

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

---

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ИНСТИТУТ  
ТОЧНОЙ МЕХАНИКИ И ОПТИКИ  
(ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ)

Н.Д. Толстоба

**КОМПЬЮТЕРНОЕ КОНСТРУИРОВАНИЕ  
ОПТИЧЕСКИХ ПРИБОРОВ**

УЧЕБНОЕ ПОСОБИЕ

Санкт-Петербург

2000

Толстоба Н.Д. Компьютерное конструирование оптических приборов.  
Учебное пособие. - СПб, 2000. - 85 с.

Излагаются основы работы в среде среды автоматизированного конструирования AutoCAD. На примере построения конструкторской документации на оптические детали происходит изучение AutoCAD и его специальных возможностей. С целью создания параметрических чертежей на детали и узлы оптических приборов излагаются основы программирования на AutoLISP.

Для студентов оптических и приборостроительных направлений и специальностей.

Пособие подготовлено на кафедре прикладной и компьютерной оптики Санкт-Петербургского государственного института точной механики и оптики (технического университета).

Составитель: аспирант Н.Д. Толстоба

Рецензенты : д.т.н., проф. С.А. Родионов; к.т.н., доц., Н.Б. Вознесенский

Одобрено на заседании кафедры Прикладной и компьютерной оптики  
7 июня 1999 г., протокол № 10.

© Санкт-Петербургский государственный  
институт точной механики и оптики  
(технический университет), 2000

© Н.Д.Толстоба, 2000

## **ВВЕДЕНИЕ**

Данный курс предназначен для ознакомления студентов с компьютерными системами и методами конструирования, а также получения практических навыков автоматизированного конструирования и выпуска чертежей с помощью системы AutoCAD.

### **Рекомендации преподавателю**

Первой части курса - непосредственному ознакомлению студентов с системой - следует уделить особое внимание, так как именно этот этап является залогом успешной дальнейшей работы по изучению дисциплины. На процесс освоения и привыкания к среде AutoCAD должно быть потрачено 1/3 всего времени обучения. Таким образом, первые два задания имеют своей целью познакомить обучаемых с пакетом AutoCAD.

На изучение дополнительных возможностей AutoCAD оставлено 2/3 времени изучения дисциплины. Наиболее преуспевшим в области программирования параметрических чертежей предлагается защищать бакалаврский проект по данному направлению и поступать в магистратуру с целью дальнейшего развития дополнительных возможностей среды автоматизированного конструирования AutoCAD.

### **Рекомендации студенту**

Для успешного освоения среды автоматизированного конструирования AutoCAD необходимы практические занятия. Своевременное выполнение продуманных преподавателем заданий помогает быстрому освоению пакета AutoCAD и получению навыков конструирования с его помощью.

Особенно интересным является раздел разработки собственных дополнительных возможностей в среде AutoCAD. Но для его плодотворного изучения нужно хорошо освоить работу со средой автоматизированного конструирования.

В области дополнительных возможностей AutoCAD существует множество задач, нерешенных в мире, или решенных только на английском языке и по зарубежным стандартам. Так что для начинающих работать в среде AutoCAD есть большой выбор направления деятельности. Преуспевшие в области программирования параметрических чертежей смогут защищать бакалаврский проект по данному направлению и поступить в магистратуру с целью дальнейшего изучения и развития возможностей среды AutoCAD.

# **1. ОСНОВЫ РАБОТЫ С УНИВЕРСАЛЬНОЙ ПРОГРАММОЙ АВТОМАТИЗИРОВАННОГО КОНСТРУИРОВАНИЯ AUTOCAD**

## **1.1. Обзор современного программного обеспечения в области CAD/CAE/CAM**

Первый вариант AutoCAD появился много лет назад. Это была сильная программа двухмерного черчения, и последующие версии создавались на ее основе. Сегодня AutoCAD является доминирующим пакетом среди систем САПР. По числу лицензируемых копий AutoCAD опережает все продукты этого класса, вместе взятые.

Известный пакет AutoCAD компании Autodesk является постоянно развивающейся средой проектирования и пользуется у специалистов по САПР заслуженным авторитетом. Это высокопроизводительное программное обеспечение с Windows-интерфейсом позволяет одинаково хорошо решать задачи, касающиеся различных областей проектирования, в том числе машиностроения, строительства, архитектуры, электроники, экономических оценок в проектировании, управления инфраструктурой и другие. На основе этой системы или, как принято говорить, в среде AutoCAD созданы различные приложения, позволяющие проектировать дома, печатные платы, станки и роботы, одежду и т.д.

Данное средство предоставляет пользователю возможности, которые ранее могли быть реализованы только на больших и дорогих вычислительных системах. С помощью AutoCAD может быть построен любой рисунок, если только его можно нарисовать вручную. Другими словами, AutoCAD способен выполнять практически любые виды графических работ.

В основу структуры AutoCAD был положен принцип открытой архитектуры, позволяющий адаптировать и развивать многие функции

AutoCAD применительно к конкретным задачам и требованиям. Это позволяет создавать собственные:

- экранные, планшетные, кнопочные, падающие и графические меню для автоматизации часто выполняемых действий;
- пакетные файлы для автоматизации выполнения длинных последовательностей команд;
- шрифты;
- типы линий;
- образцы штриховок;
- библиотеки символов и фрагментов рисунков;
- образцы штриховок формата PostScript;
- рисунки-прототипы с заданными значениями параметров по умолчанию.

Пакет поддерживает множество различных устройств ввода-вывода, включая графические мониторы с высоким разрешением, дигитайзеры и манипуляторы, графопостроители, принтеры и даже фотонаборные машины. К достоинствам пакета во всех версиях можно отнести хорошую его настройку на работу с плоттерами - основными аппаратными средствами распечатки чертежей компьютерного производства.

## **1.2. Основные понятия AutoCAD**

### **1.2.1. Примитивы**

В отличие от "художественных" графических редакторов, AutoCAD работает не с изображением как таковым, а с геометрическим описанием объектов, составляющих изображение. Так, например, отрезок во внутреннем представлении графического редактора AutoCAD описывается двумя точками, круг описывается центром и радиусом.

Все примитивы AutoCAD обладают рядом свойств (принадлежность слою, цвет, тип линии, ширина). Некоторые из этих свойств (например, цвет) присущи всем примитивам.

### **1.2.2. Системы координат**

Используется традиционная декартова система координат. Можно ввести пользовательские системы координат с помощью команды USC. В определенный момент времени пользователь работает только с одной предварительно выбранной системой координат, которая называется текущей. Вся работа с изображением проводится в текущей системе координат.

### **1.2.3. Единицы измерения и масштаб**

Расстояния между точками на рисунке измеряются в условных единицах. Конкретный формат представления размеров (дюймы, футы, миллиметры и др.) не имеет значения для AutoCAD. Иначе говоря, при создании объектов в чертеже AutoCAD "измеряет" все расстояния в относительных единицах. В AutoCAD нет масштаба в обычном понимании конструктора, конструктор задает все расстояния и координаты в реальных единицах - мы как бы работаем в масштабе 1:1.

Масштабирование различных частей изображения в соответствии с желаемым форматом документа может осуществляться в момент компоновки

чертежа (команда **SCALE** (МАСШТАБ)) или при выводе чертежа или его части на плоттер (принтер).

#### **1.2.4. Вид**

Когда вы создаете чертеж, то работаете с изображением части чертежа, выводимой на дисплей. Будем называть эту часть изображения видом. Эту видимую часть чертежа (окно зрения) можно увеличивать (при этом изображение чертежа будет уменьшаться), уменьшать (изображение будет увеличиваться) или перемещать по полю чертежа без изменения масштаба отображения (панорамирование). Изменение вида осуществляется командой **ZOOM**.

#### **1.2.5. Слой**

**AutoCAD** дает возможность распределять выбранные фрагменты чертежа по различным слоям. Работая за кульманом, конструктор имеет дело с одним листом бумаги и располагает изображения объектов на нем и только на нем. В среде **AutoCAD** располагать изображение можно как бы на нескольких совмещенных в пространстве носителях (это можно сравнить с наложенными друг на друга прозрачными кальками). Например, чертеж может содержать на одном слое построение призмы со всеми вспомогательными линиями, на другом - сам чертеж в окончательном исполнении, а на третьем - ход лучей в призме.

Количество слоёв не ограничивается. Слои можно делать видимыми и невидимыми. С каждым слоем чертежа связывается цвет и тип линий. По мере создания чертежа вы можете вводить новые слои, менять свойства существующих. Подобно системе координат, в любом чертеже **AutoCAD** всегда существует по крайней мере один слой с именем "0".

#### **1.2.6. Чертёж**

Чертёж - это файл, содержащий некую графическую и вспомогательную информацию, полностью описывающую графический объект.



AutoCAD предоставляет в распоряжение конструктора все вычислительные ресурсы микроЭВМ и новые средства создания и редактирования чертежа.

### 1.3 Интерфейс программы

#### Графический редактор и система команд AutoCAD

Графический редактор - это программа, отображающая на экране графическую информацию и исполняющая команды создания, изменения, просмотра чертежа и вывода его на плоттер или принтер.

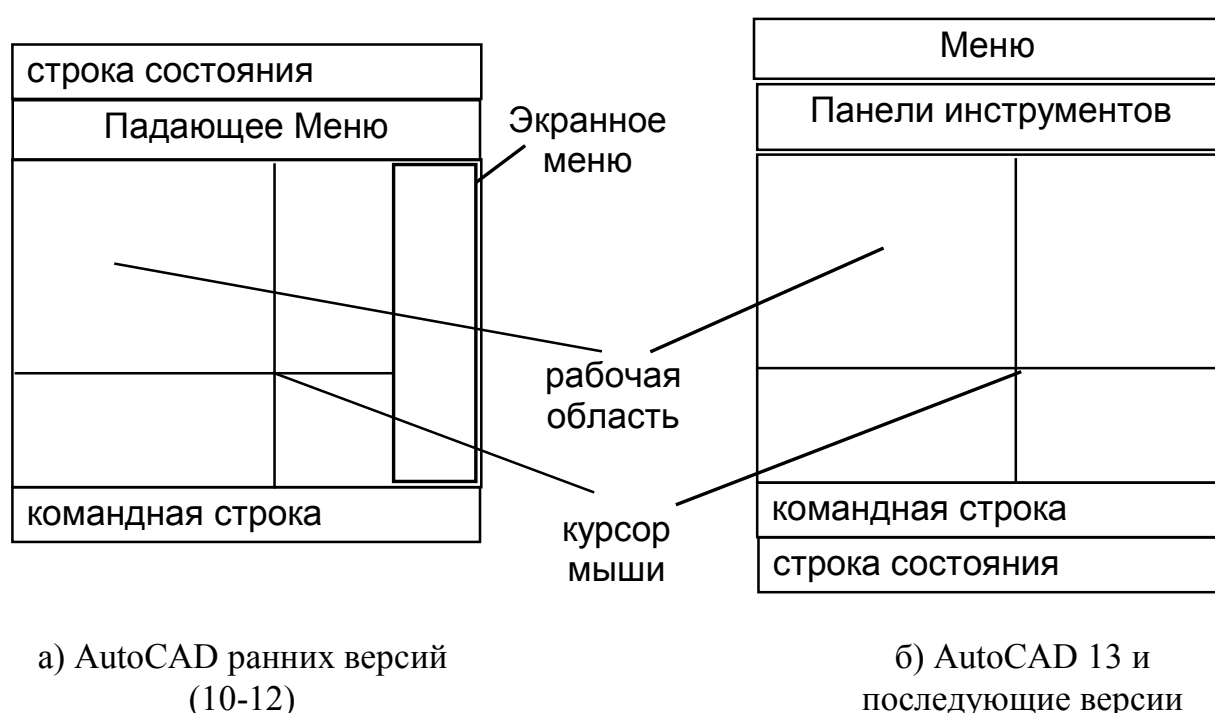


Рис. 1. Вид поля экрана при работе с ACAD

#### Командная строка

При работе с AutoCAD всегда следите за сообщениями в командной строке - в ней происходит формирование команд и в нее выводятся все сообщения графического редактора. Условно можно выделить два режима работы - режим выполнения команды и режим ожидания команды. В момент ожидания команды вы видите в зоне командной строки экрана подсказку:

Command:

В этой зоне помещается три строки. Если нужно посмотреть на предыдущие строки, нажмите клавишу "F2". Откроется окно текстового режима, и в нем появится ограниченное количество строк протокола работы графического редактора. Повторное нажатие F2 вернет вас в графический режим.

### **Функции клавиатуры**

С клавиатуры производится ввод команд. Нажатия Space, Enter и правой клавиши мыши эквивалентны и соответствуют вводу команды или значения. Если команда не задана в командной строке, то происходит вызов предыдущей команды.

Отключение команды

- в 10-й версии - Ctrl + C
- в 14-й - ESC.

### **Строка состояния**

Строка состояния информирует нас о том, каковы координаты курсора мышки в данный момент, на каком уровне чертежа мы работаем (LAYER) и о привязке к прямоугольной системе (ORTO).

### **Панели инструментов (Toolbars)**

Toolbars - основное устройство для ввода команд. После ввода команды диалог либо переходит в командную строку, либо открывается окошко для продолжения выбора опций (это так называемое графическое меню).

### **Графические меню**

Известно, что человек визуально легче и быстрее воспринимает графическую информацию, чем текстовую, и графические меню призваны сделать общение с AutoCAD еще более наглядным.

Выберите из строки падающих меню пункт Draw (Рисуй), а затем из падающего меню пункт Hatch... (Штриховка). На экране появится

графическое меню, в котором показаны типы возможных стандартных штриховок.

