

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Техническая эксплуатация автомобилей»

ИНФОРМАТИКА

*Методические рекомендации к лабораторным работам
для студентов специальностей 1-37 01 06 «Техническая
эксплуатация автомобилей (по направлениям)»
и 1-37 01 07 «Автосервис» дневной и заочной форм обучения*

Часть 2



Могилев 2019

УДК 004.41/42
ББК 32.973-018
И 146/2

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Техническая эксплуатация автомобилей»
«16» мая 2019 г., протокол № 10

Составители: ст. преподаватель О. А. Пономарева;
ассистент С. Ю. Билык

Рецензент Ю. С. Романович

Методические рекомендации к лабораторным работам для студентов специальностей 1-37 01 06 «Техническая эксплуатация автомобилей (по направлениям)» и 1-37 01 07 «Автосервис» дневной и заочной форм обучения.

Учебно-методическое издание

ИНФОРМАТИКА

Часть 2

Ответственный за выпуск	О. В. Билык
Технический редактор	А. А. Подошевка
Компьютерная верстка	Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 56 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».

Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.

Пр-т Мира, 43, 212022, Могилев.

© Белорусско-Российский
университет, 2019



Содержание

12 Лабораторная работа № 12. Графическая система AutoCAD.....	4
13 Лабораторная работа № 13. Графическая система AutoCAD.....	8
14 Лабораторная работа № 14. Графическая система AutoCAD.....	11
15 Лабораторная работа № 15. Графическая система AutoCAD.....	13
16 Лабораторная работа № 16. Программирование на алгоритмическом языке	16
17 Лабораторная работа № 17. Программирование на алгоритмическом языке	18
18 Лабораторная работа № 18. Программирование на алгоритмическом языке	24
19 Лабораторная работа № 19. Программирование на алгоритмическом языке	26
20 Лабораторная работа № 20. Программирование на алгоритмическом языке	28
21 Лабораторная работа № 21. Программирование на алгоритмическом языке	30
22 Лабораторная работа № 22. Программирование на алгоритмическом языке	33
23 Лабораторная работа № 23. Программирование на алгоритмическом языке	37
24 Лабораторная работа № 24. Программирование на алгоритмическом языке	43
Список литературы	44



Часть 2

12 Лабораторная работа № 12. Графическая система AutoCAD

12.1 Основы работы в AutoCAD. Основные примитивы AutoCAD

Цель работы: получение навыков работы в программе AutoCad; изучение основных элементов пользовательского интерфейса и основных примитивов.

Основные элементы интерфейса AutoCAD.

Панель быстрого доступа, на которой расположены часто используемые команды для работы с чертежами (*Сохранить*, *Новый* и т. д.). На данную панель можно добавлять любые команды на усмотрение пользователя.

Рабочее пространство (выбор интерфейса) – выпадающая вкладка выбора интерфейса. По умолчанию AutoCAD откроется в рабочем пространстве *Рисование и аннотации*.

Модель\лист – вкладки переключения модель/лист.

Строка состояния содержит функции точности построения, а также инструменты для работы с масштабом, аннотациями и привязками.

Видовой куб представляет собой переключатель между 3D-видами.

На **вкладках ленты** находятся все основные инструменты для работы.

Задание 1

Создать рамку с основной надписью листа формата А3 (297 × 420 мм).

1 Запускаем инструмент «Прямоугольник» (Рисование → Прямоугольник) или нажимаем на пиктограмму на панели «Рисование». Вводим координаты нижнего левого угла в командную строку через запятую (0,0) и нажимаем «Enter», чтобы подтвердить ввод. Размеры прямоугольника вводим набором с клавиатуры в динамические окна рядом с курсором (ширину 420, высоту 297). Для переключения между полями используем запятую или клавишу «Tab».

2 Аналогично строим рамку чертежа и границу основной надписи чертежа. В полученном поле листа рисуем основную рамку чертежа, введя координату вершины (20; 5), а затем ширину (395 мм) и высоту (287 мм) прямоугольника.

3 Строим таблицу основной надписи чертежа, используя координаты вершин и длины сторон прямоугольников, приведенных на рисунке 12.1.

4 Производим построение внутренних линий таблицы, воспользовавшись инструментом «Отрезок» (Рисование → Отрезок). Для построения отрезка необходимо иметь координату первой точки, длину и угол наклона радиус-вектора, соединяющего первую точку со второй. Вводим координаты начала (295; 20), нажимаем «Enter», указываем длину вектора 120, нажимаем «Tab», вводим угол поворота в полярной системе координат 0°. Чтобы прервать построение, нажимаем «Esc». Строим вторую линию с началом в координатах (365; 25) и радиус-вектором длиной 50 под углом 0°.



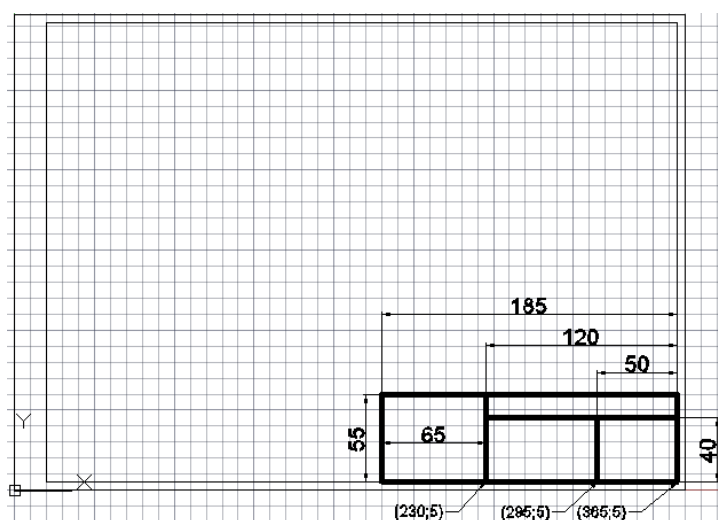


Рисунок 12.1 – Построение основной надписи

Инструмент «Отрезок» также позволяет строить вторую точку прямой, используя относительные координаты. Вносим координаты начала отрезка (365; 40), а координаты конечной точки вводим, используя относительные координаты. Заносим значение 50 (абсцисса второй точки отрезка), нажимаем «,» и вводим второе значение 0 (ординату второй точки отрезка). Результат построений представлен на рисунке 12.2.

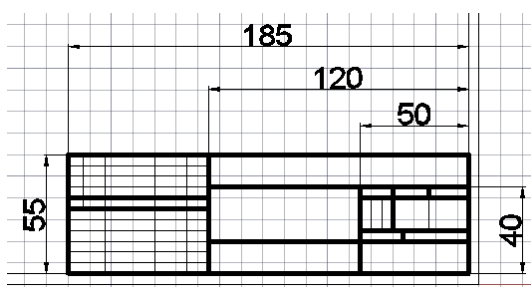


Рисунок 12.2 – Построение отрезков

5 Используя инструмент «Отрезок», строим недостающие линии рамки основной надписи чертежа. Размеры полей представлены на рисунке 12.3 в соответствии с ГОСТ 2.104–2006.

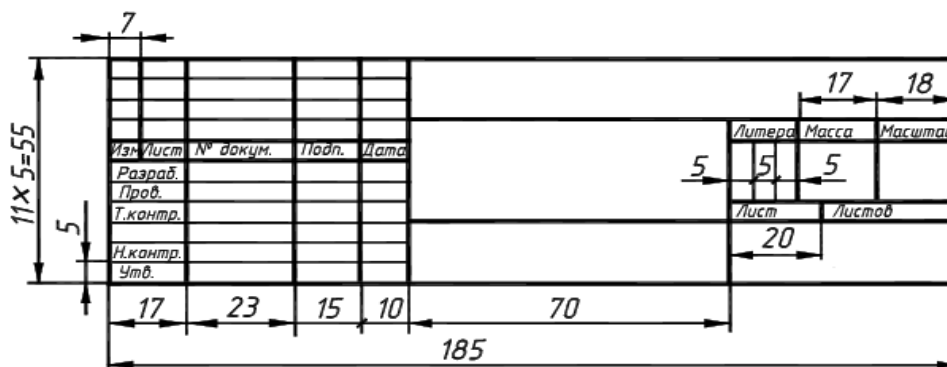


Рисунок 12.3 – Основная рамка чертежа ГОСТ2.104–2006

6 Используя инструмент «Выделить рамкой», выделяем все линии основной надписи чертежа. Затем, используя падающее меню, устанавливаем вес линий 1 мм. Аналогично изменяем толщину тонких линий основной надписи, назначив им вес 0,5 мм. Полученный результат представлен на рисунке 12.4.

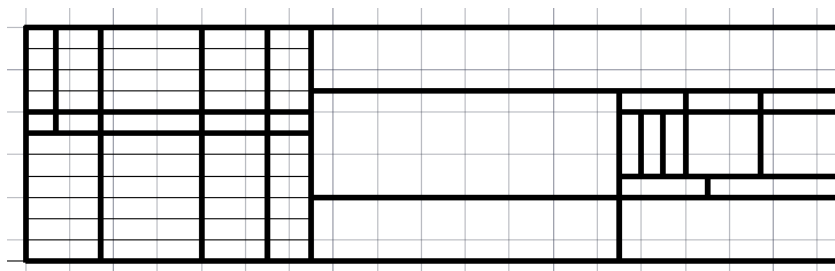


Рисунок 12.4 – Результат изменения веса линий

7 Для заполнения полей основной надписи используем инструмент «Многострочный» (Рисование → Текст → Многострочный) или нажимаем на пиктограмму «А» на панели «Рисование». Подводим курсор к нижнему левому углу ячейки, которую собираемся заполнить, и нажимаем левую кнопку мыши. Используя появившуюся динамическую рамку, определяем область для размещения текста и вводим текст высотой 3,5 мм. Чтобы отступить от линии ячейки, нажимаем пробел. Для редактирования введенного текста дважды щелкаем по нему левой кнопкой мыши, выделяем фрагмент для редактирования и устанавливаем требуемые параметры. Если текст по длине не помещается в ячейку, то нужно воспользоваться полем «Степень сжатия». Заполняем остальные блоки таблицы в соответствии с рисунком 12.3.

8 Сохраняем чертеж рамки с основной надписью в качестве шаблона чертежа (формат *.dwt). Для этого необходимо выполнить *Файл → Сохранить как → Шаблон чертежа AutoCAD*. Позже можно открыть существующий файл шаблона чертежа, изменить его, а затем сохранить его снова, к примеру, под другим именем.

Задание 2

Используя примитив **Отрезок**, начертить следующие эскизы в ранее созданной рамке (рисунок 12.5).

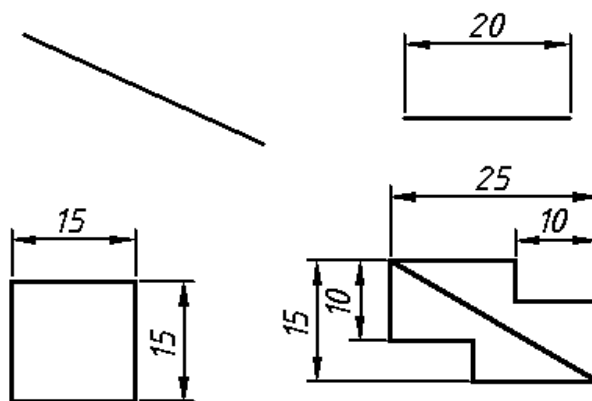


Рисунок 12.5 – К заданию 2

Задание 3

Используя примитивы **Прямоугольник** и **Круг**, начертить следующие эскизы в ранее созданной рамке (рисунок 12.6).

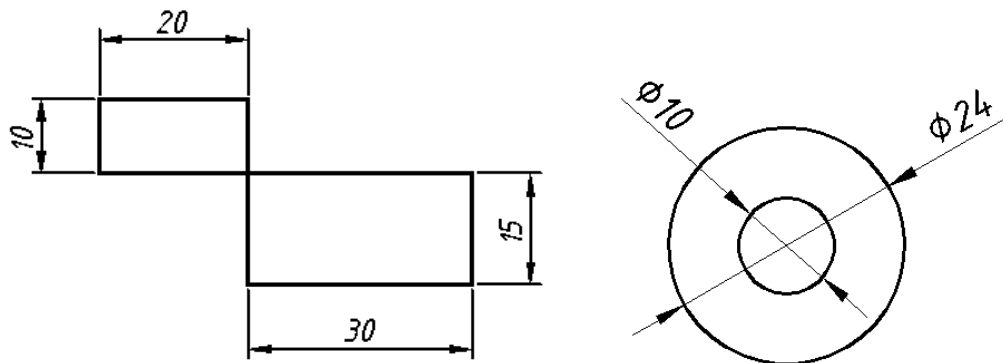


Рисунок 12.6 – К заданию 3

Задание 4

Используя различные примитивы, начертить следующий эскиз в ранее созданной рамке (рисунок 12.7).

