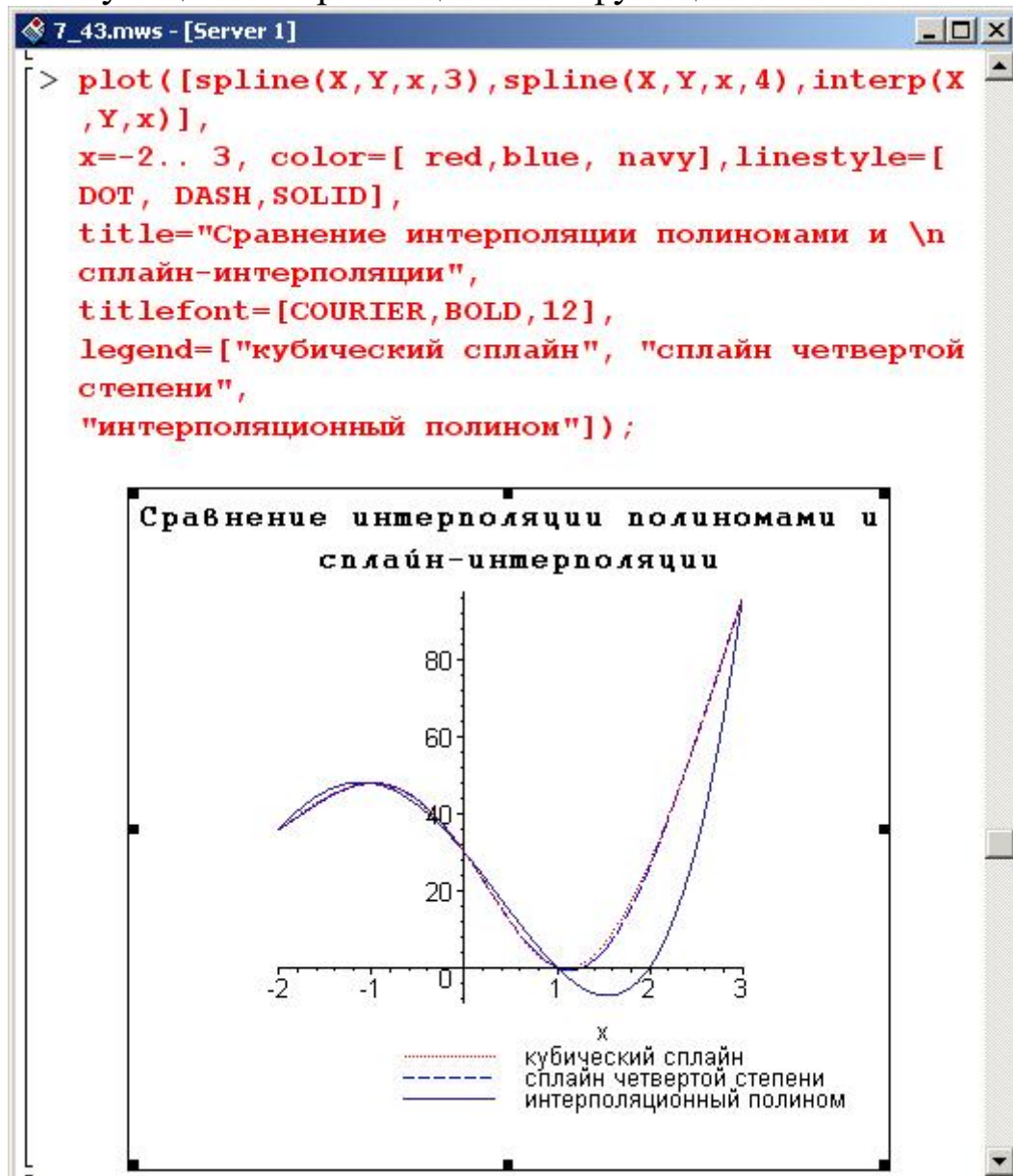


Очевидно, что линейная сплайн-интерполяция является достаточно грубой. По большому счету, это просто соединение интерполяционных точек линиями. Такой тип интерполяции используется крайне редко. Что касается использования полиномов прочих степеней, начиная со второй, то визуальной особенности между ними (во всяком случае в данном примере) нет.

Однако не следует забывать, что степень сплайн-полиномов определяет гладкость полученных кривых. Это важно особенно в тех случаях, когда от полученных в результате интерполяции функций следует брать производные. В этом смысле полиномиальная интерполяция по сравнению со сплайн-интерполяцией обладает



тем преимуществом, что и интерполяционный полином, и производные от него однозначно являются функциями непрерывными и гладкими. Поэтому интересно сравнить результаты интерполяции полиномом и сплайн-интерполяции. Для этого построим графики соответствующих интерполяционных функций.



Можно видеть, что все три графика (сплайны третьей и четвертой степени, а также интерполяционный полином четвертой степени) практически совпадают, особенно в левой части рисунка. Правда, интерполяционный полином является более гладкой функцией по сравнению с первыми двумя.