### **Рабочая область**

Это область экрана, на которой располагается непосредственно результат вашей работы (чертёж, рисунок).

## 2. СОЗДАНИЕ ИЗОБРАЖЕНИЙ

В системе AutoCAD любое изображение создается с помощью базового набора примитивов. Каждая команда предоставляет пользователю несколько способов построения одного и того же объекта по заданным геометрическим параметрам. При указании точек на экране появляются небольшие крестики, которые не являются элементами чертежа и будут удалены после любой перерисовки.

### 2.1. Команды общего назначения

#### 2.1.1. Способы задания точек в поле чертежа

##### - вводом координат с клавиатуры

10.5, 10.25, 0 - абсолютные декартовы координаты, по умолчанию Z присваивается значение текущего уровня

10-1/2, 10-1/4 - те же координаты, только в дробном формате

33<65 - абсолютные полярные координаты

15.7<45<42 - абсолютные сферические координаты

77<56.7,85 - абсолютные цилиндрические координаты

@10,10 - относительные координаты (относительно предыдущей точки )

@10<10 - относительные полярные координаты

##### - указанием точек прямо в поле чертежа

##### - любыми комбинациями этих способов

### 2.1.1. Команды вычерчивания примитивов

#### POINT (ТОЧКА)

Точка - самый простой геометрический объект. Как примитив AutoCAD она имеет дополнительные свойства: принадлежность слою, цвет, высота, на экране может отображаться различными знаками (их около 20).

К точке можно привязаться (**NODE** - узел)

Вид знака отображения точки задается значением системной переменной **PDMODE**, а размеры - переменной **PDSIZE**. Значения этих переменных можно установить, используя команду **SETVAR**.

Описание	Команда <b>Point</b> (ТОЧКА) рисует в пространстве X,Y,Z примитив "точка", который может использоваться в дальнейшем как <b>NODE</b> (УЗЕЛ) в режиме объектной привязки.
Внимание!	На точки, размещенные в слое <b>DEFPOINTS</b> при ассоциативном образмеривании, установки <b>PDMODE</b> и <b>PDSIZE</b> не влияют.  В отличие от временных крестообразных маркеров, точки являются частью рисунка и выводятся на плоттер. Если вас это не устраивает, вы можете либо стереть их, либо создать в отдельном слое и отключить его, либо присвоить <b>PDMODE</b> значение, равное 1.

#### ELLIPSE (ЭЛЛИПС)

Описание	Команда <b>ELLIPSE</b> позволяет строить эллипсы несколькими способами. Стандартный метод использует большую и малую оси, которые определяются по конечным точкам	
Опции	<b>Center point</b>	Задайте центральную точку и одну конечную точку каждой оси.
	<b>Rotation</b>	Задайте поворот вокруг большой оси. Угол поворота может быть в пределах от 0 до 89.4

		градусов.
--	--	-----------

## LINE (ОТРЕЗОК)

Отрезок - наиболее часто используемый примитив. Для вычерчивания отрезков прямых используется команда **LINE**.

Описание	Команда <b>LINE</b> позволяет рисовать отрезки прямых. Конечные точки могут быть заданы в 2М или 3М координатах.	
Опции	Продолжи	Предыдущий отрезок или дугу можно продолжить, ответив на запрос "От точки:" нажатием клавиши пробела.
	Close Замкни	Если последовательность отрезков должна образовывать замкнутый многоугольник, на последний запрос "To point:" можно ввести <b>C (Close)</b> . Тогда многоугольник "замкнется" сам.
	Undo Отмени	Чтобы стереть последний нарисованный сегмент, не покидая команды <b>LINE</b> , следует ответить <b>U</b> на последний запрос "To point: ".
Совет	Если вы хотите рисовать линиями различной толщины, используйте вместо отрезков полилинии или полосы. Другой способ - рисовать отрезки разными цветами, которым при выводе на плоттер задавать разную толщину перьев.	
Внимание!	Использование опции <b>Undo (Отмени)</b> отличается от команды <b>UNDO</b> . Первая отменяет последний нарисованный отрезок, а вторая - любую предыдущую команду целиком.	

## DONUT (КОЛЬЦО)

Описание	Команда <b>Donut</b> отрисовывает закрашенный круг или кольцо. Кольцевые объекты строятся из замкнутых широких дуговых сегментов полилиний. Команда запрашивает значения внутреннего и внешнего диаметров.
Совет	Кольцо можно преобразовать в цилиндр, присвоив ему

	высоту (толщину по оси $Z$ ).
--	-------------------------------

## CIRCLE (КРУГ)

Эта команда позволяет начертить круг пятью различными способами.

Описание	Команда CIRCLE отрисовывает круги.	
Подсказки	3P/2P/TTR/<Center point>: Diameter/<Radius>:	3T/2T/ККР/<Центр>: Диаметр/<Радиус> :
Опции	Center point	Задаёт центральную точку.
	Diameter	Задаёт диаметр.
	Radius	Задаёт радиус путем введения значения или указания двух точек.
	3P	Задаёт три точки на окружности.
	2P	Задаёт две крайние точки на диаметре.
	TTR	Задаёт две точки (на отрезке, круге или дуге), которые будут касаться окружности; после этого требуется задать радиус.

## ARC (ДУГА)

Способов построения дуги более двенадцати. Математически дуга определяется четырьмя параметрами: центром, радиусом и двумя центральными углами.

Описание	Команда ARC строит дугу (часть окружности). По умолчанию дуга строится по трем параметрам: начальная точка, вторая точка, конечная точка.
Подсказки, Опции	Размер и положение дуги можно задать, указав любую комбинацию следующих опций: Start point, Second point, End point, Center, Angle, Length of chord (Длина хорды), Radius и Direction (Направление). Возможные опции содержатся в подсказках.
Совет	Вы можете задать начальную точку координатами X,Y,Z. Координата Z задает уровень дуги.

Способы вычерчивания окружностей и дуг см. в Приложениях А и Б.



## 2.1.2. Установка режима привязки

### OSNAP (ПРИВЯЖИ)

<p>Описание</p>	<p>Команда <b>Osnap</b> задает один или несколько текущих режимов объектной привязки, которые впоследствии будут использоваться при любом указании точек. Это позволяет привязываться к различным характерным точкам рисунка. Значением по умолчанию является режим <b>None</b> "Откл" или "Ничего".</p> <p>Вводя названия режимов через запятую, можно задать сразу несколько режимов привязки. При указании отдельных точек можно задать индивидуальный режим объектной привязки; при этом действие текущих режимов временно подавляется. Текущие режимы выдаются на экран командой <b>Status</b> (Статус).</p>	
<p>Режимы объектной привязки</p>	<p>CENter ENDpoint INTersection MIDpoint NEArest NODe PERpendicular QUAdrant QUICK TANgent NONE</p>	<p>Центр окружности (дуги) Конец отрезка Ближайшее к курсору пересечение линий Центральная точка отрезка Ближайшая точка на линии Привязка к узлу Перпендикулярно к указанному отрезку Привязка к точкам на окружности, к ее "краям" Ускорение путём снижения точности привязки Касательно к указанной кривой Нет привязки</p>

**Закрепление.** Практические навыки отрабатываются в процессе выполнения заданий 1 и 2 при вычерчивании контура линзы и призмы. См. Приложения В и Г.

## 2.2. Свойства примитивов

Все примитивы обладают рядом общих свойств. Самыми понятными и простыми являются цвет и тип линии.

### 2.2.1. Цвет

Каждый примитив характеризуется цветом. Текущий цвет может быть изменен с помощью команды **COLOR** (ЦВЕТ).

Описание	Команда <b>COLOR</b> не изменяет цвета уже отрисованных примитивов.
Совет	Для изменения цвета примитивов используйте команду <b>Change</b> (ИЗМЕНИ).

### Коды цветов AutoCAD:

Основные цвета		Дополнительные цвета	
1	Красный	8	Темно-серый
2	Желтый	9	Темно-красный
3	Зеленый	10	Коричневый
4	Голубой	11	Темно-зеленый
5	Синий	12	Темно-голубой
6	Фиолетовый	13	Темно-синий
7	Черный (белый)	14	Темно-фиолетовый
		15	Белый

### 2.2.2. Тип линии

Все основные двумерные примитивы характеризуются типом линии.

Тип линии - это шаблон (последовательность чередующихся линейных сегментов), по которому вычерчиваются линии в **AutoCAD**.

## LINETYPE (ТИПЛИН)

Описание	Команда <b>LINETYPE</b> изменяет текущий тип линии.	
Подсказки:	?/Create/Load/Set:	
Опции	<b>Create</b>	Можно создать свой тип линии
	<b>Load</b>	Загрузка типа линии
	<b>Set</b>	Установка типа линии в качестве текущего

### Типы линий, которые используются чаще всего:

Dashed            — — — —

Dashdot          — · — · —

Continious        —————

### 2.2.3. Масштабирование линий

Для того, чтобы увидеть результаты изменения типа линии, чаще всего следует изменить масштаб для линий, то есть воспользоваться командой **LTSCALE** (ЛМАСШТАБ)

Описание	Команда <b>Ltscale</b> присваивает всем типам линий глобальный масштабный коэффициент. По умолчанию он равен 1.
Совет	Для определения величины <b>Ltscale</b> можно воспользоваться следующим правилом: <b>Ltscale</b> равен величине 0.375, поделенной на масштабный коэффициент.  Например, для рисунка на плоттер в масштабе 1:100 нужно 0.375 разделить на 1/100 или $0.375 \times 100$ , что дает в результате 37.5, или округленно 38. Этот прием дает неплохие результаты для большинства чертежных стандартов.
Внимание!	Вы не сможете увидеть результата до регенерации рисунка.

## 2.3. Штриховка

При создании чертежей часто приходится выполнять разрезы и сечения, в которых используется штриховка.

### НАТСН (ШТРИХ)

Описание	Команду НАТСН можно запустить из графического меню для того, чтобы сразу увидеть стиль выбранной нами штриховки.
Подсказки:	Pattern (? or name / U, style ) <по умолчанию>: Scale for pattern <1.000>: Angle for pattern <0>: Select objects:

Контур штрихования должен быть замкнут и состоять из примитивов, имеющих общие точки.

**Закрепление.** Замена сплошных осевых линий на штрих-пунктирные (тип линии); простановка размеров линзы без использования специальных команд; придание различных цветов линиям построения и лучам

## 2.4. Работа с текстом

Текст - это примитив AutoCAD, имеющий специальные свойства. Он может иметь любые размеры, его можно растягивать, сжимать и зеркально отражать.

Шрифт - это модель вычерчивания символов. Любая буква отображается последовательными векторами, параметры которых заданы в специальном файле. Этот файл на самом деле - простейший файл форм AutoCAD. Это дает возможность создавать свои собственные шрифты.

Вывод текста. Команды DTEXT (ДТЕКСТ), TEXT (ТЕКСТ)

Описание	Различия между этими командами заключаются в следующем: команда TEXT предназначена для ввода только одной текстовой строки, а DTEXT - для многострочного текста. DTEXT позволяет без выхода из команды размещать строки текста в другом месте экрана путем указания новой точки. В команде DTEXT одиночное нажатие клавиши <ENTER> во время ввода букв приводит к перемещению курсора на новую строку, а в команде TEXT - к завершению команды.	
Подсказки:	Start point or Align/Center/Fit/Middle/Right/Style: High <значение по умолчанию>: Angle <значение по умолчанию>: Text:	
Опции	Style	Устанавливает новую гарнитуру шрифта. Гарнитура должна быть загружена командой Style.
	<ENTER>	При нажатии <ENTER> AutoCAD подсвечивает последний введенный текст и предлагает ввести новую текстовую строку прямо под ней с теми же гарнитурой, высотой и углом поворота.
	Angle	Задаёт ориентацию линии текста.
	Text	Эта подсказка приглашает ввести строку текста.

В ответ на запрос нужно выбрать гарнитуру шрифта или один из возможных способов размещения текста

### Способы размещения текста

Start point	В указанную точку помещается нижний левый угол первой буквы строки. Строки выравниваются по левой границе.
Aligned	Эта опция требует указания двух точек, определяющих длину всех вводимых строк. Размер текста определяется числом символов в строке, буквы масштабируются без изменения пропорций так, чтобы заполнить всю строку.
Fit	Текст вписывается в строку заданной длины и выравнивается по высоте. Пропорции букв при этом меняются.
Sen	Задается точка, относительно которой текстовая строка центрируется по горизонтали.
Middle	Задается точка, относительно которой текстовая строка центрируется как по горизонтали, так и по вертикали.
Right	Все строки текста выравниваются по правому краю, которым становится указанная точка.

**Закрепление.** Простановка размеров линзы без использования команды DIM, а также вывод надписей в заданиях 1 и 2.

### **3. РАБОТА ПО МОДЕРНИЗАЦИИ ЧЕРТЕЖА**

Общеизвестно, что в процессе проектирования чертежа конструктор много времени тратит на редактирование. Несмотря на то, что мы с вами, возможно, традиционным способом сможем начертить новый чертеж быстрее, чем с помощью системы AutoCAD, наверняка для исправления его нужно будет потратить немалые усилия, т.е. извести массу стирательной резинки, пожертвовать товарным видом чертежа в пользу его правильности или долго и нудно перечерчивать чертеж. С использованием редакторских возможностей AutoCAD мы можем значительно облегчить свою жизнь.