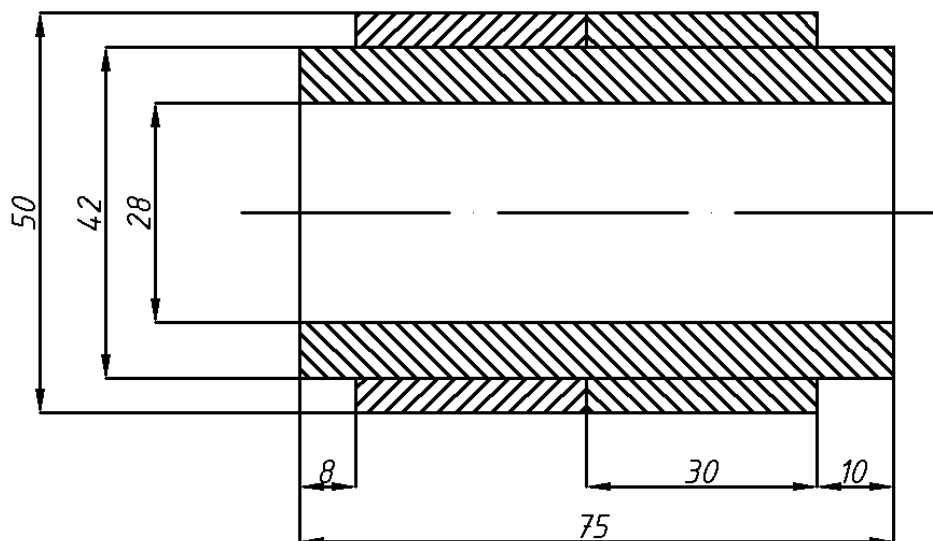


Рисунок 12.7 – К заданию 4

Контрольные вопросы


- 1 Назначение графической системы AutoCad.
- 2 Что называется шаблоном чертежа?
- 3 Как настроить размерный стиль в AutoCad?
- 4 Как настроить слои в AutoCad?
- 5 Перечислите основные примитивы в AutoCad.

13 Лабораторная работа № 13. Графическая система AutoCAD

13.1 Использование объектных привязок и изучение команд редактирования в AutoCad

Цель работы: приобретение навыков использования объектных привязок и редактирования объектов в AutoCad.

Объектная привязка позволяет строить новые точки на чертеже, привязываясь к характерным точкам существующих объектов или относительно них.

В процессе работы режим *Объектная привязка* можно включать или отключать по мере необходимости, используя кнопку , расположенную в строке состояния, или клавишу *F3*.

Возможны следующие режимы объектной привязки.

Конточка – привязка к ближайшей конечной точке линии или дуги, границы области или трехмерного тела.

Середина – средняя точка прямой, дуги.

Центр – центр окружности, дуги, эллипса.

Узел – привязка к точечному элементу.

Квадрант – привязка к точке квадранта дуги, эллиптической дуги, эллипса или круга (точка, лежащая на пересечении осевой линии и окружности).

Пересечение – пересечение двух прямых, прямой с дугой или окружностью, двух окружностей, сплайнов, областей и т.д.

Продолжение – создание временной вспомогательной линии, являющейся продолжением объекта, над конечной точкой которого проходит курсор.

Вставки – привязка к точке вставки текста, атрибута, формы, определение атрибута или блока.

Ближайшая – привязка к точке на прямой, дуге или окружности, которая оказалась ближайшей к курсору.

Кажущееся пересечение – привязка к точке предполагаемого пересечения.

Нормаль – привязка к точке на прямой, окружности, эллипсе, сплайне или дуге, которая образует совместно с данной точкой нормаль к этому объекту.

Касательная – привязка к точке на окружности или дуге, которая при соединении с данной точкой образует касательную.

Все режимы объектной привязки могут быть использованы в любой комбинации из раскрывающегося списка (ПКМ на кнопке *Объектной привязки* в строке состояния).

Для редактирования чертежа можно использовать ручки, которые имеют вид квадратных меток, находятся в характерных точках выбранных объектов и появляются при выделении объекта или группы объектов, или команды редактирования, расположенные на соответствующей панели инструментов вкладки ленты *Главная*.

Задание 1

Используя режимы объектной привязки, выполнить эскизы, представленные на рисунке 13.1.

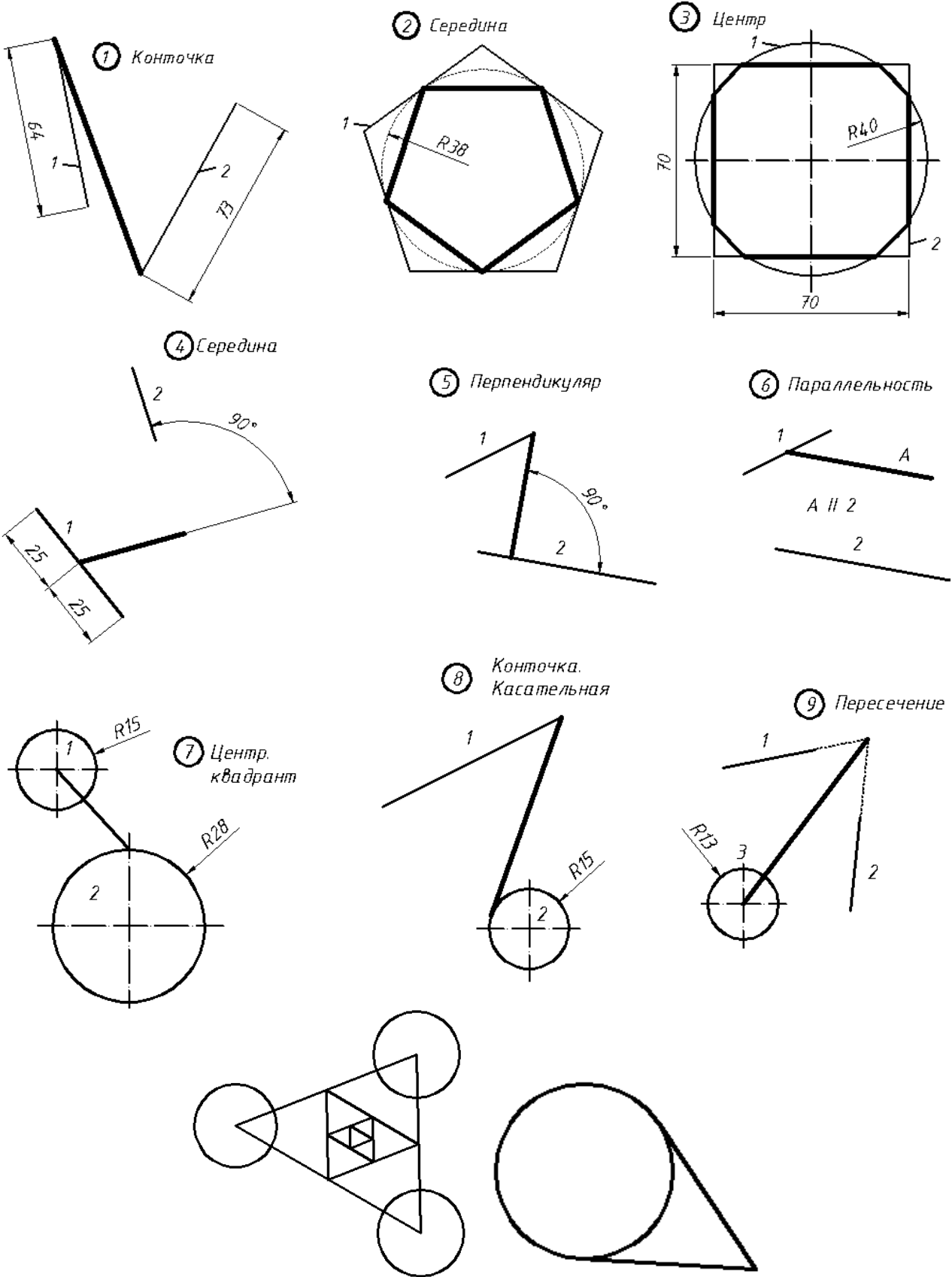


Рисунок 13.1 – К заданию 1

Задание 2 Изучить команды редактирования и выполнить эскизы, показанные на рисунках 13.2 и 13.3.



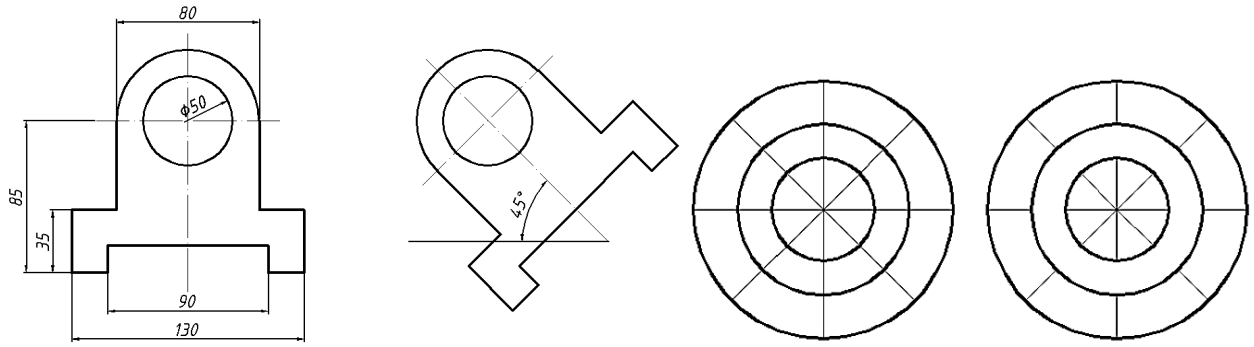


Рисунок 13.2 – К заданию 2

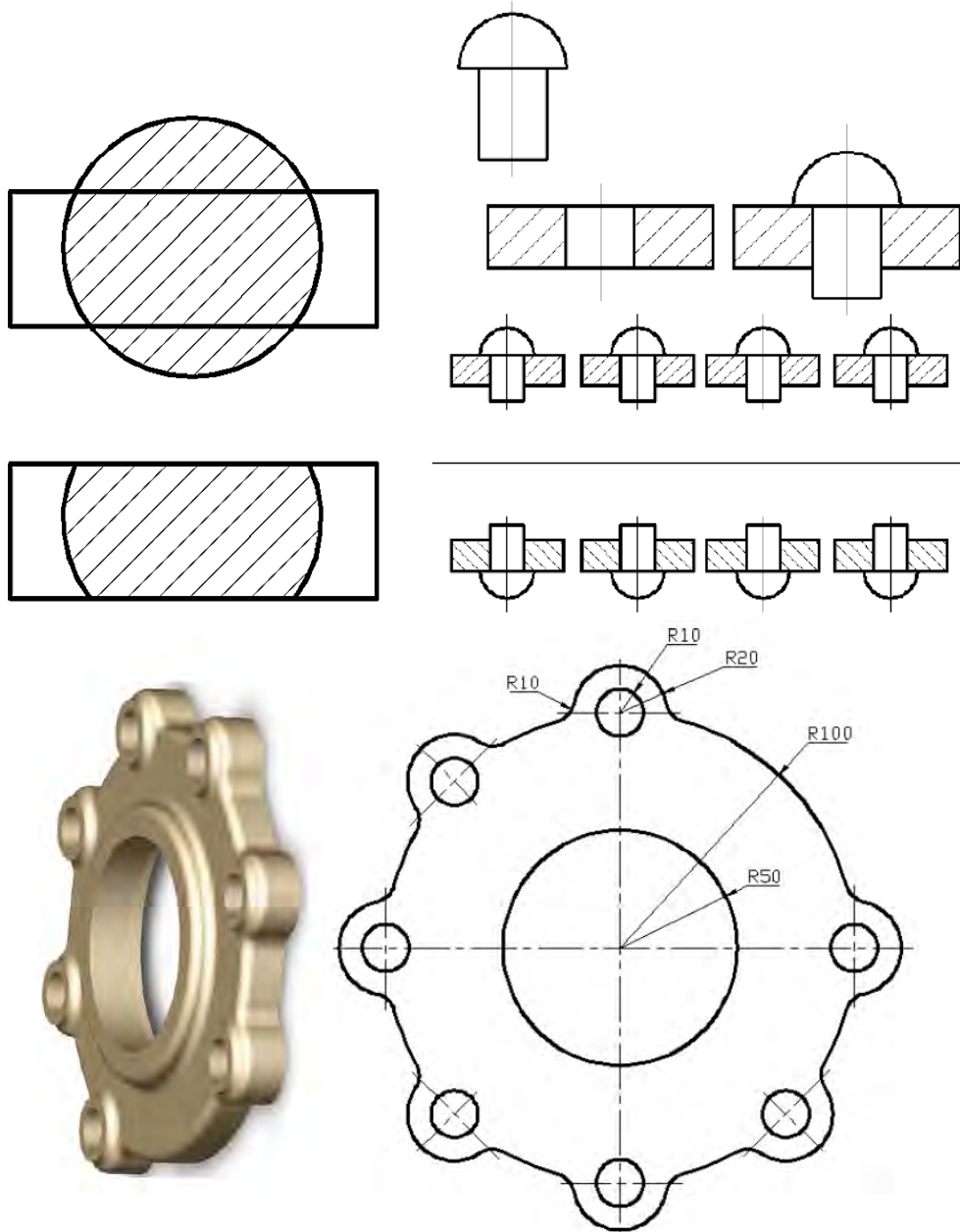


Рисунок 13.3 – К заданию 2

14 Лабораторная работа № 14. Графическая система AutoCAD

14.1 Нанесение размеров в AutoCad

Цель работы: получение навыков нанесения размеров в AutoCad.