## 2. Задание:

1. Вычислить значения заданной функции (см. таблицу)  $y_i = f(x_i)$  в узлах интерполяции  $x_i = a + h \cdot i$ , где  $h = (b - a)/10$ ,  $i = 0, 1, \dots, 5$ , на отрезке  $[a, b]$ .

2. Для вычисленных значений построить интерполяционную функцию `spline` (при интерполяции линейными зависимостями, интерполяции параболоми, кубическом сплайне) и `interp`.

3. Построить графики исходной функции и полученных интерполяционных функций на отрезке  $[a, b]$  и сравнить.

№ варианта	$f(x)$	$[a, b]$	№ варианта	$f(x)$	$[a, b]$
1	$\sin x^2$	[0, 2]	9	$x \cdot \cos(x + \ln(1 + x))$	[1, 5]
2	$\cos x^2$	[0, 2]	10	$10 \cdot \ln 2x / (1 + x)$	[1, 5]
3	$e^{\sin x}$	[0, 5]	11	$\sin x^2 \cdot e^{-(x/2)^2}$	[0, 3]
4	$1/(0.5 + x^2)$	[0, 2]	12	$\cos(x + \cos^3 x)$	[0, 2]
5	$e^{-(x + \sin x)}$	[2, 5]	13	$\cos(x + e^{\cos x})$	[3, 6]
6	$1/(1 + e^{-x})$	[0, 4]	14	$\cos(2x + x^2)$	[0, 1]
7	$\sin(x + e^{\sin x})$	[0, 3]	15	$e^{\cos x} \cdot \cos x^2$	[0, 2]
8	$e^{-(x + 1/x)}$	[1, 3]			

## Пример выполнения работы

Рассмотрим функцию

$$f(x) = x^2 e^{-x^2}$$

на отрезке от  $[0; 2.5]$  для случая с 5-ю интервалами.

Ниже приведен код программы в Maple/

> `restart : readlib(spline) : with(plots) : with(plottools) :`

> `f := x -> x^2 * e^-x^2;`

$$f := x \rightarrow x^2 e^{-x^2}$$

> `a := 0; b := 3;`

$$a := 0$$

$$b := 3$$

> `h := (b - a) / 5;`



$$h := \frac{3}{5}$$

>  $X := [a + h \cdot i \mid i = 0..5];$

$$X := \left[ 0, \frac{3}{5}, \frac{6}{5}, \frac{9}{5}, \frac{12}{5}, 3 \right]$$

>  $Y := \text{evalf}(\text{map}(f, X));$

$$Y := [0., 0.2511634774, 0.3411759725, 0.1268910201, 0.01815040280, 0.001110688237]$$

>

>

>  $f1 := \text{spline}(X, Y, x, \text{linear});$

$$f1 := \begin{cases} 0.4186057957x & x < \frac{3}{5} \\ 0.1611509823 + 0.1500208252x & x < \frac{6}{5} \\ 0.7697458773 - 0.3571415873x & x < \frac{9}{5} \\ 0.4531128720 - 0.1812343622x & x < \frac{12}{5} \\ 0.08630926105 - 0.02839952427x & \text{otherwise} \end{cases}$$

>  $f2 := \text{spline}(X, Y, x, \text{quadratic});$

$$f2 := \begin{cases} -1.38777878078145 \cdot 10^{-16}x + 1.05829508139 \\ 0.00848202640198481 + 0.404469084996692x - 0.3841799 \\ 0.523945604476182 - 0.152308026646818x - 0.54378191 \\ 0.701278172908410 - 0.319103973782450x + 0.26578866 \\ 0.225923031580353 - 0.0865719286584805x + 0.12176474 \\ 0.00111068823700000 + 1.44560289664734 \cdot 10^{-18}x + 0.02252 \end{cases}$$

>  $f3 := \text{spline}(X, Y, x, 3);$

$$f3 := \begin{cases} 0.451536552115152x - 0. \\ 0.0395169077381818 + 0.352744282769697x - 0.16465378 \\ 0.529136557332727 - 0.156633820693939x - 0.68430972 \\ 0.752519335789091 - 0.347571286493939x + 0.36608061 \\ 0.181851719192727 - 0.0682088818303030x + 0.0995233939 \end{cases}$$

>  $f4 := \text{interp}(X, Y, x)$

$$f4 := -0.1046675093x^5 + 0.8057919153x^4 - 2.069295831x^3 + 1.809438790x^2 - 0.0825971238x$$

>  $\text{plot}([f(x), f1, f2, f3, f4], x = a..b, \text{color} = \text{black}, \text{linestyle} = [\text{DOT}, \text{SOLID}, \text{DASHDOT}, \text{DASH}, \text{longdash}], \text{legend} = [ \text{"Исходная функция"}, \text{"Линейный сплайн"}, \text{"Параболический сплайн"}, \text{"Кубический сплайн"}, \text{"Интерполяционный полином"} ])$