#### **функции редактирования позволяют:**

- удалять фрагменты изображения;
- восстанавливать случайно удаленные фрагменты;
- перемещать или поворачивать фрагменты или отдельные изображения относительно других;
- копировать созданные фрагменты и располагать их в указанном месте;
- увеличивать или уменьшать объекты;
- создавать зеркальносимметричное изображение;
- изменять свойства (принадлежность к слою, цвет и тип линии) созданных объектов;
- сопрягать линии и строить фаски;
- делить объекты на равные части или размечать на сегменты с заданным интервалом;
- расчленять блоки или полилинии на составные части;
- редактировать полилинии (сглаживать, изменять свойства и т. д.);
- растягивать части рисунка;
- проводить линии, расположенные на заданном (постоянном) расстоянии относительно других.

### 3.1. Перечень команд редактирования

SELECT (ВЫБЕРИ)	Обеспечивает предварительный выбор объектов для дальнейших различных операций редактирования (при ссылке "текущий")
MIRROR (ЗЕРКАЛО)	Позволяет формировать зеркальные отражения существующих объектов, удаляя или сохраняя при этом оригиналы
CHANGE (ИЗМЕНИ)	Изменяет объекты и их свойства
COPY (КОПИРУЙ)	Копирует созданные объекты, оставляя оригиналы нетронутыми, и размещает копии в заданном месте
ARRAY (МАССИВ)	Позволяет получать несколько копий выбранных объектов, группируя их в прямоугольной или круговой матрице
SCALE (МАСШТАБ)	Изменяет размеры одного или нескольких созданных объектов. Изображение при этом изменяется относительно базовой точки
TRIM (ОБРЕЖЬ)	Удаляет части объектов между пересекающимися его другими объектами (т. н. режущими кромками)
UNDO (ОТМЕНИ)	Отменяет последовательно выполненные действия предыдущих команд
MOVE (ПЕРЕНЕСИ)	Обеспечивает плоскопараллельный перенос одного или нескольких объектов в указанное место
ROTATE (ПОВЕРНИ)	Поворачивает группу объектов вокруг заданной точки на заданный угол
DIVIDE (ПОДЕЛИ)	Делит примитив на заданное число равных частей и может размещать блоки в точках деления
OFFSET (ПОДОБИЕ)	Позволяет проводить эквидистантные линии на заданном расстоянии
PEDIT (ПОЛПРЕД)	Позволяет редактировать двумерные и трехмерные полилинии и трехмерные многоугольные сети



MEASURE (РАЗМЕТЬ)	Располагает метки вдоль объекта с заданным интервалом. В точки разметки можно вставлять блоки
BREAK (РАЗОРВИ)	Стирает часть отрезка, полосы, окружности, дуги или двумерной полилинии и разбивает эти примитивы на две части
STRETCH (РАСТЯНИ)	Обеспечивает перемещение выбранной части изображения, сохраняя при этом связь с остальной частью
EXPLODE (РАСЧЛЕНИ)	Заменяет блоки на отдельные примитивы, разбивает полилинии на сегменты. Заменяет многоугольную сеть набором трехмерных граней
PROPERTIES (СВОЙСТВА)	Обеспечивает подмножество действий команды CHANGE (ИЗМЕНИ): изменяет свойства примитивов - цвет, принадлежность слою, тип линии и т. Д.
FILLET (СОПРЯГИ)	Плавно сопрягает примитивы дугами разного (в т. ч. и нулевого) радиуса, а также сопрягает полилинии. При этом "лишние" части удаляются
ERASE (СОТРИ)	Удаляет из чертежа выбранные объекты
CHAMFER (ФАСКА)	Проводит линию фаски, удаляя ненужные части примитивов
EXTEND (УДЛИНИ)	Удлиняет отрезки, дуги и двумерные полилинии до пересечения их с аналогичными примитивами (т. н. Граничными кромками)

Мы не будем рассматривать подробно каждую из команд, Вы и сами прекрасно разберетесь, как с ними работать. Но есть некоторые особенности при использовании команд редактирования чертежа, на которых хотелось бы остановиться.

## **Средства выбора объектов**

Все действия редактирования могут производиться над несколькими объектами одновременно. Поэтому каждая команда редактирования требует выбора объектов, которые будут составлять набор объектов.

Последний набор объектов хранится в памяти, и при выполнении новой команды редактирования можно либо использовать предыдущий набор, либо составить новый.

Процедура выбора объектов одина для всех команд редактирования и включает в себя несколько режимов выбора объектов. Во всех командах редактирования она вызывается автоматически, однако, используя команду **SELECT**, можно составить набор объектов без вызова команд редактирования.

Режимы выбора можно комбинировать, вызывая один режим за другим. Завершение выбора объектов осуществляется нажатием клавиши "ENTER" или правой клавиши "мыши".

При любом режиме выбора объектов могут быть выбраны только видимые на экране примитивы. После каждого указания AutoCAD просматривает изображение на экране и выделяет выбранные объекты подсветкой.

Выбор объектов прицелом является основным режимом выбора объектов. При завершении других режимов выбора объектов он возобновляется автоматически. В этом режиме перекрестие курсора заменяется на квадратик, который называется прицелом выбора объектов. При нажатии на левую клавишу будет выбран один объект, пересекающий или попавший в прицел. Из нескольких объектов будет выбран тот, который был создан последним. Для выбора закрашенных объектов необходимо, чтобы прицел пересекал их край. Размер прицела в пикселах определяется системной переменной **MCKBOX** (устанавливается с помощью **SETVAR**).

### 3.2. Режимы выбора объектов для редактирования

Window (Рамка)	Объекты выбираются путем обрамления их прямоугольной рамкой. Рамка определяется указанием двух точек - ее противоположных вершин. В набор редактирования заносятся те объекты, которые целиком находятся в рамке
Crossing (Секрамка)	То же, что Window, но выбираются и те объекты, которые пересекаются рамкой
Box (Бокс)	Этот режим объединяет в себе свойства Window и Crossing. Если второй угол рамки указывается правее первого, то работает Window, если левее - Crossing
Auto (Авто)	Этот режим объединяет в себе режим указания объектов прицелом и режим Box. При вызове этого режима сначала предлагается указать объекты прицелом; если при таком указании не будет найдено ни одного объекта, то задействуется режим Box, причем в качестве первого угла рамки берется центр прицела при предыдущем указании
Multiple (Несколько)	В этом режиме, в отличие от всех остальных, можно указать несколько объектов; рисунок будет просмотрен только по завершении режима (правая клавиша "мыши" или ENTER)
Single (Единственный)	Выбор этой опции приводит к автоматическому завершению выбора объектов при нахождении первого же примитива
Last (Последний)	Выбирается последний отрисованный объект (из видимых на экране)
Previous (Текущий)	Передает команде редактирования набор объектов, составленный в предыдущей команде редактирования или командой SELECT
Undo (Отмени)	Отменяет результат работы последней опции

Remove (Удали)	По умолчанию объекты заносятся в набор. Выбрав эту опцию, вы можете, еще раз указав подсвеченные объекты, вывести их из набора
Add (Добавить)	Эта подкоманда отменяет предыдущий режим, позволяя продолжить заполнение набора

**Закрепление.** Перенос, копирование, разворот элементов чертежа, создание таблиц, удаление примитивов.

## 4. СЛОИ

### 4.1. Создание слоев

LAYERS - система нескольких уровней (слоев). Изначально работаем на уровне “0” и при загрузке нового чертежа автоматически выходим на нулевой слой.

При загрузке старого чертежа автоматически открывается уровень DEFPOINT, на котором хранятся точки с отметками.

Уровни, слои обладают следующими свойствами: (on/off) включить/выключить отображение того или иного уровня. FROZEN - слой обладает признаком «заморожен». FROZEN - более сильный признак, чем (on/off).

Уровню приписан цвет color, к которому можно обращаться, а также присваивать тип линии linetype.

Команда Settings выводит список слоёв.

Can	Name	On	Frozen	Color	Linetype
	Defpoint	✓	✓	White	Dashed
	Prizma		✓	Magenta	Continuous
...	..... .....	.....	.....	..... ...	..... .....

**Закрепление.** Использование слоев на различных этапах построения чертежа. Все линии построения, ход лучей и окончательный верный вариант расположить на разных слоях. Создание нескольких слоев чертежа в задании 2. Слой Rays - для лучей, слой Size - для размеров, и т.д.

## 5. БЛОКИ

### 5.1. Создание блоков

С помощью команды **BLOCK** можно связать несколько примитивов и присвоить им точку вставки.

### 5.2. Вставка блоков

**INSERT** - команда вставки блока. Можно использовать блоки, которые использовались в другом чертеже и записывать с помощью команды **WBLOCK** (\*.dwg).

Можно ввести различный масштаб по *X* и по *Y* (но в этом случае нельзя будет использовать команду **EXPLODE**)

Команда **MINSERT** позволяет вставлять несколько блоков в шахматном порядке.

**Закрепление.** Создание блоков технических требований, рамки с нужными надписями, таблицы оптических параметров.

## 6. ПРОГРАММИРОВАНИЕ В КОДАХ

Форма - совокупность отрезков прямых или дуг и окружностей.

Вызывается командой SHAPE:

Command: Shape > имя формы

>точка вставки

>масштаб по X и Y

> поворот

.shp - расширение файла, который содержит описание форм.

### 6.1. Файл с текстом программы .shp

Общий вид описания выглядит таким образом:

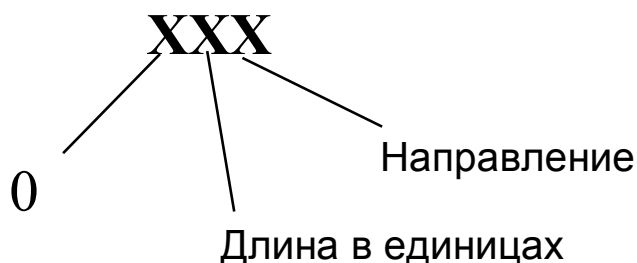
\* № формы, число байт, имя

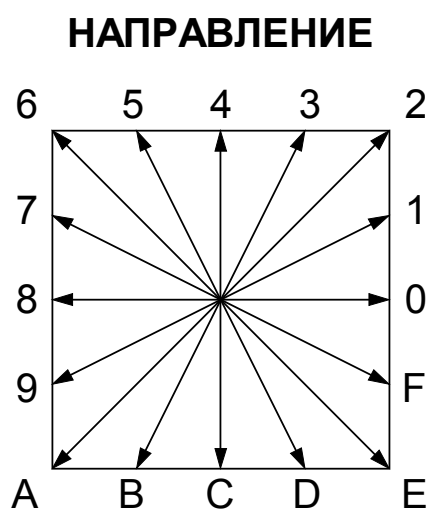
дальше - одна или более строк кодов, разделенных пробелом, запятыми и скобками. Запятыые - обязательные разделители кодов.

Последний код формы - всегда 0. При подсчете числа байт это необходимо учитывать. На каждый код приходится по одному байту, следовательно, подсчет байт заключается в подсчете кодов.

### 6.2. Основные коды

Запись основного кода выглядит таким образом:





**ДЛИНА**

В данном случае все показанные  
ниже отрезки считаются отрезками  
единичной **длины**

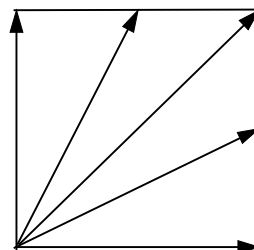


Рис. 2. Основные коды форм

Примеры простейших программ в кодах расположены в Приложении Д.

### 6.3. Вспомогательные коды

**001 - Опустить перо (начать рисовать)**

**002 - поднять перо (прекратить рисовать)**

**003 - масштаб уменьшения**

**004 - масштаб увеличения**

003,2 - все последующие величины делим на 2

004,2 - умножаем на 2

Принцип - не отмена действий, а накопление их.

**005 - текущую точку - в память**

**006 - извлечь из памяти**

**007 - выполнение ранее запрограммированного кода**

007, № субформы

Каждая субформа должна иметь уникальный код и имя.



## 008 - смещение

008,  $\Delta X$ ,  $\Delta Y$

Обязательны 2 параметра.

Диапазон величин  $\Delta X$ ,  $\Delta Y$ : от -127 до +127. Расширения диапазона достигают с помощью масштабирования.

## 009 - цепочка, ломаная

009, {  $\Delta X, \Delta Y$  }, 0, 0

Нули в конце перечисления означают, что запись окончена.

## 00A - Октантные дуги

Описание октантной дуги выглядит таким образом:

00A, R, 0XY

0 - всегда

X - начальная точка

Y - число октантов

Само число 0XY может быть отрицательным, если отсчет октантов производится по часовой стрелке, и положительным, если против.

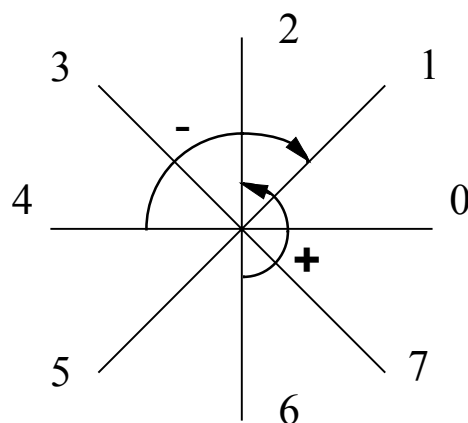


Рис. 3. Схема для октантных дуг

Пример программы вычерчивания дуги показан в Приложении Е.

## 00B - вычерчивание произвольной дуги (долго)

Вычерчивает дугу не обязательно с октантными границами. Определение состоит из 5 байт спецификации.

00B, start\_offset, end\_offset, high\_radius, radius, ( $\pm$ )0XY

Start\_offset и end\_offset показывают, насколько далеко от октантной границы дуга начинается или заканчивается.