Размер – это особая конструкция, которая содержит следующие составные элементы: размерную линию, стрелки, выносные линии, размерный текст и т. д. Несмотря на то, что размер состоит из многих элементов, в AutoCAD все составные части записываются в отдельный блок и рассматриваются компьютером, как один примитив.

Для нанесения размеров, допусков и выносных линий может быть использована вкладка ленты *Главная* группа *Аннотация* или вкладка ленты *Аннотации* группа *Размеры*.

Задание

Изучить инструменты для нанесения размеров и выполнить чертежи, представленные на рисунках 14.1–14.3.

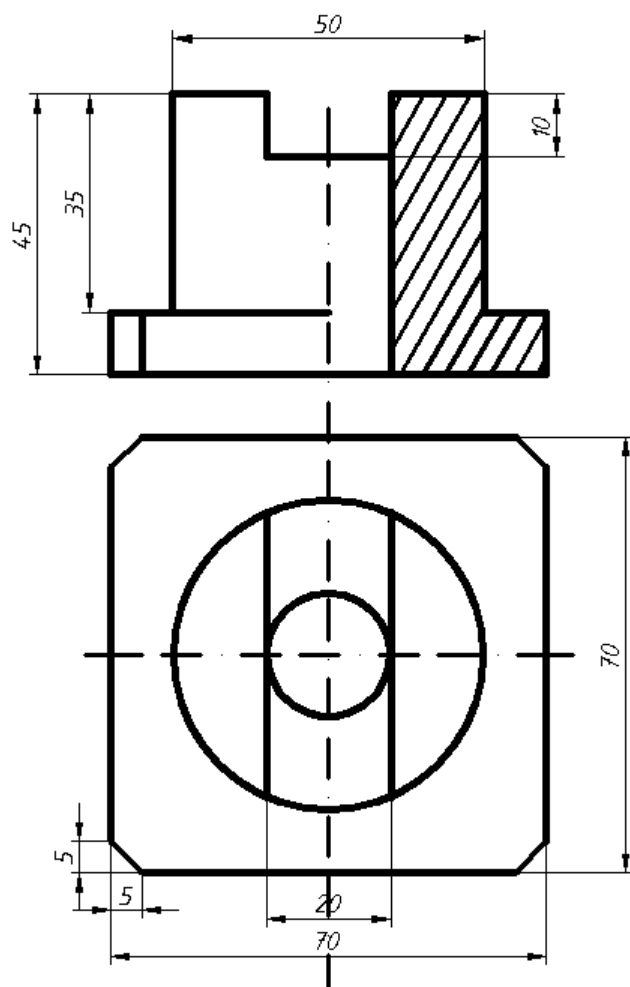


Рисунок 14.1 – К заданию

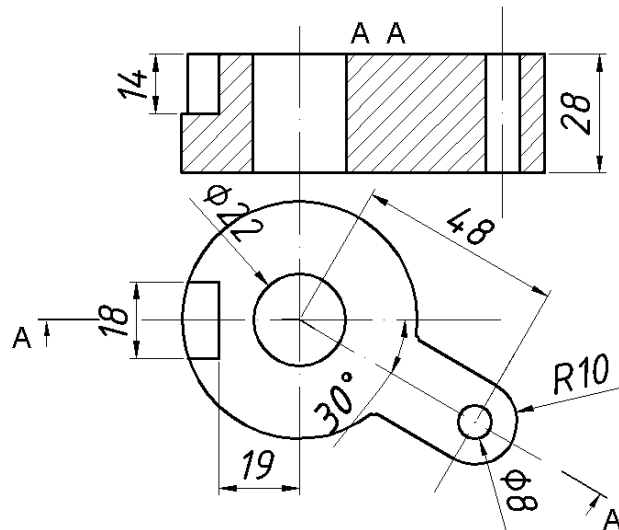


Рисунок 14.2 – К заданию

Отвод M45 (2 шт)

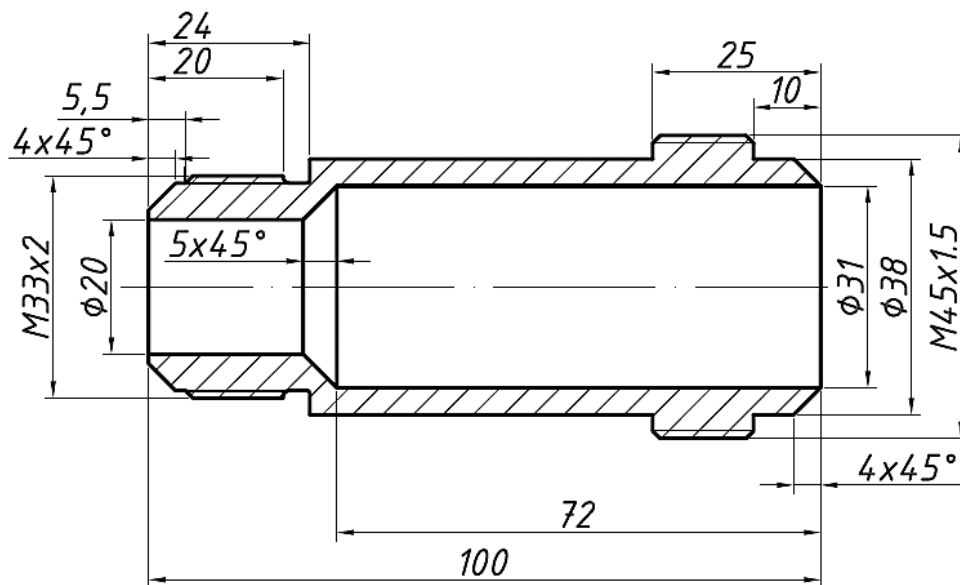


Рисунок 14.3 – К заданию

Контрольные вопросы

- 1 Что такое размер в AutoCAD?
- 2 Что такое ассоциативный размер?
- 3 Что такое неассоциативный размер?
- 4 Как нанести линейный размер в AutoCAD?
- 5 Как нанести угловой размер в AutoCAD?
- 6 Как нанести допуски в AutoCAD?

15 Лабораторная работа № 15. Графическая система AutoCAD


15.1 Трехмерное моделирование

Цель работы: получение навыков создания 3D-моделей в AutoCAD.

Задание 1

Выполнить действия из примеров использования команд 3D-моделирования.

Пример использования команды **ВЫДАВИТЬ**

1 Нажимаем кнопку , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*. Вычерчиваем объект (рисунок 15.1).

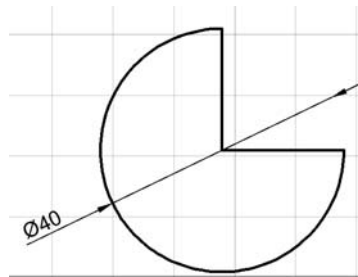
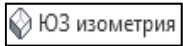



Рисунок 15.1 – Объект

2 Нажимаем команду **область** . Рамкой выделяем объект. Эскиз основания создан.

3 Нажимаем кнопку , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*.

4 Нажимаем команду . Выбираем созданный эскиз, мышью указываем направление выдавливания и вводим с клавиатуры расстояние, на которое необходимо выдавить элемент.

5 Нажимаем кнопку . Вычерчиваем объект (рисунок 15.2).

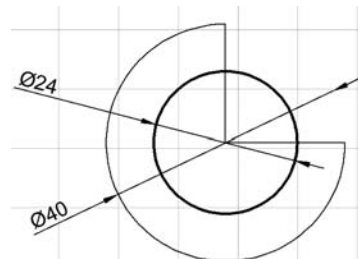
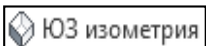



Рисунок 15.2 – Объект

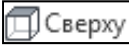
6 Нажимаем команду **область** . Рамкой выделяем объект.

7 Нажимаем кнопку , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*.

8 Нажимаем кнопку . Выбираем эскиз, мышью указываем направление выдавливания и вводим с клавиатуры расстояние.

9 Выбираем команду **ВЫЧИТАНИЕ**  и указываем сначала на внешнюю, затем на внутреннюю поверхности.

Пример использования команды СДВИГ

1 Нажимаем кнопку , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*. Вычерчиваем объект (рисунок 15.3).

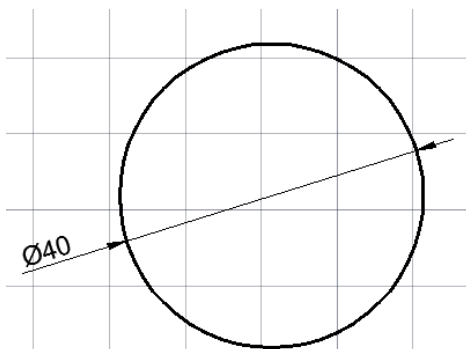

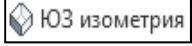



Рисунок 15.3 – Объект

2 Нажимаем команду **область** . Рамкой выделяем объект (рисунок 15.4) и правой клавишей мыши подтверждаем выделение.

3 Нажимаем кнопку , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*.

4 Нажимаем кнопку **СДВИГ** . Выделяем левой клавишей мыши круг, правой клавишей мыши подтверждаем выделение. Выделяем левой клавишей мыши прямую, являющуюся направляющей для сдвига. В итоге получаем поверхность.

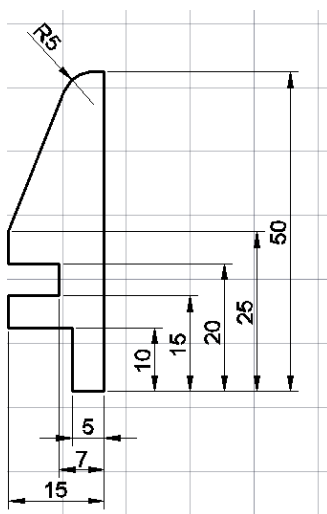


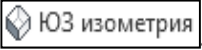



Рисунок 15.4 – Объект

Пример использования команды ВРАЩАТЬ

1 Нажимаем кнопку  , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*. Вычерчиваем объект (см. рисунок 15.4).

2 Нажимаем команду ОБЛАСТЬ  . Рамкой выделяем объект (см. рисунок 15.4) и подтверждаем выделение. Эскиз создан.

3 Нажимаем кнопку  , находящуюся на вкладке *Главная* → *Вид* → *Несохраненный вид*.

4 Нажимаем команду ВРАЩАТЬ  . Рамкой выбираем созданную область. Далее указываем направление оси вращения двумя точками, затем – направление выдавливания и вводим с клавиатуры угол, на который необходимо повернуть эскиз.

Задание 2

Построить тело, представленное на рисунке 15.5. Диаметр средней части колонны – 100 мм, радиус углублений – 4 мм, высота – 250 мм. Высота одной части основания – 25 мм.

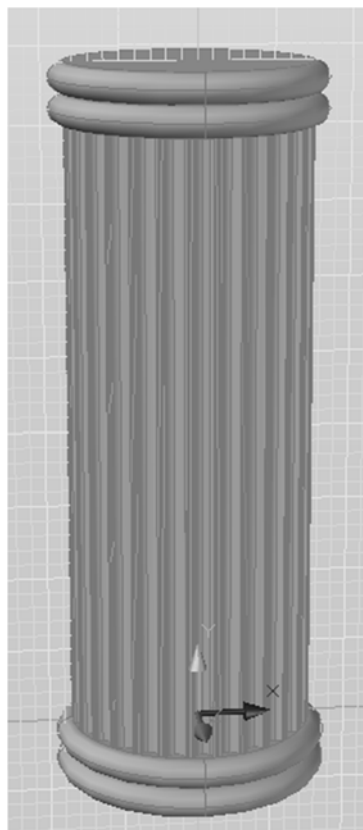


Рисунок 15.5 – К заданию 2

16 Лабораторная работа № 16. Программирование на алгоритмическом языке

16.1 Линейная программа. Организация ввода-вывода

Цель работы: изучение *Visual Basic for Application* на примере линейной программы; получение навыков организации ввода и вывода данных с помощью диалоговых окон и ячеек листа Excel.

Задание

Написать линейную программу для вычисления значения выражения по формулам из таблицы 16.1 в соответствии с вариантом, выданным преподавателем (предполагается, что значениями переменных могут быть любые действительные числа). Предусмотреть ввод и вывод данных двумя способами: с листа Excel и с помощью встроенных диалоговых окон.