Если радиус меньше 255,  $high\_radius = 0$ . Умножьте значение  $high\_radius$  на 256 и прибавьте его к радиусу для отображения дуги радиусом большим, чем 255.  $Radius$  и код дуги ( $\pm$ )0XY такие же, как и для определения октантной дуги (00A). Вы определяете  $start\_offset$ , вычисляя разницу в градусах между начальной октантной границей (производной от 45 градусов) и началом дуги. Затем умножаете на 256 и делите на 45. Если дуга начинается на октанте, начальное смещение равно нулю.  $End\_offset$  вычисляется так же.

Например, дуга от 55 до 95 градусов с  $r=3$  записывается так:

00B,(56,28,0,3,012)

$start\_offset = 56$ , т.к.  $((55 - 45) * 256 / 45) = 56$

$end\_offset = 28$ , т.к.  $((95 - 90) * 256 / 45) = 28$

$high\_radius = 0$ , т.к.  $(radius < 255)$

$radius = 3$

### 00C - дуга по $\Delta X, \Delta Y$ и стрелке

00C,  $\Delta X, \Delta Y, \tilde{h}$

$\Delta X, \Delta Y$  учитывается со знаками

$\tilde{h}$  рассчитывается по формуле:

$$\tilde{h} = \frac{h}{\sqrt{\Delta X^2 + \Delta Y^2}}$$

Самая крутая дуга - половина окружности.

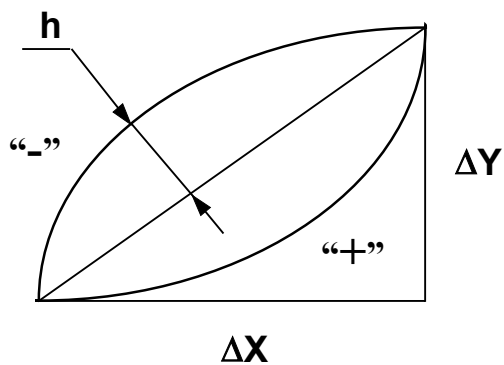


Рис. 4. Схема для дуги по стрелке

Пример программы вычерчивания дуги показан в Приложении Е.

### 00D - последовательность дуг

00D,  $\{\Delta X_i, \Delta Y_i, \tilde{h}_i, \}$  0, 0

## 6.4. Создание шрифта

\* код ASCII, число байт, строчными буквами пояснение

Еще требуются описание шрифта и перенос на следующую строку:

Описание шрифта	Перевод на новую строку
* 0, 4, название шрифта (строчными) высота, нижнее смещение, 0/2 (горизонтальный/ вертикальный), 0	* 10, 4, lf 002, перенос пера по Y, 001, 0

Пример программы сложной формы показан в Приложении Ж.

## 6.4. Компилирование программы

### AutoCAD до 13 версии

В главном меню AutoCAD выбираем пункт 7, AutoCAD запрашивает имя файла, который нужно откомпилировать, и дальше, если необходимо, выдает сведения об ошибках.

### AutoCAD, начиная с 13 версии

Компиляция происходит посредством команды compile. Рекомендуется при работе с одним и тем же файлом и повторной загрузке предварительно очистить память от предыдущей версии. Очистка памяти производится с помощью команды Purge.

## 6.5. Считывание и загрузка форм

Команда LOAD название файла производит загрузку формы.

## 6.6. Вставка формы

С помощью команды SHAPE вставляем форму. Если программа для шрифта, то команда STYLE установит созданный шрифт в ранг текущего.

**Закрепление.** Задание 3. Создание своих программ с формами. См. Приложение 3.

## 7. СТИЛИ ЛИНИЙ

Мы имеем возможность создавать свои типы линий.

### 7.1. Файл с текстом программы acad.lin

Для этого нужно в файле с расширением .lin записать описание линии, как это делается в стандартных библиотеках. Это обычные текстовые файлы, поэтому библиотеку AutoCAD можно восполнить с помощью обычного текстового редактора, но при условии, что выбранный текстовый редактор не должен добавлять в файл служебные символы.

### 7.2. Создание типа линий

Библиотека описаний типов линий может содержать неограниченное количество определений типов линий; каждое определение состоит из двух строк: заглавной строки и собственно строки определения.

\* имя [, описание ]

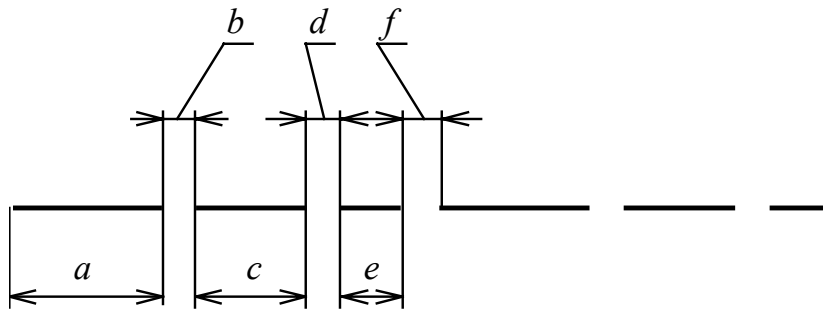
выравнивание, штрих1, штрих2, штрих3, ...

Имя типа линии будет выводиться на экран по команде LINETYPE ?. Описание не определяет тип линии, а является комментарием.

**Определением типа линий является вторая строка, которая использует следующие соглашения:**

- определение типа линии записывается в одну строку, в начале которой помещается буквенный код выравнивания, а затем идут разделенные запятыми действительные числа, задающие длину вычерчиваемых штрихов;
- если число больше нуля, то отображается штрих, длина которого равна указанному числу в условных единицах AutoCAD;
- если число меньше нуля, то делается пропуск, длина которого равна указанному числу в условных единицах AutoCAD. При выравнивании типа A первый штрих в определении шаблона не может быть пропуском;
- если число равно нулю, то отображается точка.

В качестве иллюстрации создадим новый тип линии и назовем его LINE.



тогда определение будет выглядеть так:

\*LINE , новый тип линии

A, a, -b, c, -d, e, -f

Рис. 5. Описание типа линии LINE

**Закрепление.** Задание 4. Создание стилей линии. См. Приложение И.

## 8. ШТРИХОВКИ

Создание новых типов штриховок аналогично созданию типов линий, но это - несколько более трудоёмкий процесс.

### 8.1. Файл с текстом программы acad.pat

Шаблоны штриховок хранятся в файлах с расширением .pat. Новый образец штриховки может быть как дописан в файл acad.pat, так и в отдельный файл. Имя файла должно соответствовать названию штриховки.

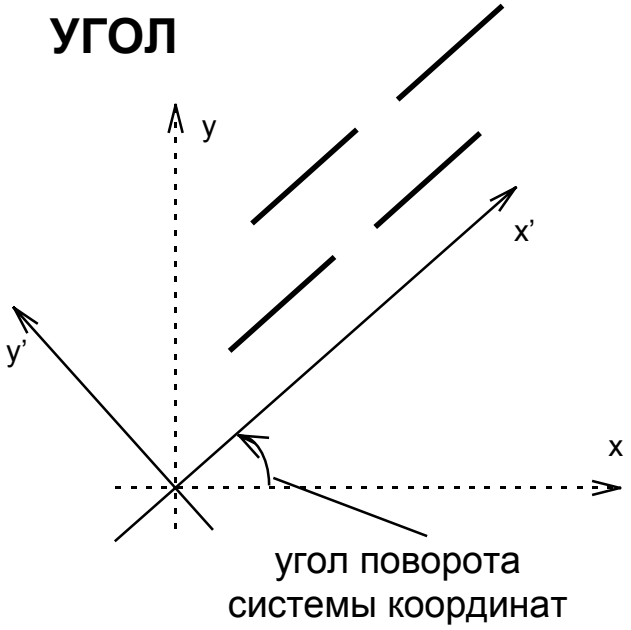
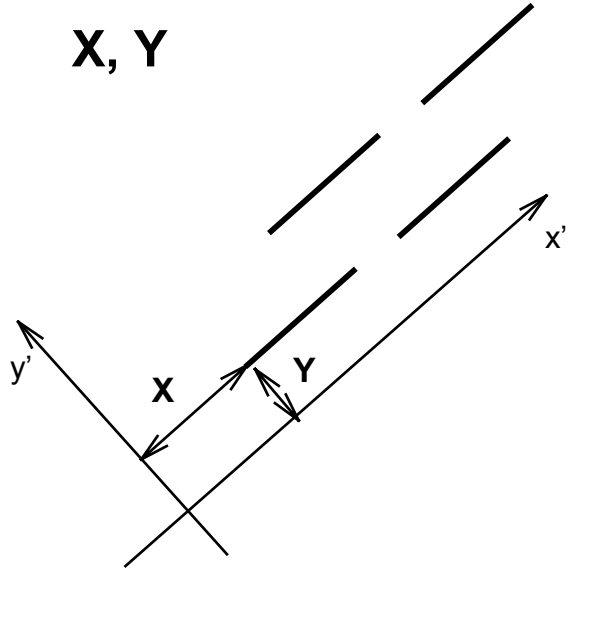
### 8.2. Описание штриховок

Описание штриховки состоит из заглавной строки и одной или нескольких строк определения:

\* имя [, описание ]

угол, X, Y, DX, DY, штрих1, штрих2, штрих3, . . .

угол, X, Y, DX, DY, штрих1, штрих2, штрих3, . . .

<b>УГОЛ</b>  <p>угол поворота системы координат</p>	<b>X, Y</b>  <p>X и Y - координаты точки приложения штриха в повернутой системе координат</p>
Угол поворота относительно текущей системы координат выбирается так, чтобы штрих располагался параллельно оси X	X и Y - координаты точки приложения штриха в повернутой системе координат

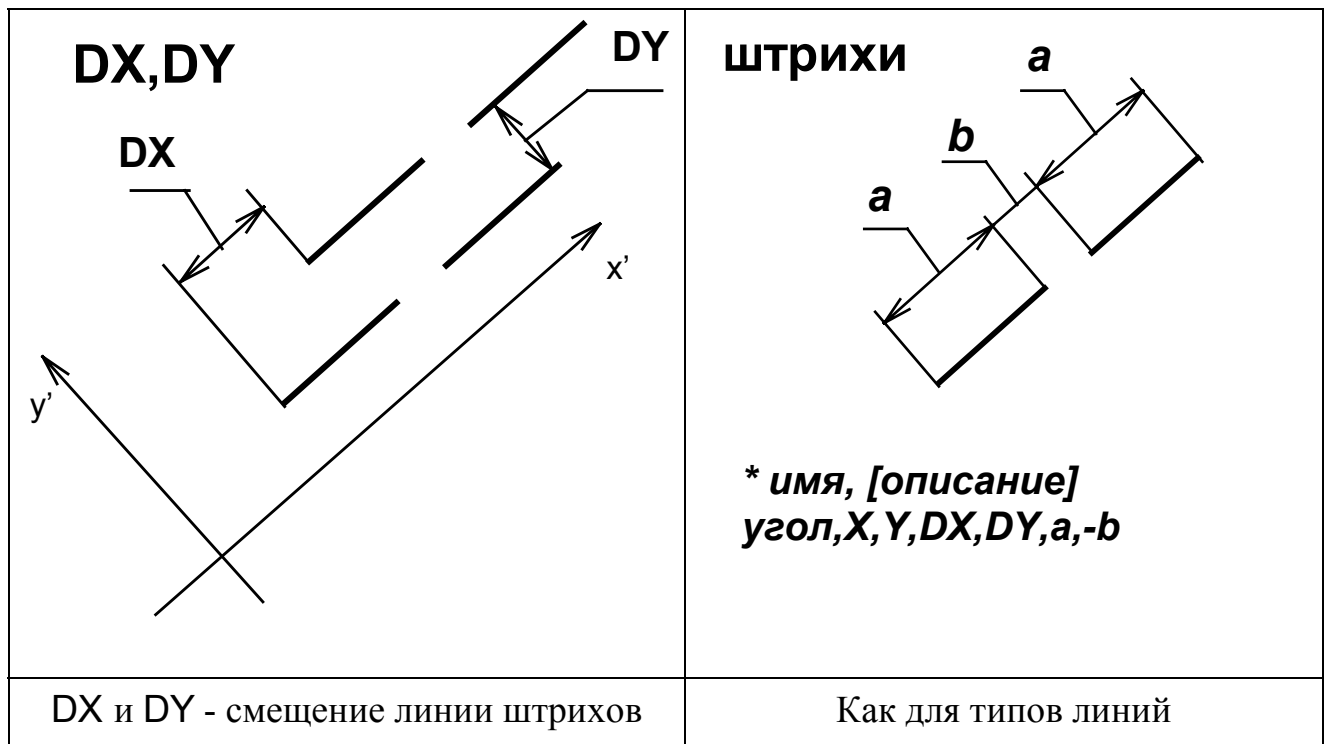


Рис. 6. Пояснения к описанию стиля штриховки

В качестве примера приведем описание стандартной штриховки GLASS.

\* GLASS

45, 0, 0, 2, 3, 1.8, -2.2

45, 0.3, 0.1, 2, 3, 1, -3

45, 0.25, 0.45, 2, 3, 1, -3

Таким образом, можно рассмотреть процесс создания штриховки подробнее:

Все линии штриховки - под 45 градусов, повторение “орнамента” - с периодом 2 по X и 3 по Y. Относительно штрихов: штрихи первой линии явно больше по длине, а штрихи второй и третьей линии одинаковы, отличаются друг от друга только точкой приложения.

**Закрепление.** Задание 5. Создание стиля штриховки. См. Приложение 3.

## 9. ПРОГРАММИРОВАНИЕ НА AUTOLISP

### 9.1. Синтаксис языка

Переменная - это объект, имеющий уникальное имя и способный хранить в себе некоторое значение.

#### Системные переменные

Все настройки графического редактора управляются так называемыми системными переменными AutoCAD. Когда мы изменяем установки, например, ORTHO (ОПТО), AutoCAD сохраняет только что установленный режим в соответствующей системной переменной (т.о., значение системных переменных изменяется в процессе работы с графическим редактором).

Системные переменные делятся на модифицируемые пользователем и защищенные. Выведите на экран список (SETVAR ?) переменных, и вы увидите, какие названия вы не можете использовать в своей программе.

#### 9.1.1. Переменные AutoLISP

Имя переменной может состоять из любых печатных символов (в том числе и цифр), но не должно включать в себя зарезервированные AutoLISP символы, так как они интерпретируются специальным образом.

Зарезервированные символы:	. ' " ; ( ) или пробел
Знаки операций AutoLISP:	- * = > < + - /
“Плохие” символы:	? ' ! \

AutoLISP не различает большие и малые буквы. Кроме того, при создании новых переменных просьба не увлекаться и не вводить названия, длина которых выходит за пределы шести символов (имя, длина которого больше, требует для хранения большой объем памяти). Также не рекомендуется начинать имя переменной с цифры.



### 9.1.2. Типы данных AutoLISP

- Строковые переменные - совокупности букв и констант, заключенных в кавычки.
- Целые переменные - положительные или отрицательные целые числа (без дробей и десятичной точки). Целые числа представлены в машине двумя байтами и поэтому не могут выходить за диапазон (-32 768, 32 767).
- Действительные переменные - положительные или отрицательные числа с десятичной точкой. Особенность: если значение меньше 1, то нужно явно указывать 0 перед десятичной точкой, иначе будет выдаваться сообщение об ошибке.

Переменные любого из простых типов называют **атомами**.

**Списком** называется набор разделенных пробелами атомов и/или списков, заключенных в круглые скобки.

#### Перечислим некоторые типы данных AutoLISP:

INT	Целые величины
REAL	Числа с плавающей точкой
STR	Строковые константы
FILE	Дескрипторы файлов
SYM	Символы
LIST	Списки и функции пользователя
SUBR	Внутренние функции AutoLISP
.....	и так далее ...

### 9.1.3. Выражения AutoLISP

Список, первым элементом которого является функция, называется **выражением**.

#### Свойства выражений

- Каждая открывающая круглая скобка должна иметь закрывающую.

- Сразу после открывающей круглой скобки должен идти идентификатор операции (функции), выполняемой при вычислении выражения (имя функции).

- Следующие за именем функции аргументы функции должны быть отделены от имени функции и друг от друга по крайней мере одним пробелом (дополнительные пробелы и переводы строк игнорируются, так что выражение **AutoLISP** может занимать несколько строк, что в действительности и происходит).

- Каждое выражение вычисляется (выполняется) и результат возвращается. Результатом может быть нуль (**nil**) или результат вычисления последнего подвыражения.

- С логической точки зрения любое возвращаемое значение либо истинно, либо ложно. Если значение выражения вычислено быть не может и возвращается нуль, то оно считается ложным. Если выражение вычисляется, то оно считается истинным - не-нуль (**non-nil**).

Итак, выражение **AutoLISP** имеет вид:

(функция аргумент1 аргумент2 ... аргументN)

Функция - имя операции (в том числе и арифметической), которая должна быть выполнена. Число аргументов может быть больше 2.

Произведение трех чисел: (\* 2 3 4)

Вложенные выражения: (\* 4.4 (- 3.3 (+ 2.2 1.1 )))

Выражение анализируется **AutoLISP** слева направо, пока не встретится скобка. Если встречается закрывающая скобка, то завершается анализ выражения, выполняется функция и вычисленное значение передается на более старший уровень вложенности или в **AutoCAD**. Если же встречается открывающаяся скобка, **AutoLISP** переходит к анализу выражения более младшего уровня вложенности и, пока не завершит его анализ, не перейдет к дальнейшему анализу выражения предыдущего уровня. Предел вложенности выражений - 100.

## Функции присвоения

Используются две функции. В переменной может храниться не только значение, но и целое выражение, а точнее, его адрес. Функция SETQ позволяет нам об этом не думать и считать, что мы обращаемся прямо к значению переменной. Функция SET считает, что мы обращаемся к самой переменной. Апостроф перед переменной означает значение переменной.

```
( set 'A 5.0 )
```

Возвращает 5.0 и присваивает это значение A.

```
( set 'B '(A +C))
```

Возвращает выражение (A+C) и присваивает его B.

Итак, в переменной хранится выражение, причем именно выражение, а не текстовая константа - изменяя значения переменных A и C мы меняем значение B. Эта запись эквивалентна записи

```
( setq B '(A +C))
```

Если две переменные вложены друг в друга, то к младшей можно обращаться через старшую, косвенно:

```
( set 'B 'A)
```

Устанавливает A в B и возвращает A.

```
( set B 640)
```

Устанавливает A в 640 и возвращает 640.

## 9.2. Обзор функций AutoLISP

### 9.2.1. Математические функции

(+ число1 число2 ...) Возвращает сумму всех аргументов.

(- число1 число2 ...) Вычитает число1 из числа2, если более двух аргументов, то из первого аргумента вычитается сумма всех остальных, если задан один аргумент, то он вычитается из нуля (инвертируется его знак).

(\* число1 число2 ...) Возвращает произведение всех чисел.

(/ число1 число2 ...) Делит число1 на число2, а если аргументов более двух, то первое число делится на произведение всех остальных.

(1+ число) Увеличение целого или действительного аргумента на единицу.

(1- число) Уменьшение целого или действительного аргумента на единицу.

(atan число1 [число2]) Если не задано число2, то возвращает арктангенс переменной число1 в радианах, область допустимых значений -  $[-\pi, \pi]$  радиан. Если заданы оба числа, то возвращает арктангенс переменной число1/число2 в радианах. Если число2 - нуль, то в зависимости от знака переменной число1 возвращается + или - 1.570796 радиан ( $+90^\circ$  или  $-90^\circ$ ).

(abs число) Вычисление абсолютного значения действительного или целого числа.

(cos число) Возвращает значение косинуса угла, заданного аргументом в радианах.

(exp степень) Вычисляет значение экспоненциальной функции с основанием  $e$  и аргументом, равным степени.

(expt основание степень) Вычисляет значение экспоненциальной функции с указанным основанием и степенью.

(gcd число1 число2 ...) Возвращает наибольший общий делитель.

(log число) Натуральный логарифм аргумента.

(max число1 число2 ...) Наибольший аргумент.

(min число1 число2 ...) Наименьший аргумент.

(rem число1 число2 ...) Остаток от деления переменной число1 на переменную число2.

(sin число) Возвращает значение синуса угла.

(sqrt число) Извлекает квадратный корень из аргумента.

### **9.2.2. Логические функции AutoLISP**

Логический оператор - это функция, сравнивающая между собой два или более аргументов. Результат может быть либо истина (non-nil), либо ложь (nil).

(and выражение1 выражение2 ) Возвращает результат выполнения логического И над списком выражений.

(not аргумент ) Возвращает результат выполнения логического НЕ над своим аргументом.

(or выражение1 выражение2 ) Возвращает результат выполнения логического И над списком выражений.

### **9.2.3. Использование функций GET для ввода данных**

(getangle точка "текст запроса-подсказки") Возвращает угол между заданным пользователем вектором и положительным направлением оси X. Всегда в радианах.

(getcorner точка "текст запроса-подсказки") Возвращает координаты указанной пользователем точки.

(getdist точка "текст запроса-подсказки") При любых текущих единицах измерения эта функция всегда возвращает действительное число.

(getenv имя переменной) Возвращает строковое значение, присвоенное переменной среды DOS.

(getint "текст запроса-подсказки") Ввод целого числа. Только с клавиатуры.

(getorient точка “текст запроса-подсказки”) То же самое, что и getangle, но измерение угла происходит относительно текущего направления измерения углов.

(getpoint точка “текст запроса-подсказки”) Позволяет ввести точку.

(getreal “текст запроса-подсказки”) Позволяет вводить действительное число. Только с клавиатуры.

(getstring флаг пробела “текст запроса-подсказки”) Запрашивает ввод текстовой константы. Если флаг пробела указан и не равен нулю, то строковая константа может содержать пробелы, и завершением ввода считается нажатие клавиши ENTER, в противном случае строка не может содержать пробелы и клавиша ПРОБЕЛ работает как терминатор ввода.

(getpvar имя переменной) Возвращает значение системной переменной AutoCAD.

#### **9.2.4. Вывод данных в командную строку**

(prin1 выражение дескриптор\_файла) Выражение выводится на экран и возвращается в AutoLISP. Если указан дескриптор файла и файл открыт, то запись идет сразу на два устройства: на экран и в файл. Печатается только указанное выражение; переход на новую строку не осуществляется, и никакие пробелы не печатаются.

(princ выражение дескриптор\_файла) То же, что и prin1, но управляющие символы не расшифровываются, а выводятся на экран.

(print выражение дескриптор\_файла) То же, что и prin1, но перед печатью осуществляется переход на другую строку.

(prompt строка\_символов) Выводит сообщение в командную строку и возвращает nil.

### 9.2.5. Условное ветвление программ

Каждая программа имеет свою логическую структуру. Ветвление - это способ управления ходом выполнения программы. Условные операторы - средство управления ветвлением программ.

(cond ( тест1 результат1 ... ) ...) Воспринимает в качестве аргументов любое число списков. Оценивает по очереди первые элементы списков, пока не встретится элемент, отличный от nil. Затем вычисляется то выражение, которое следует за тестом, и возвращается значение последнего выражения в субсписке. Если в субсписке только одно выражение (например, результат отсутствует), то возвращается значение выражения тест.

Как видно из примера, в котором функция COND возвращает 0 или 1 в зависимости от введенной пользователем строковой переменной s, эта функция может использоваться в качестве условного оператора переключения (“case” во многих языках программирования) :

```
(cond ((= s "Y") 1)
      ((= s "y") 1)
      ((= s "Д") 1)
      ((= s "д") 1)
      ((= s "N") 0)
      ((= s "n") 0)
      ((= s "H") 0)
      ((= s "h") 0)
)
(if тест-выражение выражение-тогда выражение-иначе )
```

Иногда по условию требуется выполнить не одно, а несколько выражений. Для этого используют функцию PROG. Последовательность выражений, объединенных функцией PROG, считается одним выражением.

```
(if ( = a b) ( progn
     setq a ( + a 10)
     setq b ( - b 10)
    )
)
```

### 9.2.6. Организация циклов

(repeat число выражение1 выражение2 ... ) Функция повторяет операторы указанное число раз.

```
(while тест-выражение выражение1 выражение2 ... )
```

Выход из цикла осуществляется по условию.

(foreach имя список выражение) Эта функция, проходя по списку, присваивает каждому элементу имя и вычисляет каждое выражение для каждого элемента списка.

```
( foreach n '( a b c ) ( print n ))
эквивалентно ( print a ) ( print b )
               ( print c )
```

### 9.2.7. Вызов команд AutoCAD из программы на AutoLISP

В этом нам поможет функция

```
( command аргумент1 аргумент2 ... )
```

Аргументы - это команды AutoCAD и их опции. Если в списке встречается ключевое слово Pause, то функция COMMAND приостановит свое действие.

### 9.2.8. Файловый ввод/вывод

( open имяфайла режим ) Открыть файл - значит подготовить дескриптор файла к использованию его функциями AutoLISP. Поэтому возвращаемое функцией open значение должно присваиваться некоторой символьной переменной.



( `setq a ( open "file.ext" "r" )` ) Здесь `a` - дескриптор файла `file.ext`, открытого для чтения.

Допустимые значения флага чтения/записи :

Режим <code>open</code>	Описание
<code>r</code>	Открыть файл для чтения. Если нет такого файла, то <code>nil</code> .
<code>W</code>	Открыть файл для записи. При необходимости создается новый файл.
<code>A</code>	Открыть файл для добавления данных. При необходимости он создается.

Имя может включать в себя путь ( имена надкаталогов ).

( `close дескриптор файла` ) Закрытие файла.

( `read строка` ) Эта функция возвращает первый список или атом из данной строки, причем строка не должна содержать пробелы.

( `read-char дескриптор файла` ) Считывает единичный символ из буфера клавиатуры или из открытого файла. Возвращает целое число - код ASCII считанного символа. Если дескриптор файла не задан и буфер клавиатуры пуст, функция ожидает ввода символа ( пользователь должен ввести что-либо с клавиатуры, заканчивая нажатием `ENTER` ).

( `read-line дескриптор файла` ) Считывает строку символов с клавиатуры или из открытого файла. Возвращается считываемая строка.

( `write-char число дескриптор файла` ) Записывает один символ на экран или в открытый файл. Здесь число - код ASCII символа - является возвращаемым функцией значением.

( `write-line строка дескриптор файла` ) Записывает строковую константу строка на экран или в открытый файл. Возвращает строку, взятую в кавычки, и опускает кавычки при записи в файл.

Пример простейших подпрограмм представлен в Приложении Л.