Таблица 16.1 – Варианты заданий

Номер варианта	Выражение	Номер варианта	Выражение
1	$\frac{b + \sqrt{b^2 - 4ac}}{2a} - a^3c + b^{-1}$	8	$\frac{a}{b} \cdot \frac{c}{d} - \frac{ab - c}{cd}$
2	$\frac{\sin x + \cos y}{\cos x - \sin y} \cdot \operatorname{tg}(xy)$	9	$\frac{a+b}{c+1} - \frac{bc-1}{3+a}$
3	$\ln \left \left(y - \sqrt{ x } \right) \left(x - \frac{y}{x+x^2} \right) \right $	10	$a - \frac{b^3}{3} + \frac{c^5}{5}$
4	$\frac{\ln \cos x }{\ln(1+y^2)}$	11	$\left(\frac{a+1}{b-1} \right)^c + 8abc$
5	$\left(1 + \frac{1}{x^3} \right)^y - 12x^2y$	12	$\frac{a^2 - 7b + 10}{3b^2 - 8a + 11}$
6	$\cos x + 16x \cdot \cos(xy) - 2$	13	$2^{-a} - \cos b + \sin 2ab$
7	$3^x - 4x + \left(y - \sqrt{ x } \right)$	14	$\sin \sqrt{a+1} - \sin \sqrt{b-1}$

Пример выполнения задания

Написать линейную программу для вычисления значения выражения $\frac{\sin a}{\cos b}$.

1 Запускаем Excel и заходим в редактор VBA (ALT+F11).

2 Выполняем команду ВСТАВКА / МОДУЛЬ (меню Insert / Module). Появится **Окно кода**, где необходимо составить программу (макрос).

3 Вставляем процедуру: ВСТАВКА / ПРОЦЕДУРА (меню Insert / Procedure). Указываем имя (Name) процедуры – primer, тип (Type) – Sub. В окне кода появится следующий код:

```
Public Sub primer()
End Sub
```


4 Набираем программу:

```
Public Sub primer ()
Dim a As Double, b As Double
Dim c As Double
a = Worksheets(1).Range("a1").Value

b = Worksheets(1).Range("b1").Value

c = Sin(a) / Cos(b)

Worksheets(1).Range("c1").Value = c

End Sub
```

Заголовок процедуры
Раздел описания переменных

Присваивание переменной *a* числового значения ячейки A1 рабочего листа
Присваивание переменной *b* числового значения ячейки B1

Вычисление арифметического выражения

Вывод вычисленного значения в ячейку C1 рабочего листа 1

Конец процедуры

5 В ячейки A1 и B1 на листе Excel вводим числовые данные.

6 Запускаем программу на выполнение. Меню **RUN** → **Run Sub/UserForm**, или F5, или кнопка на панели инструментов. Просматриваем результаты выполнения программы на рабочем листе Excel в ячейке C1.

7 Вычисляем значение выражения = **Sin(A1)/Cos(B1)**, используя функции Excel. Сравниваем результаты.

8 Проверяем работу программы по шагам (меню **Debug / Step Into**). Для этого устанавливаем курсор возле переменной **a**, нажимаем правую кнопку мыши, вызвав контекстное меню, выбираем пункт **Add Watch / OK**. Внизу экрана появится окно **Watches**. Аналогично добавляем переменные **b** и **c**. Последовательно нажимаем клавишу F8 и наблюдаем за изменением значений переменных в окне **Watches**.

9 Оформляем отчет, в котором представляем код программы и алгоритм решения (рисунок 16.1).



Рисунок 16.1 – Алгоритм программы



Контрольные вопросы

- 1 Перечислите способы ввода данных в VBA.
- 2 Перечислите способы вывода данных в VBA.
- 3 Как описать переменные в VBA?
- 4 Перечислите типы данных в VBA.

17 Лабораторная работа № 17. Программирование на алгоритмическом языке

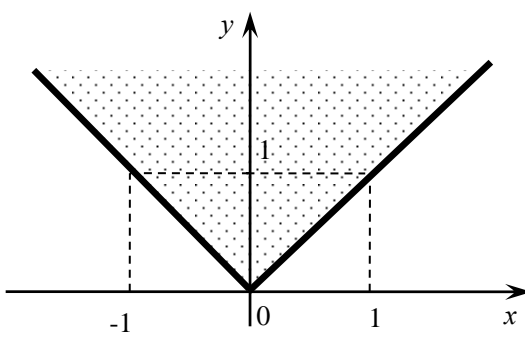
17.1 Условный оператор IF

Цель работы: изучение условного оператора IF.

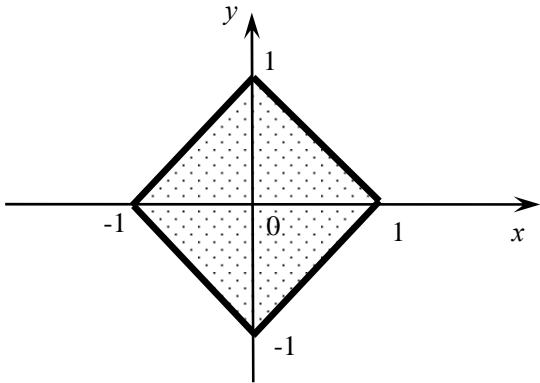
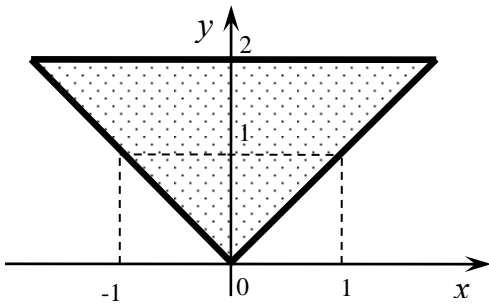
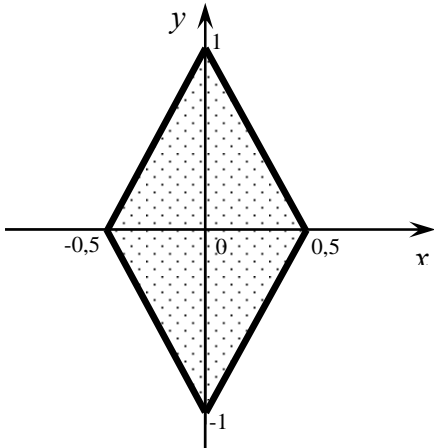
Задание

Написать программы с использованием условного оператора в соответствии с вариантом из таблицы 17.1.

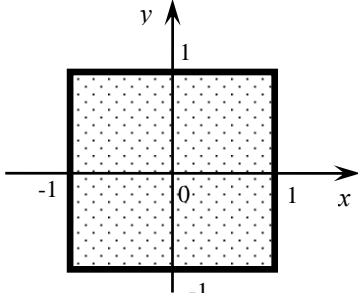
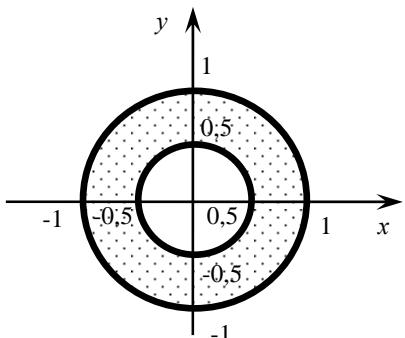
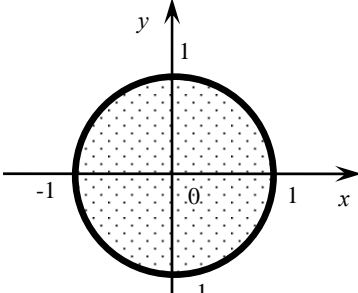
Таблица 17.1 – Варианты заданий

Номер варианта	Номер задачи	Условие
1	1	Вычислить $a = \begin{cases} (x+y)^2 - \sqrt{x \cdot y}, & \text{если } x \cdot y > 0; \\ (x+y)^2 - 25, & \text{если } x \cdot y < 0; \\ (x+y)^2 + 1, & \text{если } x \cdot y = 0 \end{cases}$
	2	Ввести три целых числа, используя InputBox, и определить, сумма каких двух является наибольшей
2	1	Вычислить $b = \begin{cases} \ln(x/y) + (x^2 + y)^3, & \text{если } x \cdot y > 0; \\ \ln(y/x) + (x^2 - y)^3, & \text{если } x \cdot y < 0; \\ x + y, & \text{если } x = 0 \end{cases}$
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 

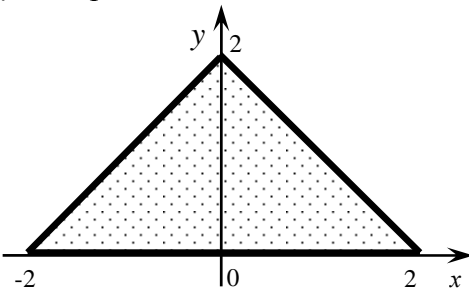
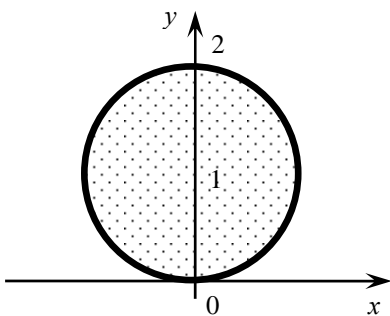
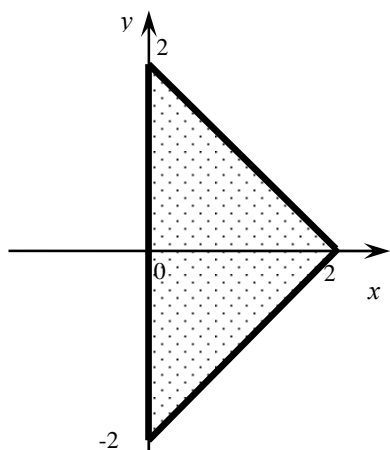
Продолжение таблицы 17.1

Номер варианта	Номер задачи	Условие
3	1	Ввести число X и определить, делится ли оно на 8 и на 6. Результаты проверок вывести. Числа вводить, используя InputBox
	2	<p>Даны действительные числа x, y. Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости</p> 
4	1	Ввести число X и определить, является ли оно четным
	2	<p>Даны действительные числа x, y. Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости</p> 
5	1	Ввести число N и определить, делится ли оно без остатка на число M . Числа вводить, используя InputBox
	2	<p>Даны действительные числа x, y. Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости</p> 

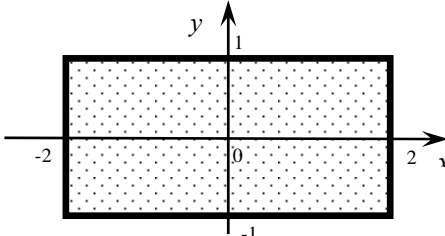
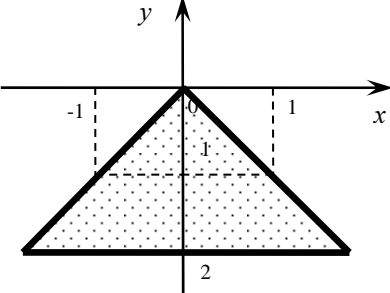
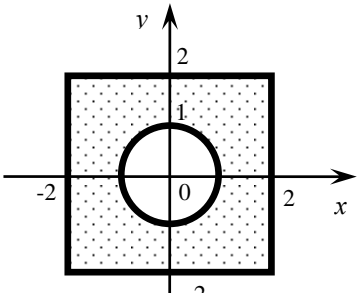
Продолжение таблицы 17.1

Номер варианта	Номер задачи	Условие
6	1	Ввести три целых числа и определить, произведение каких двух является max. Числа вводить, используя InputBox
	2	<p>Даны действительные числа x, y. Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости</p> 
7	1	<p>Вычислить $x = \begin{cases} i \cdot a^3, & \text{если } i - \text{нечетное, } a < 0; \\ 2 \cdot i \cdot \sqrt{a}, & \text{если } i - \text{четное, } a > 0; \\ i - a, & \text{иначе} \end{cases}$</p>
	2	<p>Даны действительные числа x, y. Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости</p> 
8	1	<p>Вычислить $x = \begin{cases} i^2 \cdot \sqrt{a}, & \text{если } i - \text{нечетное, } a > 0; \\ i / a , & \text{если } i - \text{четное, } a < 0; \\ i \cdot a, & \text{иначе} \end{cases}$</p>
	2	<p>Даны действительные числа x, y. Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости</p> 

Продолжение таблицы 17.1

Номер варианта	Номер задачи	Условие
9	1	Вычислить $s = \begin{cases} x^2 - 3x + 7, & \text{если } x > 0; \\ \sin^2(x + 3), & \text{если } x < 0; \\ 88, & \text{если } x = 0 \end{cases}$
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 
10	1	Составить программу, которая проверяет, являются ли введенные три числа углами треугольника
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 
11	1	Составить программу для нахождения минимального из трех чисел
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 