### 9.3. Последовательность действий при создании параметрического чертежа

Формат А4 должен быть распланирован таким образом:

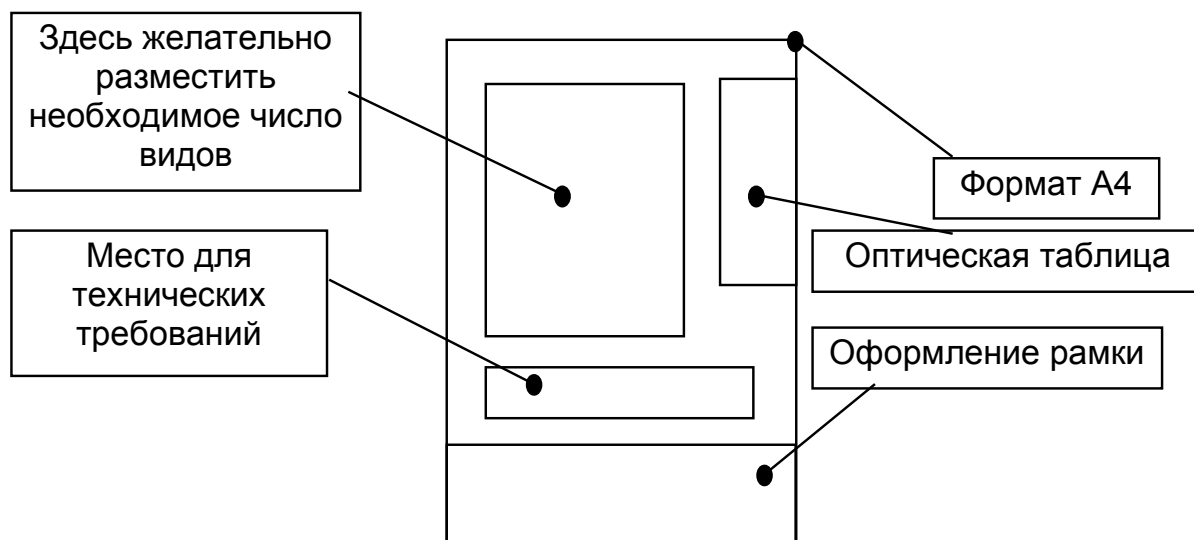


Рис. 7. Размещение основных элементов на параметрическом чертеже

#### 9.3.1. Вставка рамки

Вставить формат следует в самом начале программы, чтобы понимать, как располагается наша деталь на листе. Вставляемый объект должен быть максимально загружен, то есть на формате должно присутствовать все, кроме элементов, которые связаны непосредственно с конкретной деталью. Нужно создать чертеж, начертить формат рамочки, оптическую таблицу и сохранить чертеж как блок.

Далее в программе можно вставить наш блок с помощью строки:

```
(command "insert" "A4" "0,0" "" "" "")
```

#### 9.3.2. Ввод основных параметров

Для корректной загрузки параметров можно создать файл, из которого будут вводиться данные. В нем следует расписать переменные, значения которых нам заранее известны.

```
(setq ff (open "shmidt.ini" "r"))
```

```
(while (setq line_ (read line))
```

Если файл является позиционным, то извлекаем информацию через функцию `SUBSTR` и преобразование переменных из строкового в вещественный и целый тип, в противном случае - через функцию `READ` и поиск пробелов.

### **9.3.3. Опрос пользователя по начальным установкам**

Основная команда для загрузки параметров чертежа, если они не загружены из текстового файла:

```
(setq имя (getreal "подсказка"))
```

### **9.3.4. Вычерчивание**

Формирование контура детали производится с помощью обычных команд вычерчивания примитивов (`LINE`, `ARC` и т.д.).

Для простановки размеров лучше всего воспользоваться режимом `DIM`.

Простановка технических требований состоит из выбора набора технических требований и вывода их на экран посредством команд `Dtext`, `Mtext`.

**Закрепление.** Задание 6. Создание программы выполнения параметрического чертежа. См. Приложение М.

## **10. ОСОБЕННОСТИ РАБОТЫ В СРЕДЕ AUTOCAD 14 ВЕРСИИ**

### **10.1. Введение. Работа с чертежом. Создание, сохранение**

При входе в AutoCAD автоматически создается новый чертеж или (в английской версии) Вы можете выбрать один из ранее созданных. Когда создан новый чертеж, необходимо его сразу же сохранить в своем рабочем каталоге, то есть там, где хранятся все файлы Ваших чертежей.

#### **10.1.1. Сохранение**

- В меню **File** (файл) выбирается пункт **Save As...** (сохранить как...). Появляется окно сохранения.

- В строке **Save in** (Папка) производится выбор каталога, в котором необходимо сохранить файл.

- Затем в строке **Name** (имя файла) вписывается имя создаваемого файла.

- **ОК.**

При работе с чертежом постоянно требуются различные панели инструментов. Необходимо выбрать стратегию работы: или все время держать все панели на экране (это требует большого пространства), или выводить требуемые панели на экран по мере необходимости.

#### **10.1.2. Вывод панелей инструментов на экран**

- В меню **View** (вид) нужно выбрать пункт **Toolbars** (панели инструментов). Появляется окно выбора панелей.

- Щелчком мыши в квадрате около панели производится ее выбор. Тогда панель выводится на экран, и окно выбора можно закрыть.

- За заголовок панели или за свободное поле между кнопками производится ее перемещение.

## 10.2. Режимы работы в среде AutoCAD

**Текстовый** - для просмотра последних диалогов с AutoCAD. Включается клавишей F2. Затем можно или перейти обратно на поле чертежа (не закрывая окна), или закрыть текстовое окно, тогда переход произойдет автоматически.

**Ортогональный** (вычерчивание ортогональных линий) вкл./выкл. клавишей F8.

**Сетка информационная** - видимые точки, расположенные по координатам AutoCAD на заданном расстоянии.

- Вкл/выкл. видимости сетки - клавишей F7.
- Изменение шага сетки (размера ячейки) обеспечивает команда **grid** (с клавиатуры). Эта сетка носит информационную нагрузку, помогает ориентироваться в размерах.

**Сетка привязки курсора мышки.**

- Вкл/выкл. режима F9. При включенном режиме происходит выбор точек, расположенных в узлах сетки.
- Изменение шага сетки производится командой **snap** (с клавиатуры). Можно ввести нужный размер ячейки привязочной сетки.

Для удобства работы рекомендуется устанавливать шаг по информационной сетке в 5 раз большим, чем по привязочной - например, 1 и 0.2.

## 10.3. Свойства объекта

Изменение свойств объекта - это изменение слоя, которому он принадлежит, цвета, положения, толщины по оси Z (в перпендикулярном плоскости чертежа направлении), размера и др.

Изменение свойств объекта: меню **Modify** (изменить) - **Properties** (свойства), затем выбрать требуемые объекты.

#### **10.4. Увеличение чертежа и его частей**

Если элементы чертежа вышли за пределы рабочей области, можно набрать команду **Zoom** и выбрать опцию **All**.

Выбор опций в любой команде **AutoCAD** производится при наборе на клавиатуре того сочетания букв, которое в названии опции выделено заглавными буквами.

Существует множество видов увеличений. Наиболее используемые -

окном  и реальное увеличение 

#### **10.5. Установка привязок к точкам чертежа**

Постоянная привязка устанавливается в меню **Tools** (инструменты) пунктом **Object snap** (захват объекта). Там необходимо выбрать нужные типы привязок (обычно **Intersection**, **Perpendicular**, **Nearest**, **Endpoint** (пересечение, перпендикуляр, ближайшая, конечная)).

Переход от привязки к привязке во время работы производится с помощью клавиши **TAB**.

Для эффективной работы привязки лучше устанавливать в начале работы над чертежом или даже в файле-прототипе.

#### **10.6. Особенности модифицирования элементов чертежа**

Многие команды модифицирования требуют выбора объектов. Это значит, что нужно указать на чертеже мышкой требуемые объекты и нажать **Enter** для завершения списка. Это не относится к команде **pedit**.

Для обрезания лишних отрезков используется команда **trim**, для продления до границы **extend**.

Для откладывания расстояний на чертеже очень удобно пользоваться командой **offset**, которая дает возможность скопировать отрезок, линию на заданное расстояние в заданную сторону.

Для не очень точных операций можно пользоваться **grips** (ручками). Это синие (обычно) квадраты малого размера, появляющиеся на выделенном объекте при отсутствии команды. Если навестись на ручку, то можно изменить её положение и, следовательно, положение точки объекта.

Следует очень осторожно пользоваться ручками. Для соединения линий и дуг в полилинии, для изменения толщины полилиний используется команда редактирования полилиний **Pedit** на панели **Modify II** (модифицировать II). Опция **Join** - для объединения элементов в одну полилинию, опция **Width** - толщина полилинии.


Для разъединения элементов полилинии, а также блоков используется команда **Explode** на панели **Modify** (модифицировать).

## 10.7. Работа с текстом

Вывод текста производится с помощью команд **Dtext** и **Mtext**. Положение текста можно изменять с помощью стандартной команды переноса или с помощью ручек.

Для успешной работы следует определить стили текста заранее. Для каждого начертания (размера, шрифта) лучше создать свой стиль текста.

Для изменения текста (то есть его содержания) нужно вывести на экран дополнительную панель **Bonus text tools** из группы **Bonus**. Нужно нажать

кнопку **Find and replace text** . Далее появляется окно, в котором есть две строки: первая - что найти; вторая - на что заменить; ниже - флажки :

- **Match Case** - регистр (обращать внимание на регистр или нет),

- **Global change** - глобальная замена (заменить везде, где встретится данное буквосочетание).

Дальше, после нажатия **OK**, нужно выбрать те текстовые надписи, в которых Вам нужно заменить текст.

После нажатия **Enter** появляется окно с несколькими кнопками (если не установлен **Global change**). Тогда:

- **Replace** (заменить) - заменить найденный элемент,
- **Auto** (автоматически) - заменить все фрагменты,
- **Skip** (пропустить) - пропустить этот элемент,
- **Cancel** (отмена) - отмена всей команды.

Лучше пользоваться глобальной заменой, так как в процессе работы команды трудно понять, какие из элементов пропускать, а какие заменять.

## **10.8. Слои чертежа**

Для каждого плана чертежа (черновик, чистовик, электрическая схема, вентиляционная схема) лучше создать свой слой с понятным названием, соответствующим назначению).

Создание слоя: меню **Format** (формат) - **Layer** (слой). Дальше - **New** (новый).

Перенос объектов с одного слоя на другой - в свойствах данного объекта.

## **10.9. Размеры, их простановка и использование**

Для простановки размеров необходимо заранее создать стили размеров. Главное при создании стилей размеров - в пункте **Annotation** (аннотация) - **Unit** указать масштаб, используемый в чертеже.

Простановка размеров - с помощью кнопок на панели **Dimension**.



Изменение размеров: редактирование положения текста после нанесения размеров, а также изменение точек приложения размеров производится с помощью ручек (перетаскивание ручки в требуемое положение).

## **10.10. Блоки**

### **10.10.1. Создание блока**

В текущем чертеже блоки создаются для многократного использования одинакового набора примитивов. Библиотеки блоков создаются для использования блока во многих чертежах.

#### **Указание пути к библиотеке**

Меню Tools (инструменты) - Preferences... (соглашения) - Support File Search Path (путь к файлу поддержки) - кнопка Add (добавить). Далее есть два варианта: или написать путь в месте, указанном курсором, или, нажав на кнопку Browse (поиск), указать нужный каталог в окошке выбора каталога.

#### **Создание блока в библиотеке**

Команда Wblock - нужно выбрать каталог, где находится Ваша библиотека. Далее - ввести имя файла-блока. Enter.

На вопрос Block name - Enter (нужно согласиться с предложенным именем, которое будет таким же, как имя файла).

Далее - точка вставки (точка, относительно которой будет производиться вставка всех элементов блока), после этого выбираете те примитивы, которые должны входить в этот блок.

### **10.10.2. Вставка блока**

Вставка блока производится с помощью команды Insert. Требуется ввести имя блока, точку вставки, масштабы и угол поворота.

## **10.11.Трехмерное моделирование**

### **10.11.1. Пространство и координаты**

Выбор расположения видов и их количества: в меню **View (вид) - Tiled viewports - Layout**, и далее производится выбор того вида экрана, который необходим Вам для моделирования.

Для вычерчивания двумерных фигур не на стандартной плоскости X и Y нужно создать пользовательские системы координат. Используйте панель инструментов **USC (ПСК)**. Задание системы координат по трем точкам - самое простое, когда имеется трехмерный объект, на сторонах которого будут начерчены двумерные объекты. Три точки определяют: 1 - точка начала координат, 2 - положительное направление оси X, 3 - положительное направление оси Y.

### **10.11.2. 3D-объекты: тела, поверхности**

#### **Типы трехмерных объектов**

- Тела (заполненные) можно получить только из замкнутых двумерных фигур (прямоугольник, многоугольник, замкнутая полилиния и др.).
- Поверхности - из любых фигур.

### **10.11.3. Способы получения 3D объектов**

- Вращение. Должна быть задана ось (линия), относительно которой вращать объект.
- Вытягивание - нужно задать высоту и угол наклона образующих.
- Изменение толщины объекта - в его свойствах.

#### **10.11.4. Модифицирование 3D - объектов**

Перенос, удаление, копирование - с помощью стандартных команд модифицирования двумерных примитивов. Только точки задаются по привязкам в разных видовых окнах или на изометрическом виде.

Отражение, поворот, массив - меню **Modify** (изменить, модифицировать) - **3D operations** (3D операции). Там находятся эти три команды для трехмерных объектов. Главное - правильно выбрать плоскость или ось, относительно которой произойдет вращение или отражение.

Соединение объектов, исключение объектов и сложение. Эти команды - на панели **Modify II** (модифицировать II).

#### **10.11.5. Операции над объектами в двухмерном сечении**

Генерация **Section** (сечений) и **Slice** (разрезов). Сечения и разрезы производятся с помощью плоскости разреза. Плоскость может быть выбрана из списка предложенных и может быть задана пользователем по трем указанным точкам. Главное отличие команд сечение и разрез - в том, что сечение физически не разрезает фигуру. Просто создается плоская фигура, отображающая контуры объекта в данном сечении. Разрез же физически режет фигуру заданной плоскостью.

### **10.12. Принтеры. Плоттеры**

Настройка принтера/плоттера. В пункте меню **File** (файл) **Printer setup** (настройки принтера) можно выбрать новый или уже установленный принтер/плоттер и, нажав кнопку **Modify**, переконфигурировать его заново.

Проще производить установки в окне при непосредственной печати чертежа. Окно печати содержит все пункты конфигурирования, только в более наглядном виде.

### 10.12.1. Вывод чертежей на принтер/плоттер

Меню File (файл) Print (печать). Грамотное заполнение окна печати приводит к правильному результату.

#### Пункты:

- **Device and default information** (устройство печати) - здесь можно выбрать различные устройства для печати, установленные/присоединенные к Вашей машине.

- **Pen parameters** (параметры перьев) - можно поменять цвет или установить, чтобы красный выводился на принтер как зеленый, черный и т.д. Здесь же можно изменить оптимизацию - наиболее оптимальный режим вычерчивания, а также задать толщину пера для определенного цвета.

- **Additional parameters** (дополнительные параметры): справа - что печатать (окно, весь чертеж без масштабирования, все вставить в область вычерчивания); слева - некоторые параметры печати (отображение текста, печать скрытых линий, толщина пера, печать в файл).

- **Paper size and orientation** (размер бумаги и ориентация) - выбор размера обычно не требуется, так как установлен максимальный. Здесь же располагается переключатель единиц чертежа - дюймы/мм.

- **Scale, rotation and origin** (масштаб, поворот и смещение).

- **Plot preview** (вид печати). Это - проверка правильности ваших действий. Здесь можно посмотреть, как будет выглядеть Ваш чертеж при печати.

- Нажатие **ОК** запустит на печать Ваш чертеж.

### **10.13. Создание чертежа-прототипа**

Используется для применения стандартных настроек в новых чертежах.

Обычно создается пользователем.

**В этом файле производятся следующие настройки:**

- стили размеров,
- стили текста,
- указание привязок,
- нужное расположение видов (если требуется) и др.

Затем файл сохраняется как обычный чертеж. При создании нового чертежа Вы открываете этот и записываете его под новым именем, тем самым используя все настройки чертежа-прототипа в новом файле.

### **10.14. Работа со ссылками на чертежи и картинки. Работа с AutoCAD в сети**

#### **10.14.1. Применение ссылок**

- Просмотр другого чертежа, без выхода из текущего.
- Использование части чертежа в качестве примера.
- Использование для просмотра другого чертежа, с целью согласования в проекте чертежей, созданных разными пользователями.

#### **10.14.2. Установка ссылок**

Ссылки устанавливаются в панели инструментов Reference (ссылки).

Можно ссылаться на файлы:

- картинки,
- чертежи,
- части чертежа,
- стили размеров и текстов в ссылке.

### **10.14.3. Работа в сети**

При работе в сети можно ссылаться на файл, расположенный на видимом пространстве сетевого диска. При этом при обновлении чертежа он будет при новой загрузке обновляться в файле, в котором использована ссылка. Ссылки легко выводятся на печать.

# ПРИЛОЖЕНИЯ

## Приложение А. Способы построения окружности

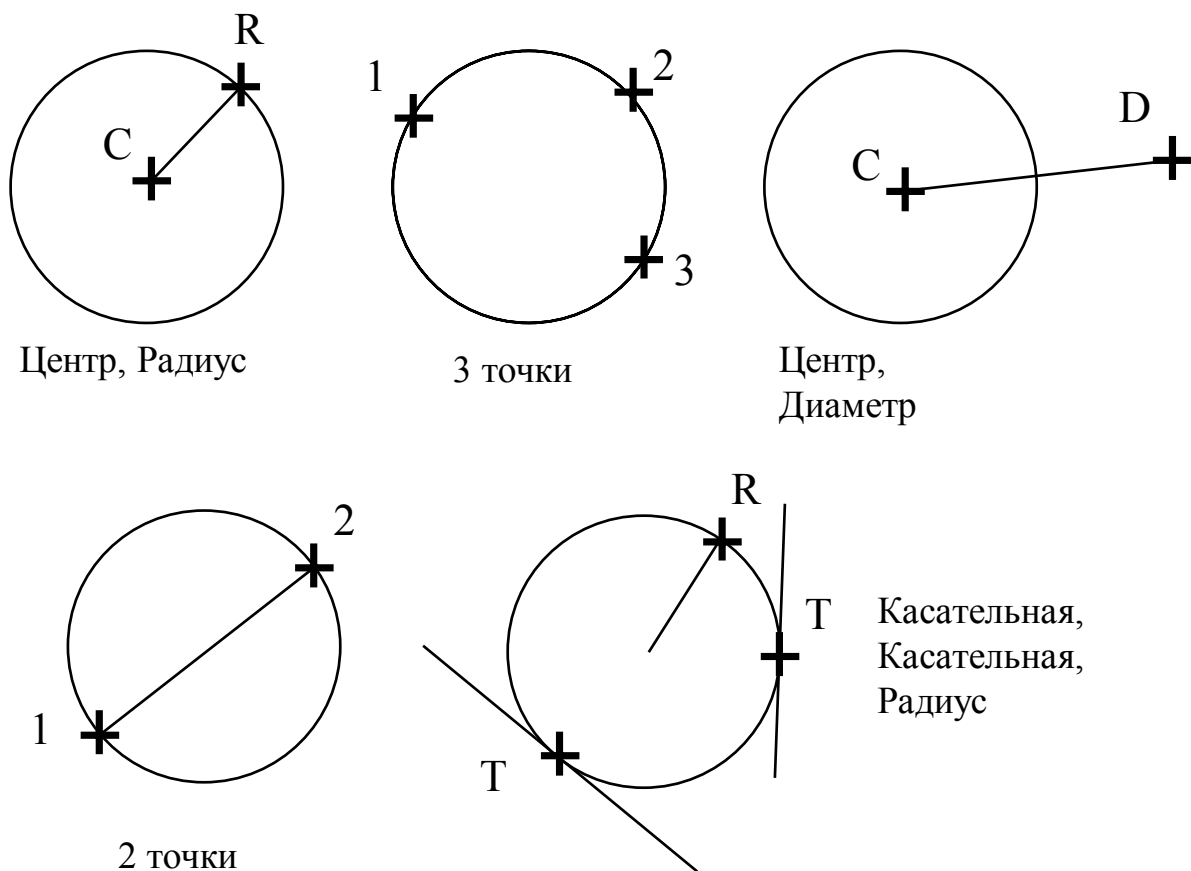


Рис. 8. Выполнение окружностей различными способами

## Приложение Б. Способы построения дуги

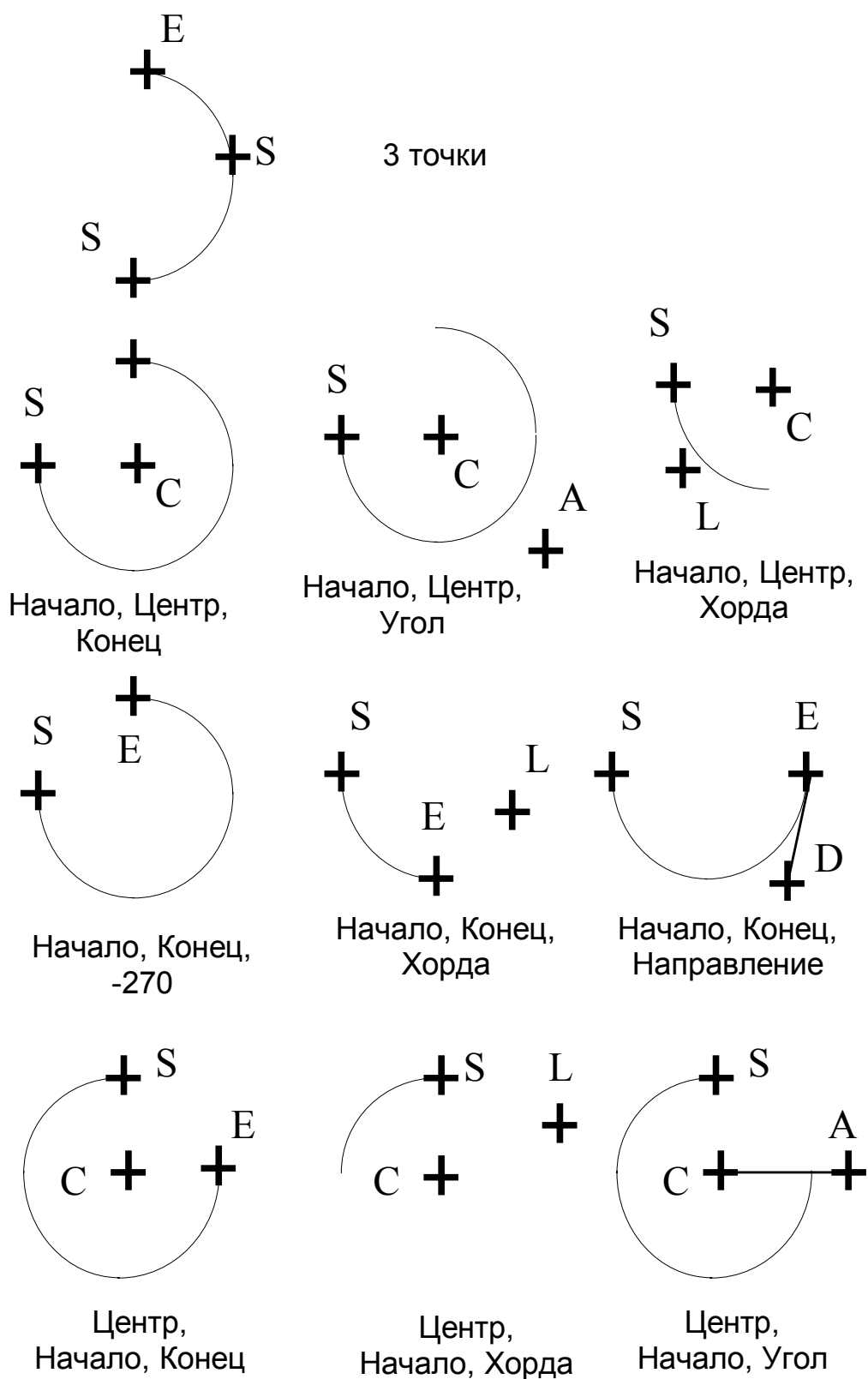


Рис. 9. Выполнение дуг различными способами



## Приложение В. Задание 1. Выполнение чертежа линзы

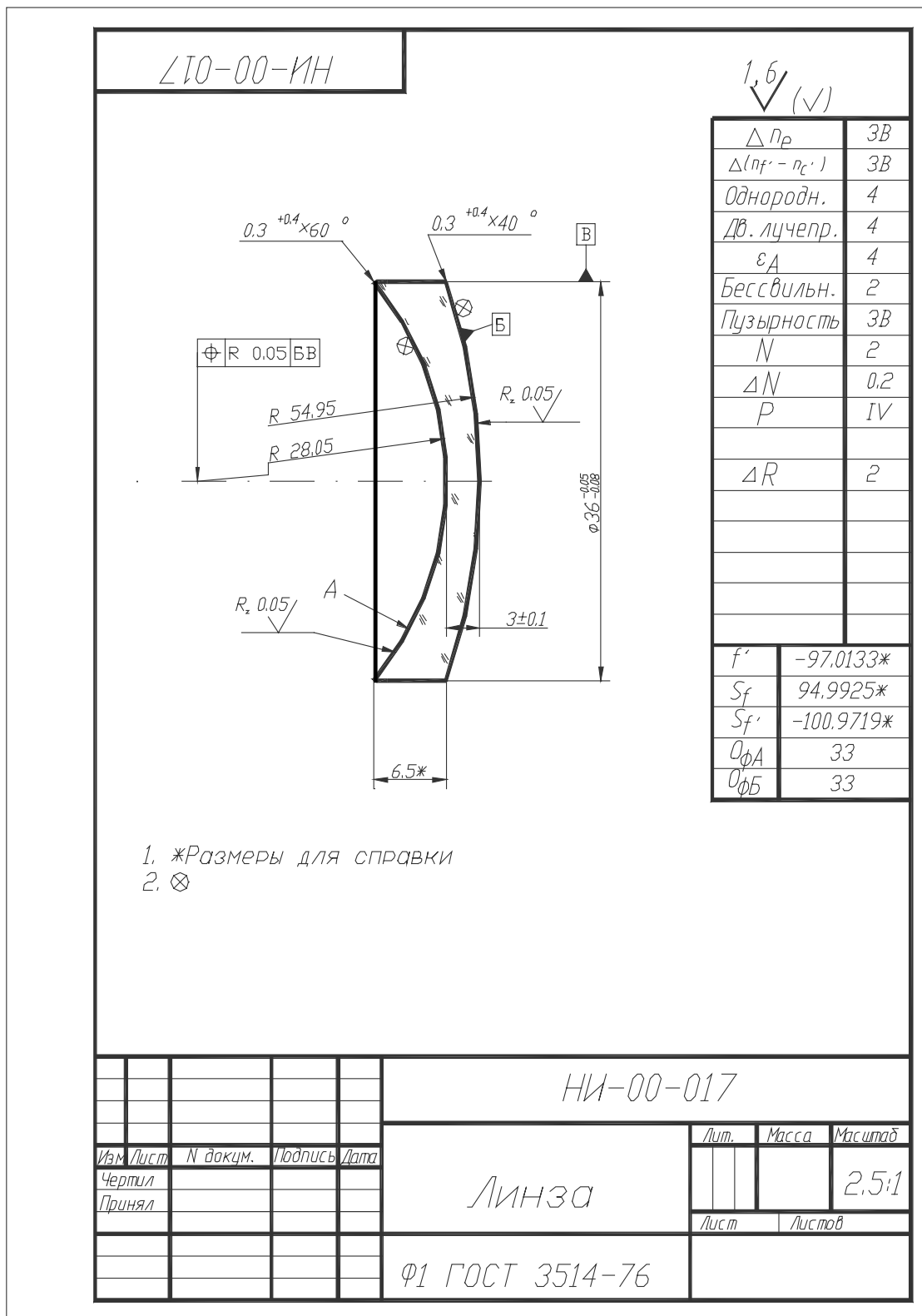


Рис. 10. Пример выполнения чертежа

Выполнение чертежа линзы и простановка размеров без использования специальных возможностей AutoCAD. Данные для чертежа расположены в Таблице А.