Продолжение таблицы 17.1

Номер варианта	Номер задачи	Условие
12	1	Вычислить $f(x) = \begin{cases} \sin \sqrt{x}, & \text{если } x > 0; \\ \cos^2 x - 4, & \text{если } x < 0; \\ 0, & \text{если } x = 0 \end{cases}$
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 
13	1	Составить программу для нахождения максимального из трех чисел
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 
14	1	Вычислить $y = \begin{cases} i \cdot \sqrt{a^3}, & \text{если } i - \text{нечетное, } a < 0; \\ i \cdot \sqrt{a}, & \text{если } i - \text{четное, } a > 0; \\ i - 2^a, & \text{иначе} \end{cases}$
	2	Даны действительные числа x, y . Определить, принадлежит ли точка с координатами x, y заштрихованной части плоскости 

Пример выполнения задания

Ввести x и вычислить значение функции F по формуле

$$F = \begin{cases} x/2, & \text{если } x > 0; \\ (x+1)/2, & \text{если } x < 0. \end{cases}$$

- 1 В ячейку A1 ввести текст “Исходные данные”.
- 2 В ячейку A2 ввести текст “X=”.
- 3 В ячейку B2 ввести числовое значение X.
- 4 В ячейку C1 ввести текст “результат при $x > 0$ ”.
- 5 В ячейку D1 ввести текст “результат при $x < 0$ ”.
- 6 Запустить редактор VBA (ALT+F11).
- 7 Выполнить команду ВСТАВКА/МОДУЛЬ. Ввести имя модуля SUB LL().
- 8 Набрать текст программы:

```
Sub LL ()
Dim x As Double
X= Worksheets(1).Range("B2").Value

IF X>0 Then
F=X/2
Worksheets(1).Range("C2").Value=F
ELSE
F=(X+1)/2
Worksheets(1).Range("D2").Value=F
End If
End Sub
```

Присвоение переменной X числового значения ячейки B2

Блок условного оператора IF

Конец процедуры

- 9 Оформить отчет: представить код программы и алгоритм решения (рисунок 17.1).

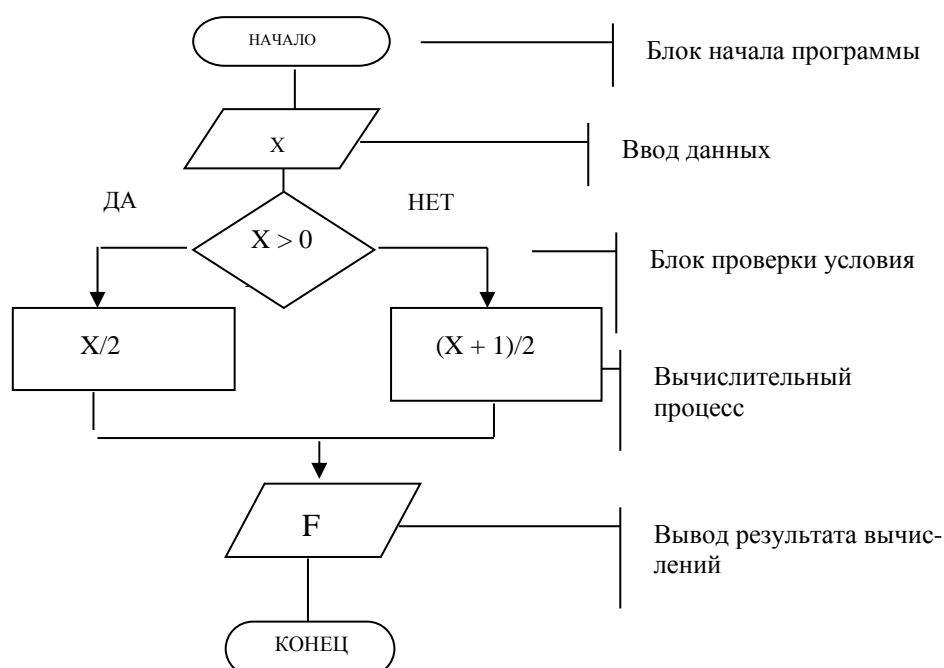


Рисунок 17.1 – Алгоритм программы



Контрольные вопросы

- 1 Назначение условного оператора IF.
- 2 Как работает условный оператор IF?
- 3 Какие формы записи условного оператора вы использовали в своих программах?

18 Лабораторная работа № 18. Программирование на алгоритмическом языке

18.1 Изучение оператора цикла с параметрами For-Next

Цель работы: изучение оператора цикла с параметрами For-Next.

Задание

Написать программу с использованием оператора цикла в соответствии с вариантом, выданным преподавателем из таблицы 18.1.

Таблица 18.1 – Варианты заданий

Номер варианта	Задание
1	Найти количество положительных чисел в последовательности из n чисел (последовательность вводится пользователем)
2	Найти сумму положительных чисел в последовательности из n чисел, вводимой с клавиатуры
3	Найти количество отрицательных чисел в последовательности из n чисел (последовательность вводится пользователем)
4	Вводится n чисел. Найти их среднее арифметическое
5	Найти произведение положительных чисел последовательности из n чисел
6	Найти факториал числа (факториалом числа n называют произведение целых чисел от 1 до n : $n! = 1*2*...n$)
7	Ввести n целых чисел и определить, сколько из них попадают в диапазон $-m < \text{число} < m$, n вводится и должно быть больше 0
8	Ввести n вещественных чисел и определить, что больше: их сумма или произведение, n вводится и должно быть больше 0
9	Возвести число m в целую степень n , не используя операции возведения в степень
10	Вывести таблицу соответствий между весом в фунтах и килограммах для значений от n до m фунтов, n и m вводятся. 1 фунт = 400 г
11	Вычислить квадраты натуральных чисел от n до m с шагом 2 (через одно число), n и m вводятся. Введенные данные необходимо проверить на допустимость ($n < m$)
12	Вывести таблицу перевода температуры из градусов Цельсия в градусы по шкале Фаренгейта для температур от 0 до 100 °C с заданным шагом h (шаг вводится). Фаренгейт = $1,8 \text{ } ^\circ\text{C} + 32$
13	Составить таблицу степеней для числа a от степени 0 до n
14	Найти площади n квадратов. Сторона каждого квадрата увеличивается на h

Пример выполнения задания

Вычислить значение функции $Y = \sum_{i=1}^n \ln\left(\frac{X}{2^i}\right)$, где $n = 1, 2, \dots, 5$.

- 1 Ввести в ячейку A1 значение переменной X.
- 2 Запустить редактор VBA (ALT+F11).
- 3 Выполнить команду ВСТАВКА/МОДУЛЬ. Ввести имя модуля SUB mm().
- 4 Набрать текст программы:

```
Sub mm ()
Dim X As single
Dim Y As single
  n=5 : Y=0
  X=Worksheets(1).Cells(A1)
  For i=1 to n
    Y=Y+log(x)/2^i
  Next i
  Worksheets(1).Range ("A6").Value = "результат"
  Worksheets(1).Range ("A7").Value = Y
End Sub
```

- 5 Оформить отчет: представить код программы и алгоритм решения (рисунок 18.1).

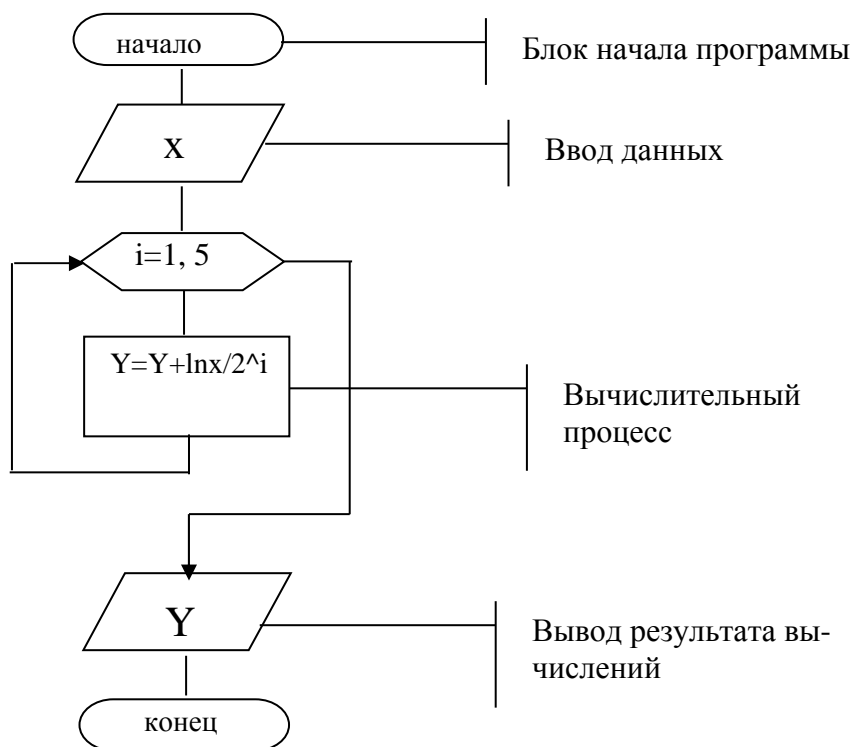


Рисунок 18.1 – Блок-схема

Контрольные вопросы

- 1 Назначение оператора цикла с параметрами For-Next.
- 2 Как работает оператор цикла с параметрами For-Next?
- 3 Какое количество раз будет пройден цикл в вашей работе?



19 Лабораторная работа № 19. Программирование на алгоритмическом языке

19.1 Изучение способов обработки массивов

Цель работы: изучение способов обработки массивов.

Задание 1

Написать алгоритм и программу обработки одномерного массива согласно варианту из таблицы 19.1.

Таблица 19.1 – Варианты заданий

Номер варианта	Задание
1	Создать одномерный массив из N целочисленных значений. Найти наибольший среди положительных элементов массива и вывести его на экран
2	Создать одномерный целочисленный массив из N значений. Подсчитать среднее арифметическое значение всех элементов и вывести его на экран
3	Создать одномерный целочисленный массив из N значений. Расставить в нем элементы в обратном порядке
4	Создать одномерный целочисленный массив из N значений. Поменять местами каждую пару чисел, например, A_1 с A_2 , A_3 с A_4 и т. д
5	Создать одномерный массив из N целочисленных значений. Вывести все отрицательные элементы массива и подсчитать их количество
6	Создать одномерный массив из N целочисленных значений. Вывести все положительные элементы массива и подсчитать их количество
7	Создать два одномерных массива по 10 целочисленных значений каждый. Сформировать третий массив из 20 элементов следующим образом: на нечетные места ставить элементы из первого массива, на четные – из второго
8	Создать два одномерных массива по 10 целочисленных значений каждый. Сформировать третий массив из 20 элементов следующим образом: сначала ставить элементы из первого массива, затем – из второго
9	Создавать одномерный массив из 20 целочисленных значений. Вывести элементы, стоящие на нечетных местах
10	Создать два одномерных массива по 20 целочисленных значений. Совершить обмен данными: в одном массиве – с 1-го по 10-й элемент, в другом – с 11-го по 20-й элемент, т. е. X_1 на Y_{11} , Y_{11} на X_1 и т. д
11	Создать одномерный массив из N целочисленных значений. Расставить в нем в том же порядке сначала положительные элементы, затем – отрицательные, далее нулевые
12	Создать одномерный массив из N целочисленных значений. Найти минимальное среди них, вывести его и индекс данного элемента на экран.
13	Создать одномерный массив из N целочисленных значений. Найти минимальное и максимальное значения, вывести их разность на экран
14	Создать одномерный массив из N целочисленных значений. Найти наибольшее среди отрицательных и вывести его на экран

Задание 2

Написать алгоритм и программу для обработки двумерного массива согласно варианту из таблицы 19.2.

Таблица 19.2 – Варианты заданий

Номер варианта	Задание
1	Дана целочисленная прямоугольная матрица. Определить количество строк, содержащих хотя бы один нулевой элемент
2	Дан массив размерностью $n \times m$. Заменить все положительные элементы на 1, все отрицательные на -1
3	Дан массив размерностью $n \times m$. Подсчитать сумму элементов, у которых сумма номера строки и номера столбца $n = 1$
4	Дан массив размерностью $n \times m$. Подсчитать сумму элементов, у которых модуль разности номера строки столбца равен 1
5	Дан массив размерностью $n \times m$. Подсчитать сумму элементов, которые меньше номера своей строки
6	Дан массив размерностью $n \times m$. Подсчитать сумму элементов, которые больше суммы номера своего столбца и строки
7	Дан массив размерностью $n \times m$. Подсчитать сумму положительных элементов и распечатать их номера
8	Дан массив размерностью $n \times m$. Подсчитать сумму нечетных элементов и распечатать их номера
9	Дан массив размерностью $n \times m$. Подсчитать сумму элементов, имеющих одинаковые остатки при делении на 7 и на 2, и распечатать номера таких элементов
10	Дан массив размерностью $n \times m$. Преобразовать элементы массива по следующему правилу: если элемент четный, то разделить его на 2, если нечетный – заменить его остатком от деления на 3
11	Дан массив размерностью $n \times m$. Преобразовать элементы массива по следующему правилу: если элемент положительный, то умножить его на 2, а если отрицательный – поменять знак на противоположный
12	Дан массив размерностью $n \times m$. Преобразовать элементы массива по следующему правилу: если элемент четный, то прибавить к нему 1, если нечетный – умножить на 2
13	Дан массив размерностью $n \times m$. Преобразовать элементы массива по следующему правилу: если элемент положительный, то прибавить к нему 10, если отрицательный – умножить на 10
14	Дан массив размерностью $n \times m$. Подсчитать сумму элементов, имеющих одинаковые остатки при делении на 3 и на 5, и распечатать номера таких элементов

Контрольные вопросы

- 1 Что такое массив?
- 2 Классификация массивов.
- 3 Способы описания массивов.
- 4 Способы ввода динамического массива.

20 Лабораторная работа № 20. Программирование на алгоритмическом языке

20.1 Применение пользовательских подпрограмм

Цель работы: изучение возможности создания и использования пользовательских подпрограмм.

Задание

Решить задачу по вариантам из таблицы 20.1. Задание выполнить двумя способами: используя подпрограмму-процедуру и подпрограмму-функцию.

Таблица 20.1 – Варианты заданий

Номер варианта	Задание
1	Рассчитать значение $x = \frac{\sqrt{6+6}}{2} + \frac{\sqrt{13+13}}{2} + \frac{\sqrt{21+21}}{2}$
2	Даны два натуральных числа. Выяснить, в каком из них сумма цифр больше
3	Рассчитать значение $x = \frac{\sqrt{6+17}}{6+\sqrt{17}} + \frac{\sqrt{17+13}}{17+\sqrt{13}} + \frac{\sqrt{21+32}}{21+\sqrt{32}}$
4	Даны основания и высоты двух равнобедренных трапеций. Найти сумму их периметров и сумму их площадей
5	Найти значение выражения $x = \frac{2 \cdot 5! + 6 \cdot 8!}{6! + 4!}$, где $n!$ означает факториал числа n
6	Рассчитать значение $x = \frac{\sqrt{6+1}}{1+\sqrt{8}} + \frac{\sqrt{10+1}}{1+\sqrt{12}} + \frac{\sqrt{14+1}}{1+\sqrt{16}}$
7	Даны стороны двух треугольников. Найти сумму их периметров и сумму их площадей
8	Написать программу вычисления суммы факториалов всех нечетных чисел от 1 до 9
9	Даны три квадратных уравнения $ax^2 + bx + c = 0$, $bx^2 + ax + c = 0$, $cx^2 + ax + b = 0$. Определить, сколько из них имеют вещественные корни
10	Даны шесть различных чисел. Определить минимальное из них (написать подпрограмму, находящую минимум из двух чисел)
11	Дано n окружностей с радиусами r_1, r_2, \dots, r_n соответственно. Найти сторону прямоугольника, площадь которого равна сумме n площадей окружностей, если одна сторона прямоугольника равна a . Вычисление площади круга оформить в виде подпрограммы
12	Найти значение выражения $x = \frac{7! \cdot 8! + 9! \cdot 10!}{11! + 12!}$, где $n!$ означает факториал числа n
13	Написать программу вычисления суммы факториалов всех четных чисел от 2 до 10
14	Даны шесть различных чисел. Определить максимальное из них (написать подпрограмму, находящую максимум из двух различных чисел)

Пример выполнения задания

Вычислить сумму членов ряда $\sum_{i=1}^n \frac{1}{i!}$, где $i!$ – факториал числа i (произведение натуральных чисел от 1 до i).

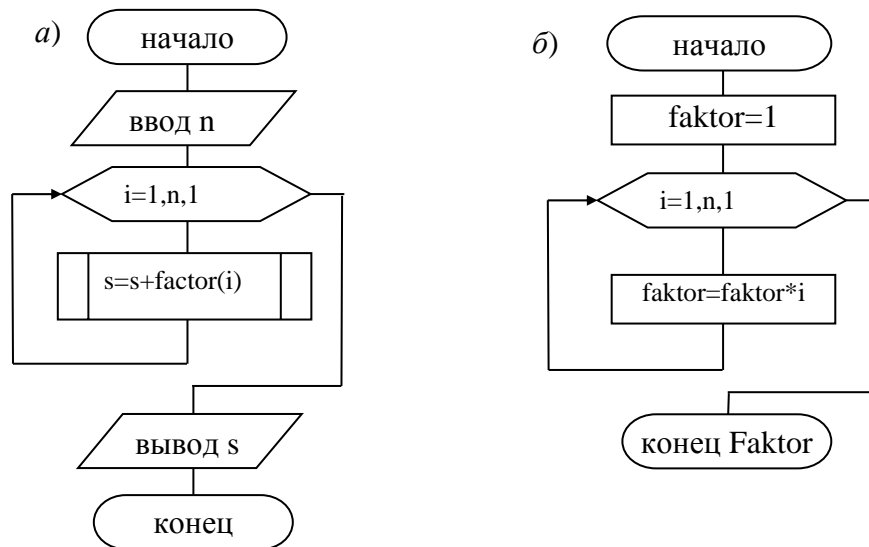
Программа с описанием представлена далее, а блок-схема основной программы и подпрограммы-функции – на рисунке 20.1.

```
Public Sub prog()
Dim i As Integer, n As Integer
Dim s As Double
n = CInt(InputBox("Введите n"))
For i = 1 To n
s = s + 1 / faktor(i)
Next
MsgBox s
End Sub

Public Function faktor(x As Integer)
As Long
faktor = 1
For i = 1 To x
faktor = faktor * i
Next i
End Function
```

Начало основной программы
Описание переменных основной программы
Ввод конечного значения ряда
Задание цикла для суммирования
При вычислении искомой суммы производится вызов функции faktor и передается ее аргумент i

Начало подпрограммы функции
При описании функции типу ее результата присваивается тип длинный целый (Long), т. к., например, $10! = 40320$, что выходит за диапазон типа Integer. При выполнении функции результат присваивается ее имени



а – основная программа; б – подпрограмма-функция

Рисунок 20.1 – Блок-схема программы

Контрольные вопросы

- 1 Что такое подпрограмма-процедура?
- 2 Что такое подпрограмма-функция?

21 Лабораторная работа № 21. Программирование на алгоритмическом языке

21.1 Линейная программа с использованием пользовательской формы. Отладка программ и обработка ошибок

Цель работы: изучение возможности создания и использования пользовательских форм; изучение процедуры обработки ошибок и использование их при решении задач.

Задание

Создать пользовательскую форму для вычисления по вариантам из таблицы 21.1, предусмотрев на ней функцию очистки окон и обработки ошибок.

Таблица 21.1 – Варианты заданий

Номер варианта	Задание
1	Вычислить $\gamma = x - y - (1 + 2x)^y + e^{\sqrt{ x-y }}$
2	Вычислить $f(x, y, z) = \frac{x + y + z}{x^2 + y^2 + z^2}$
3	Вычислить площадь прямоугольного треугольника по двум катетам
4	Вычислить $f(x, y) = \frac{x}{1+y} + \frac{y}{1+x} + \frac{y}{y+x}$
5	Вычислить площадь куба по его стороне
6	Вычислить $f(x, y, z) = \frac{x + y + z}{x \cdot y \cdot z}$
7	Вычислить длину гипотенузы по заданным длинам катетов
8	Вычислить дискриминант квадратного уравнения по заданным коэффициентам
9	Вычислить $f(x, y) = (x + y)(x^2 + y^2)(x^3 + y^3)$
10	Вычислить объем шара по заданному радиусу
11	Вычислить объем цилиндра по заданным радиусу и высоте
12	Вычислить $f(x, y, z) = \frac{x \cdot y \cdot z}{x + y^2 + z^3}$
13	Вычислить $f(x, y, z) = \frac{x + 2y + 3z}{x^3 + y^2 + z}$
14	Вычислить произведение четырех вещественных чисел

Пример выполнения задания

Найти сумму $a + b = c$.

- 1 Запустить редактор Visual Basic.
- 2 Выполнить команду *Вставка (Insert) → UserForm*.
- 3 Поместить на форму элементы, требуемые для решения задачи, с панели элементов и расположить их нужным образом, как показано на рисунке 21.1.



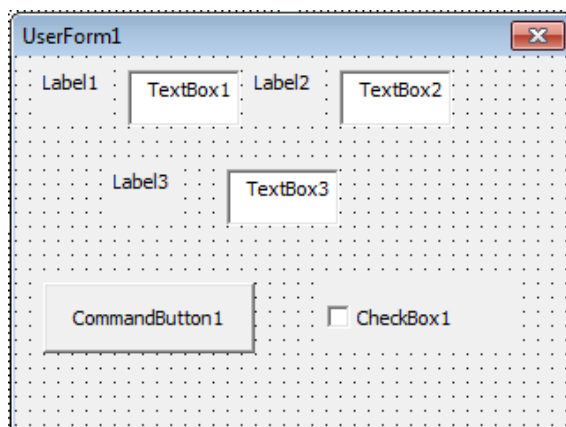


Рисунок 21.1 – Вид пользовательской формы

4 Изменить свойства объектов (таблица 21.1) на форме с помощью окна свойств. Вид пользовательской формы после назначения свойств представлен на рисунке 21.2.

Таблица 21.1 – Свойства объектов

Свойство	Значение
Label1.Caption	<i>a</i>
Label2.Caption	<i>b</i>
Label3.Caption	<i>c</i>
CommandButton1.Caption	Результат
CheckBox1.Caption	Очистка окон
Свойство .BackColor	По своему вкусу выбрать цвет из палитры цветов
Для Label1, Label2, Label3 свойство .Font	В ДО «Шрифт», которое появится после щелчка по кнопке, расположенной напротив свойства Font, выбрать размер 16

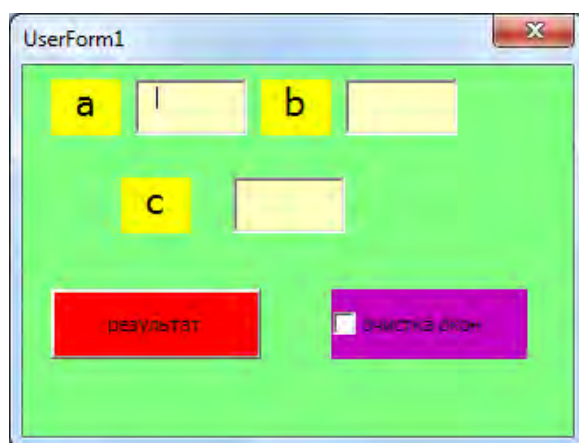


Рисунок 21.2 – Вид пользовательской формы после назначения свойств

5 Написать программный код. Для этого рекомендуется выполнить двойной щелчок по кнопке *результат* и перейти в окно программы, где набрать текст процедуры обработки события Click() для кнопки CommandButton1:

```

Private Sub CommandButton1_Click()
Dim a As Integer
Dim b As Integer
Dim c As Integer
a = CInt(TextBox1.Text)
b = CInt(TextBox2.Text)
c = a + b
MsgBox "результат смотри в TextBox3"
TextBox3.Visible = True
TextBox3.Text = c
End Sub

```

6 Двойной щелчок по элементу управления CheckBox1 вызовет процедуру обработки события Click(), где необходимо написать программный код для очистки полей TextBox:

```

Private Sub CheckBox1_Click()
TextBox1.Text = ""
TextBox2.Text = ""
TextBox3.Text = ""
TextBox3.Visible = False
TextBox1.SetFocus
CheckBox1.Value = False
End Sub

```

Рассмотрим процесс создания приложения, в котором предотвращается появление ошибки на конкретном примере, в котором программа производит деление числителя на знаменатель по нажатию кнопки «Ответ» без контроля появления возможных ошибок. Пользовательская форма представлена на рисунке 21.3, а текст программы приведен далее.

```

Private Sub CommandButton1_Click()
Dim Числитель As Single, Знаменатель As Single, Результат As Single
Числитель = CDb1(TextBox1.Text)
Знаменатель = CDb1(TextBox2.Text)
Результат = Числитель / Знаменатель
TextBox3.Text = CStr(Результат)
End Sub

```

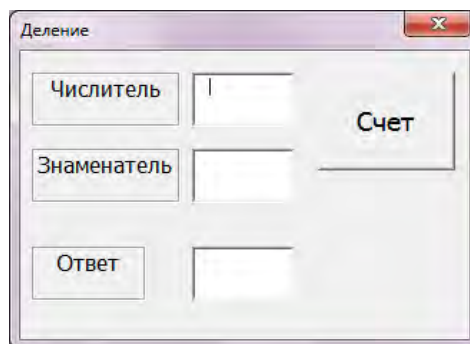


Рисунок 21.3 – Пользовательская форма



Полный текст программы с обработкой ошибок:

```
Private Sub CommandButton1_Click()
Dim Числитель As Single, Знаменатель As Single, Результат As Single
If IsNumeric(TextBox1.Text) = False Then
MsgBox "Ошибка в числителе", 32
TextBox1.SetFocus
Exit Sub
End If
If IsNumeric(TextBox2.Text) = False Then
MsgBox "Ошибка в знаменателе", 32
TextBox2.SetFocus
Exit Sub
End If
If CSng(TextBox2.Text) = 0 Then
MsgBox "Знаменатель не может быть нулем", 32
TextBox2.SetFocus
Exit Sub
End If
Числитель = CDb1(TextBox1.Text)
Знаменатель = CDb1(TextBox2.Text)
Результат = Числитель / Знаменатель
TextBox3.Text = CStr(Результат)
End Sub
```

Контрольные вопросы

- 1 Перечислите элементы управления в VBA.
- 2 Перечислите общие свойства элементов управления.
- 3 Перечислите свойства элемента управления TextBox.
- 4 Перечислите свойства элемента управления Label.



22 Лабораторная работа № 22. Программирование на алгоритмическом языке

22.1 Элемент управления Image (рисунок)

Цель работы: изучение основных свойств и методов элемента управления Image; построение графика на листе Excel и на пользовательской форме.

В Excel различаются два вида диаграмм: внедренная на рабочий лист (объект ChartObject) и созданная на специальном листе для диаграмм (объект Charts).

Задание

Построить график функции на заданном диапазоне с заданным шагом на листе Excel и на пользовательской форме согласно варианту из таблицы 22.1. Предусмотреть обработку ошибок.

Таблица 22.1 – Варианты исходных данных

Номер варианта	Функция	Отрезок $[a; b]$	Шаг h , град
1	$y = 2 \sin(x) \cos(x)$	$[0^\circ; 360^\circ]$	20
2	$y = 2 \sin^2(x) + \cos(x)$	$[0^\circ; 200^\circ]$	10
3	$y = \sqrt[3]{2 \sin(x) \cos(x)}$	$[-100^\circ; 100^\circ]$	20
4	$y = \sin(2x) \cos^3(x)$	$[30^\circ; 360^\circ]$	15
5	$y = \sqrt[3]{2 \sin(x) \cos(x^2)}$	$[-45^\circ; 90^\circ]$	15
6	$y = 2 \operatorname{tg}(x+1)$	$[0^\circ; 180^\circ]$	9
7	$y = -\sin^3(x) \sqrt{\cos(x)}$	$[-30^\circ; 120^\circ]$	15
8	$y = 2 \operatorname{tg}(\sqrt{x}) \cos(3x)$	$[0^\circ; 360^\circ]$	20
9	$y = x \sin(x)$	$[0^\circ; 200^\circ]$	10
10	$y = -x \cos(x)$	$[-100^\circ; 100^\circ]$	20
11	$y = \sin^2(x) + x$	$[30^\circ; 360^\circ]$	15
12	$y = \sqrt[3]{2 \sin(x) \cos(x)}$	$[-45^\circ; 90^\circ]$	15
13	$y = \operatorname{tg}^3(x) + \sqrt{\cos(x)}$	$[0^\circ; 180^\circ]$	9
14	$y = \sqrt{\sin^3(x) \cos(x^2)}$	$[-30^\circ; 120^\circ]$	15

Пример выполнения задания

Построить на листе Excel график функции $y = \frac{\sin x}{x^2 + 1}$ на отрезке $[a; b]$ с шагом h . Построение графика на листе Excel производится через процедуру.

```
Public Sub primergrafik()
Dim a As Double, b As Double, h As Double
Worksheets(1).Range("A:B").Select

Selection.Clear
Worksheets(1).ChartObject.Delete
a = Cdbl(InputBox("Введите значение a"))
b = Cdbl(InputBox("Введите значение b"))
h = Cdbl(InputBox("Введите значение h"))
j = 1
For i = a To b Step h
Worksheets(1).Range("A" & j) = i
Worksheets(1).Range("B"&j)=Sin((i*3.14)/
180)/(i^2+1)
j = j + 1
Next i
```

Выделение двух столбцов
первого листа
Очистка листа
Удаление диаграммы

Ввод данных

В цикле выводятся данные
на лист Excel, в столбец
A – значения аргумента, B –
значения функции



```
график
End Sub
```

```
Public Sub график()
n=Application.CountA(Worksheets(1).Range("A:A"))
)
```

```
Range("A1:B" & CStr(n)).Select
```

```
Charts.Add
```

```
ActiveChart.ChartType=xlXYScatterLinesNoMarkers
```

```
ActiveChart.SetSourceDataSource := Sheets
("Лист1 ").Range("A1:B" & CStr(n))
```

```
ActiveChart.LocationWhere:=xlLocationAsObject,
Name:="Лист1"
```

```
End Sub
```

Вызов процедуры график

Присваивание переменной n количества заполненных строк на листе в столбце A
Выбор данных для построения графика
Добавление листа диаграмм в рабочую книгу
Выбор типа диаграммы

Установка источника данных диаграммы

Для построения графика на пользовательской форме создадим форму, на которой предусмотрим поля для ввода данных и область для вывода графика, а также кнопки для построения графика, очистки формы и выхода (рисунок 22.1).

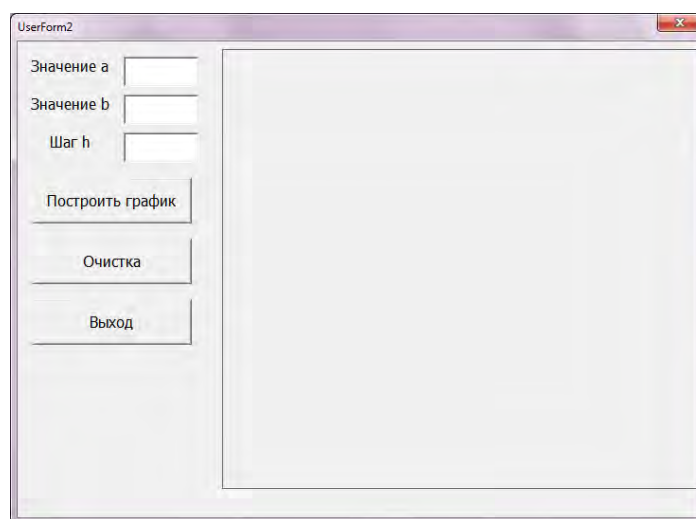


Рисунок 22.1 – Пользовательская форма

Текст программы представлен далее.

```
Dim g As String
Private Sub CommandButton1_Click()
Dim a As Double, b As Double, h As Double
Worksheets(1).Range("A:B").Select
Selection.Clear
'Worksheets(1).ChartObject.Delete
a = Cdbl(TextBox1.Text)
b = Cdbl(TextBox2.Text)
h = Cdbl(TextBox3.Text)
```

```

j = 1
For i = a To b Step h
Worksheets(1).Range("A" & j) = i
Worksheets(1).Range("B" & j)=Sin((i*3.14)/180)/(i^2+1)
j = j + 1
Next i
график
Image1.Picture = LoadPicture(g)
End Sub
Public Sub график()
n = Application.CountA(Worksheets(1).Range("A:A"))
Range("A1:B" & CStr(n)).Select
Charts.Add
ActiveChart.ChartType = xlXYScatterLinesNoMarkers
ActiveChart.SetSourceData Source:=Sheets("Лист1").Range("A1:B" &
CStr(n))
ActiveChart.Location Where:=xlLocationAsObject, Name:="Лист 1"
ActiveChart.Export "d:\grafik.gif"
g = "d:\grafik.gif"
End Sub

Private Sub CommandButton2_Click()
Worksheets(1).Range("A:B").Select
Selection.Clear
'Worksheets(1).ChartObject.Delete
Image1.Picture = LoadPicture("")
End Sub

Private Sub CommandButton3_Click()
UserForm1.Hide
End Sub

Private Sub UserForm_Initialize()
Image1.PictureAlignment = fmPictureAlignmentTopLeft
Image1.PictureSizeMode = fmPictureSizeModeStretch
End Sub

```

Контрольные вопросы

- 1 Назначение элемента управления Image.
- 2 Свойства элемента управления Image.



23 Лабораторная работа № 23. Программирование на алгоритмическом языке

23.1 Создание с помощью VBA графических объектов в AutoCAD

Цель работы: изучение основных графических методов VBA для AutoCAD; получение навыков создания графических объектов в AutoCAD.

В среде VBA графические примитивы создаются с помощью методов. Для каждого из примитивов существует свой отдельный метод.

Команда построения отрезка

RetVal = object.AddLine(StartPoint, EndPoint)

где RetVal – название создаваемого объекта «отрезок»;

object – объект или объекты, к которым применяется метод AddLine. В качестве объектов могут быть использованы ModelSpace Collection, Paper Space Collection, Block;

StartPoint – координаты первой точки отрезка в трехмерном пространстве. Представляет собой одномерный массив, состоящий из трех элементов, имеет тип Double;

EndPoint – координаты второй точки отрезка в трехмерном пространстве. Представляет собой одномерный массив, состоящий из трех элементов, имеет тип Double.

Пример 1

```
Sub Example_AddLine()
Dim lineObj As AcadLine
Dim startPoint(0 To 2) As Double
Dim endPoint(0 To 2) As Double
'Определяем начальную и конечную точки отрезка
startPoint(0) = 1#: startPoint(1) = 1#: startPoint(2) = 0#
endPoint(0) = 5#: endPoint(1) = 5#: endPoint(2) = 0#
' Помещаем отрезок в модельное пространство
Set lineObj = ThisDrawing.ModelSpace.AddLine(startPoint, endPoint)
ZoomAll
End Sub
```

Команда построения полилинии, проходящей через заданные вершины

RetVal = object.AddLightweightPoliline(Verticeslist)

где RetVal – название создаваемого объекта «полилиния»;

object – объект или объекты, к которым применяется метод AddLine. В качестве объектов могут быть использованы ModelSpace Collection, Paper Space Collection, Block;



Verticeslist – двумерные координаты вершин полилинии. Представляет собой одномерный массив, состоящий из трех элементов, имеет тип **Double**. Требуется не менее двух точек (четыре элемента) для создания полилинии. Размер массива должен быть кратен 2.

Команда создания окружности

RetVal = object.AddCircle(Center, Radius)

где **RetVal** – название создаваемого объекта «окружность»;

Object – объект или объекты, к которым применяется метод **AddCircle**. В качестве объектов могут быть использованы **ModelSpace Collection**, **PaperSpace Collection**, **Block**;

Center – координаты центра окружности в трехмерном пространстве. Представляет собой одномерный массив, состоящий из трех элементов, имеет тип **Double**;

Radius – радиус окружности, имеет тип **Double**.

Пример 2

```
Sub Example_AddCircle()
```

```
'Объявление объекта «Круг», где AcadCircle – тип объявляемого объекта
```

```
Dim circleObj As AcadCircle
```

```
'объявление массива, представляющего собой координаты центра круга в трехмерном пространстве.
```

```
Dim centerPoint(0 To 2) As Double
```

```
'объявление величины радиуса круга
```

```
Dim radius As Double
```

```
' Определяем параметры круга
```

```
centerPoint(0) = 10: centerPoint(1) = 10: centerPoint(2) = 0
```

```
radius = 5
```

```
' Помещаем круг в модельное пространство
```

```
Set circleObj = ThisDrawing.ModelSpace.AddCircle(centerPoint, radius)
```

```
ZoomAll
```

```
End Sub
```

Команда построения 2D-многоугольника

RetVal = object.AddSolid (Point1, Point2, Point3, Point4)

где **RetVal** – название создаваемого объекта «многоугольник»;

object – объект или объекты, к которым применяется метод **AddSpline**. В качестве объектов могут быть использованы **ModelSpace Collection**, **PaperSpace Collection**, **Block**;

Point1, Point2, Point3, Point4– трехмерные координаты первой, второй, третьей и четвертой точек многоугольника. Представляют собой одномерный массив, состоящий из трех элементов, имеет тип **Double** – трехмерный вектор, определяющий направление кривой сплайна в первой точке.



Первые две точки определяют один край многоугольника. Третья точка определена по диагонали напротив второй. Если четвертая точка установлена равной третьей, то создан треугольник.

Команда загрузки типа линий

object.Load LinetypeName, FileName

где object – коллекция Linetypes.

С помощью метода Load в данную коллекцию загружается указанный тип линии из файла, содержащего набор типов линий;

LinetypeName – имя загружаемого типа линии. Имеет тип String. Режим: только ввод (inputonly);

FileName – имя файла, содержащего набор типов линий. Имеет тип String. Режим: только ввод (input-only).

Команда установки указанного типа линии активным (текущим)

object.Linetype

где object – все объекты рисования, у которых изменяется тип линии с помощью свойства Linetype;

Linetype – свойство, устанавливающее или определяющее тип линии существующего объекта. Режим – чтение-запись (read-write).

Пример 3

Данный пример пытается найти уже загруженный тип линии "ACAD_ISO10W100". Если данный тип не найден, то он добавляется из файла acadiso.lin. Затем создается отрезок и к нему применяется тип линии "ACAD_ISO10W100". Поиск типа линии "ACAD_ISO10W100" осуществляется в загруженных типах линии.

```
Sub Load_ActiveLinetype()
```

```
'объявляется переменная, в которой будет содержаться имя типа линии.
```

```
Dim linetypeName As String
```

```
'переменной присваивается название, уже содержащееся в 'файле"acadiso.lin", содержащем типы линий. Данный тип линии заносится в 'коллекцию уже существующих (загруженных) типов линий
```

```
linetypeName = "ACAD_ISO10W100"
```

```
'коллекцию Linetypes (типы линий) объекта ThisDrawing (текущий чертеж) загружается тип линии, содержащийся в файле "acadiso.lin", название типа линии содержится в переменной linetypeName.
```

```
ThisDrawing.Linetypes.Load linetypeName, "acadiso.lin"
```

```
'Активным типом линии становится "ACAD_ISO10W100" и построения будут вестись с данным типом линии ("осевая линия")
```

```
ThisDrawing.ActiveLinetype = ThisDrawing.Linetypes.Item("ACAD_ISO10W100")
```

```
'Создание полилинии
```

```
Dim plineObj As AcadLWPolyline
```

```
Dim points(0 To 11) As Double
```

```
points(0) = 1: points(1) = 1
```

```
points(2) = 1: points(3) = 2
```



```

points(4) = 2: points(5) = 2
points(6) = 3: points(7) = 2
points(8) = 4: points(9) = 4
points(10) = 4: points(11) = 1
Set plineObj = ThisDrawing.ModelSpace.AddLightWeightPolyline(points)
plineObj.Closed = true

```

'свойству `LinetypeScale` (масштаб типа линий) присваивается значение 0,3. Без использования этого свойства у объектов малых размеров тип линии не будет виден на чертеже, а объект будет нарисован сплошной линией.

```

plineObj.LinetypeScale = 0.3
End Sub

```

Команда использования веса линии

object.LineWeight

где `object` – все объекты рисования, у которых изменяется вес линии с помощью свойства `LineWeight`, а также объект `Layer` (слой);

`LineWeight` – свойство, определяющее вес (толщину линии) для указанного нарисованного объекта либо для установки веса линии по умолчанию, после чего все построения будут вестись указанным весом линии. Данное свойство может принимать следующие значения: `acLnWtByLayer`, `acLnWtByBlock`, `acLnWtByLwDefault`, `acLnWt000`, `acLnWt005`, `acLnWt009`, `acLnWt013`, `acLnWt020`, `acLnWt025`, `acLnWt030`, `acLnWt050`, `acLnWt053`, `acLnWt060`, `acLnWt090`, `acLnWt100`, `acLnWt106`, `acLnWt158`.

Вес линии величиной 0 строит тончайшую линию, позволяемую указанным печатающим устройством, и показывает толщину в 1 пиксель в пространстве модели.

Пример 4

```

Sub Example_LineWeight()
Dim circleObj As AcadCircle
Dim centerPoint(0 To 2) As Double
Dim radius As Double
' Определяем окружность
centerPoint(0) = 0#: centerPoint(1) = 0#: centerPoint(2) = 0#
radius = 5#
' Создаем окружность в пространстве модели
Set circleObj = ThisDrawing.ModelSpace.AddCircle(centerPoint, radius)
ZoomAll
' Определяем вес линии у окружности
MsgBox "Текущий вес линии для окружности – " & circleObj.LineWeight
' Изменяем вес линии объекта
circleObj.LineWeight = acLnWt211
circleObj.Update
MsgBox "Текущий вес линии для окружности – " & circleObj.LineWeight
End Sub

```



Команда создания однострочного текста

RetVal = object.AddText(TextString, InsertionPoint, Height)

где RetVal – название создаваемого объекта «текст»;

object – объект или объекты, к которым применяется метод AddText. В качестве объектов могут быть использованы ModelSpace Collection, PaperSpace Collection, Block;

TextString – фактический текст, который будет размещен, тип String;

InsertionPoint – трехмерные координаты точки вставки текста. Представляют собой одномерный массив, состоящий из трех элементов, тип Double;

Height – положительное число типа Double, определяющее высоту текста.

Команда создания многострочного текста

RetVal = object.AddMText(InsertionPoint, Width, Text)

где RetVal – название создаваемого объекта «многострочный текст»;

object – объект или объекты, к которым применяется метод AddText. В качестве объектов могут быть использованы ModelSpace Collection, PaperSpace Collection, Block;

InsertionPoint – трехмерные координаты точки вставки текста. Представляют собой одномерный массив, состоящий из трех элементов типа Double;

Width – положительное число типа Double, определяющее ширину текста;

Text – фактический текст, который будет размещен в поле многострочного текста, имеет тип String.

Пример 5

```
Sub CreateText()
Dim mtextObj As AcadText
Dim insertPoint(0 To 2) As Double
Dim height As Double
Dim textString As String
insertPoint(0) = 50
insertPoint(1) = 50
insertPoint(2) = 0
height = 10
textString = "Это однострочный текст"
' Создаем текстовый объект в пространстве модели
Set mtextObj = ThisDrawing.ModelSpace. _
AddText(textString, insertPoint, height)
width = 8
textString = "Это многострочный текст"
' Создаем текстовый объект в пространстве модели
Set mtextObj = ThisDrawing.ModelSpace. _
AddMText(insertPoint, width, textString)
ZoomAll
End Sub
```



Задание

Вычертить в AutoCAD схему механизма (рисунок 23.1) в заданном положении в соответствии с вариантом. Исходные данные для выполнения задания представлены в таблице 23.1.

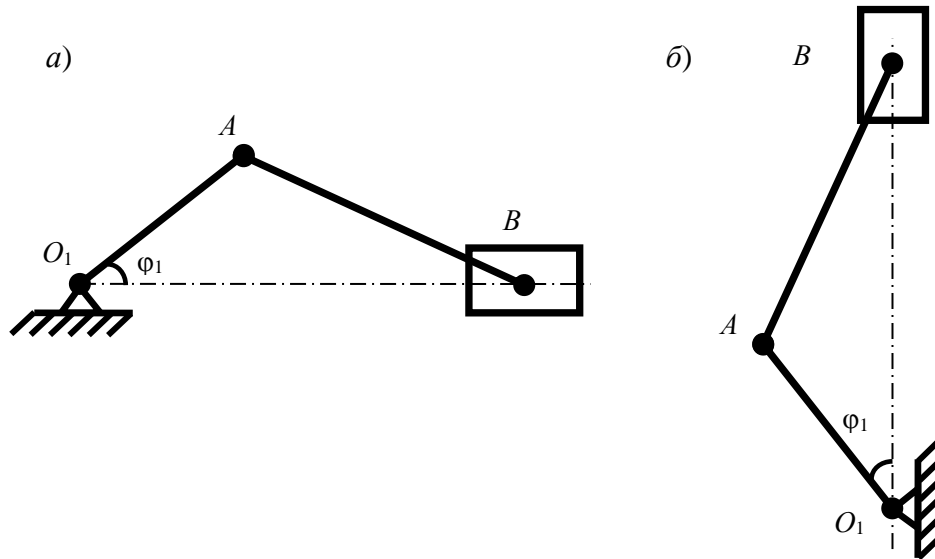


Рисунок 23.1 – Схемы механизма

Таблица 23.1 – Исходные данные

Вариант	Схема (см. рисунок 23.1)	Длина кривошипа l_{O_1A} , м	Длина шатуна l_{AB} , м	Угол поворота φ_1 , град
1	a	0,1	$1,1 \cdot l_{O_1A}$	20
2	б	0,15	$1,2 \cdot l_{O_1A}$	40
3	a	0,2	$1,3 \cdot l_{O_1A}$	60
4	б	0,25	$1,4 \cdot l_{O_1A}$	80
5	a	0,3	$1,5 \cdot l_{O_1A}$	100
6	б	0,35	$1,6 \cdot l_{O_1A}$	120
7	a	0,4	$1,7 \cdot l_{O_1A}$	140
8	б	0,45	$1,8 \cdot l_{O_1A}$	160
9	a	0,5	$1,9 \cdot l_{O_1A}$	180
10	б	0,55	$1,1 \cdot l_{O_1A}$	200
11	a	0,6	$1,2 \cdot l_{O_1A}$	220
12	б	0,65	$1,3 \cdot l_{O_1A}$	240
13	a	0,7	$1,4 \cdot l_{O_1A}$	260
14	б	0,75	$1,5 \cdot l_{O_1A}$	280
15	a	0,8	$1,6 \cdot l_{O_1A}$	300

Контрольные вопросы

- 1 Команда построения отрезка.
- 2 Команда построения окружности.
- 3 Команда изменения веса линий.
- 4 Команда изменения типа линий.

24 Лабораторная работа № 24. Программирование на алгоритмическом языке

24.1 Создание графических объектов в AutoCAD по данным из Excel через VBA

Цель работы: изучение способов связи AutoCAD и Excel через VBA; получение навыков создания графических объектов в AutoCAD по данным из Excel.

Для взаимодействия с другими приложениями через ActiveX нужно выполнить три основные операции:

- 1) установить ссылку на другое приложение;
- 2) создать экземпляр этого приложения;
- 3) написать программу, использующую методы и свойства приложения.

Чтобы сделать ссылку на объектную библиотеку другого приложения, нужно в меню *Tools – References* указать нужное, после чего в окне просмотрщика объектов будут доступны объекты другого приложения. Чтобы создать экземпляр приложения, например, MSExcel, объявляется переменная-ссылка

Dim ExcelAppObj as Excel.Application

и устанавливается указатель

Set ExcelAppObj = New Excel.Application

По окончании работы нужно закрыть запущенный экземпляр приложения: ExcelAppObj.Application.Quit.

Задание

Установив связь с Excel, построить график функции из задания (см. лабораторную работу № 22) в AutoCAD, используя данные, расположенные на листе Excel.

Пример выполнения задания

```
Sub ExcelToAutocad()
' описываем переменные для связи с Excel
Dim AP As Excel.Application
Dim WB As Excel.Workbook
Dim WS As Excel.Worksheet
'Устанавливаем связь с Excel
Set AP = Excel.Application
Set WB = AP.Workbooks.Open("полный путь к файлу Excel с данными для построения")
Set WS = WB.Worksheets("Лист1")
    Dim plineObj As AcadLWPolyline
    Dim points(0 To 13) As Double
    Dim splineObj As AcadSpline
    Dim startTan(0 To 2) As Double
```



```

Dim endTan(0 To 2) As Double
Dim fitPoints(0 To 20) As Double
' определяем точки для полилинии
For i = 0 To 13 Step 2
    points(i) = Cells(i + 2, 4)
    points(i + 1) = Cells(i + 2, 5)
Next i
' определяем точки для сплайна
startTan(0) = 0.5: startTan(1) = 0.5: startTan(2) = 0
endTan(0) = 0.5: endTan(1) = 0.5: endTan(2) = 0
k = 1
For i = 1 To 20 Step 3
    fitPoints(i - 1) = Cells(k + 1, 4)
    fitPoints(i) = Cells(k + 1, 5)
    fitPoints(i + 1) = 0
    k = k + 1
Next i
' создаем полилинию в пространстве модели
Set plineObj = ThisDrawing.ModelSpace.AddLightWeightPolyline(points)
' создаем сплайн в пространстве модели
Set splineObj = ThisDrawing.ModelSpace.AddSpline(fitPoints, startTan, endTan)
ZoomAll
'закрываем Excel
'AP.Quit
End Sub

```

Контрольные вопросы

- 1 Как установить связь Excel и AutoCAD?
- 2 Как получить данные из Excel?
- 3 Как построить сплайн по данным из Excel?

Список литературы

- 1 **Каймин, В. А.** Информатика: учебник / В. А. Каймин. – 6-е изд. – Москва: ИНФРА-М, 2010. – 285 с.
- 2 **Васильев, А. Н.** Excel 2010 на примерах: практическое пособие / А. Н. Васильев. – Санкт-Петербург: БХВ-Петербург, 2010. – 422 с.
- 3 **Макаров, Е. Н.** Инженерные расчеты: учебное пособие / Е. Н. Макаров. – Санкт-Петербург: Питер, 2011. – 400 с.
- 4 **Соколова, Т. Ю.** AutoCAD 2016. Двухмерное и трехмерное моделирование: учебный курс / Т. Ю. Соколова. – Москва: ДМК Пресс, 2016. – 756 с.: ил.
- 5 **Кильдишов, В. Д.** Использование приложения MS Excel для моделирования различных задач: практическое руководство / В. Д. Кильдишов. – Москва: СОЛОН-Пресс, 2015. – 156 с.