С правилами выполнения чертежей оптических деталей можно ознакомиться в ГОСТ 2.412-81 «Правила выполнения чертежей и схем оптических изделий».

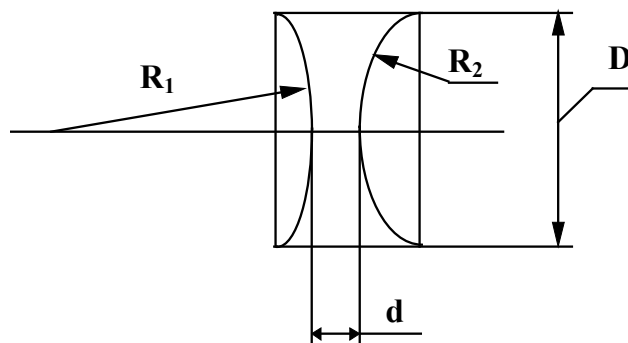


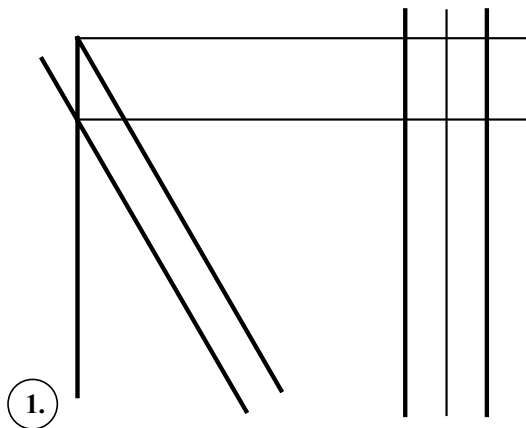
Рис. 11. Иллюстрация для задания 1

Таблица А. Данные для задания 1 по вариантам

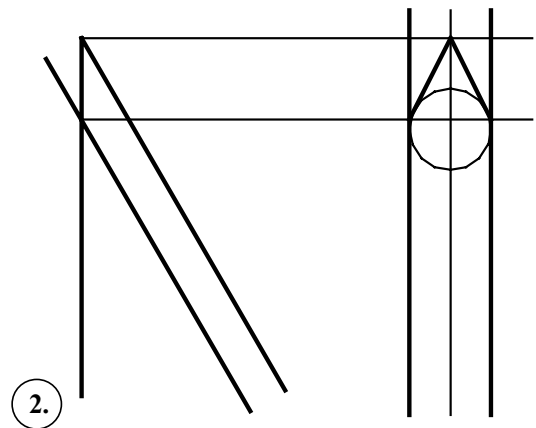
№	$R_1$	$R_2$	$D$	$d$	ВИД
1.	100	50	40	10	
2.	-100	150	60	20	
3.	80	100	40	20	
4.	-30	-100	30	5	
5.	120	-120	100	40	
6.	160	$\infty$	40	10	
7.	-130	-80	40	40	
8.	60	100	50	10	
9.	-60	100	60	8	
10.	50	40	30	5	
11.	-80	-70	30	10	
12.	60	-75	30	10	
13.	110	-85	50	10	
14.	-55	40	20	5	

## Приложение Г. Задание 2. Выполнение эскиза чертежа призмы с ходом лучей

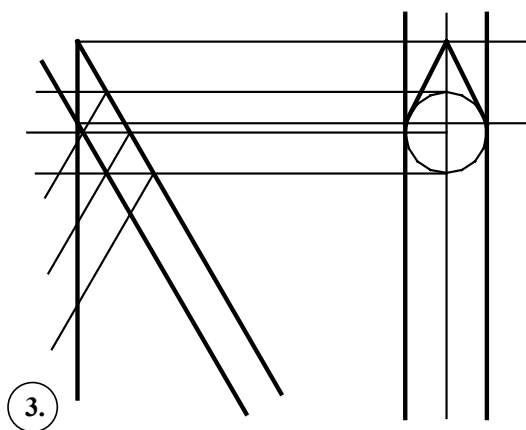
Выполнить эскиз чертежа призмы. Расположить размеры, ход лучей, линии построения и основной вид призмы на разных слоях.



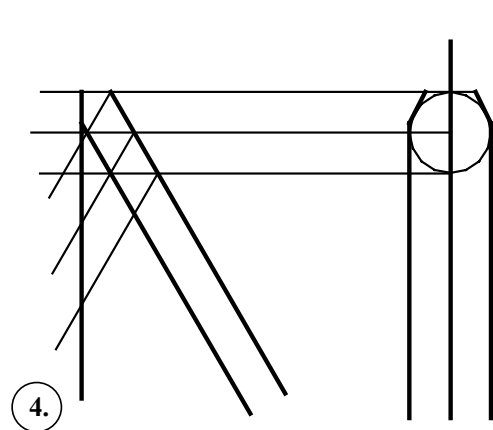
1. Отображение угла призмы при крыше и. Проецирование крыши на вид слева



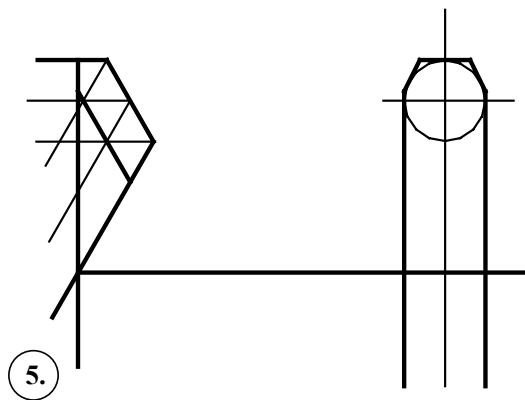
2. Отображение проекции хода лучей на виде слева



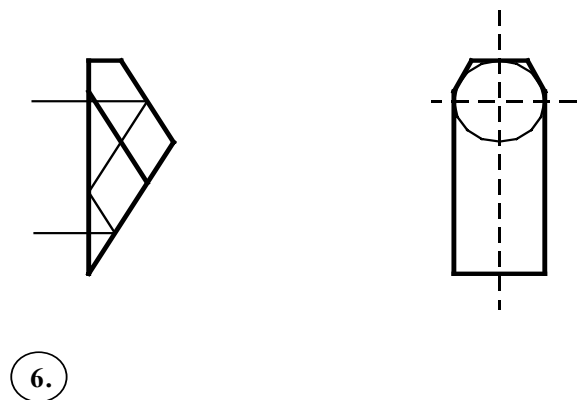
3. Перенос трех лучей на главный вид и построение их через призму



4. Удаление лишних построений



5. Формирование контура призмы



6. Результат

Рис. 12. Пример построения призмы ВКР-180

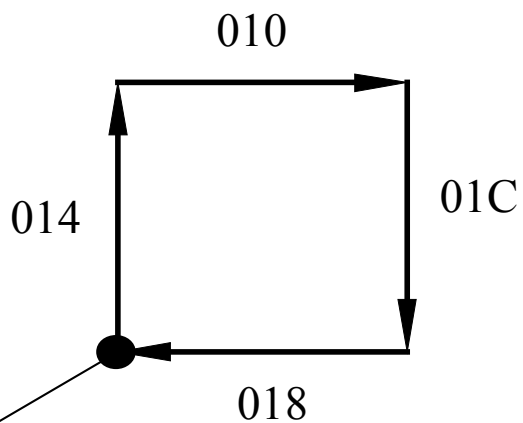
Таблица Б. Данные для задания 2 по вариантам

№	Призма	Диаметр D
1.	БКУ-60°	50
2.	БКП-90°	50
3.	БКР-180°	50
4.	ВКЛ-0°	50
5.	АКР-90°	50
6.	БКП-90°	50
7.	ВКР-45°	50

№	Призма	Диаметр D
8.	БКУ-60°	30
9.	БКП-90°	30
10.	БКР-180°	30
11.	ВКЛ-0°	30
12.	АКР-90°	30
13.	БКП-90°	30
14.	ВКР-45°	30

## Приложение Д. Примеры программ форм в основных кодах

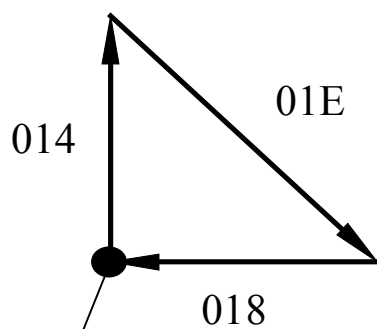
### 1) программа рисования квадрата



\*1, 5, kv  
014,  
010,  
01C,  
018,  
0

Это - точка вставки квадрата, от нее начинаем чертить. Далее направление отображения показано стрелками.

### 2) программа рисования прямоугольного треугольника

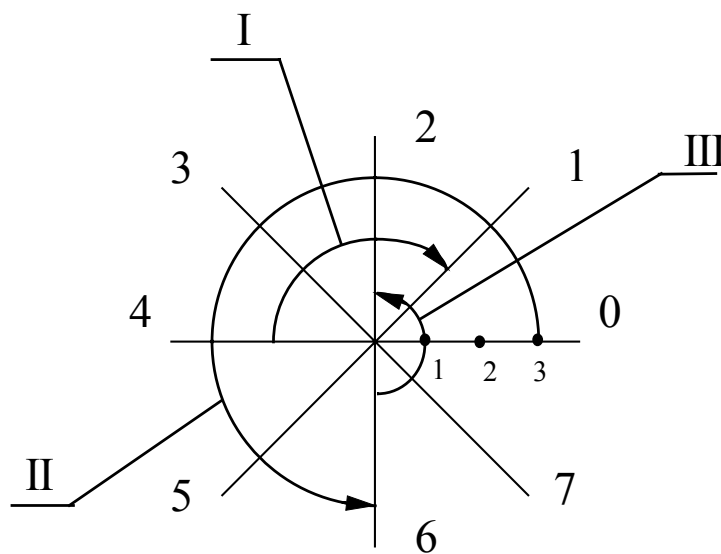


\*2, 4, tr  
014,  
01E,  
018,  
0

Это - точка вставки треугольника

## Приложение Е. Примеры описания дуг

### 1) описание октантных дуг

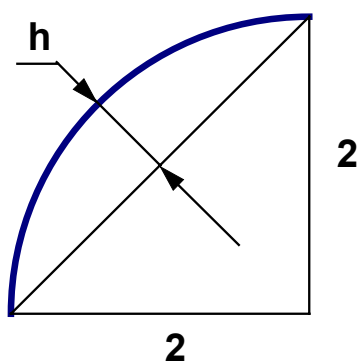


I: 00A,2,-043

II: 00A,3,006

III: 00A,1,064

### 2) описание сложных дуг



$\Delta X=2, \Delta Y=2, R=2$   
Таким образом, имеем:

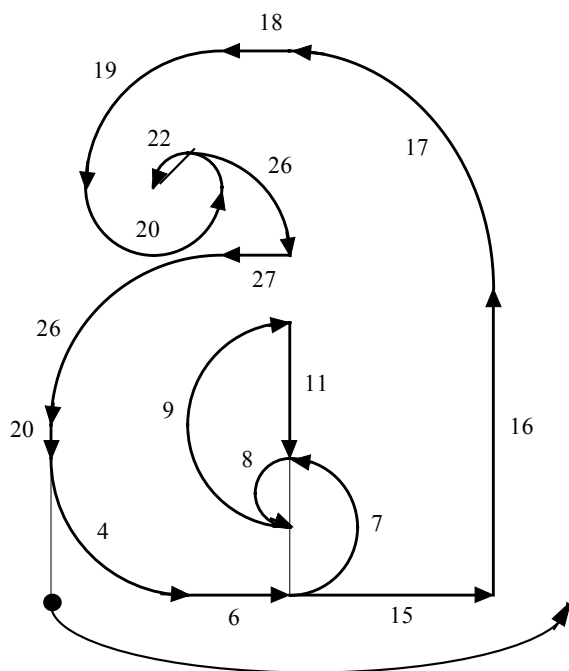
**00C, 2, 2,  $\tilde{h}$**

где:

$$\tilde{h} = \frac{2 - \sqrt{2}}{2\sqrt{2}} \cdot 256 = 128$$

## Приложение Ж. Примеры программы с использованием различных кодов

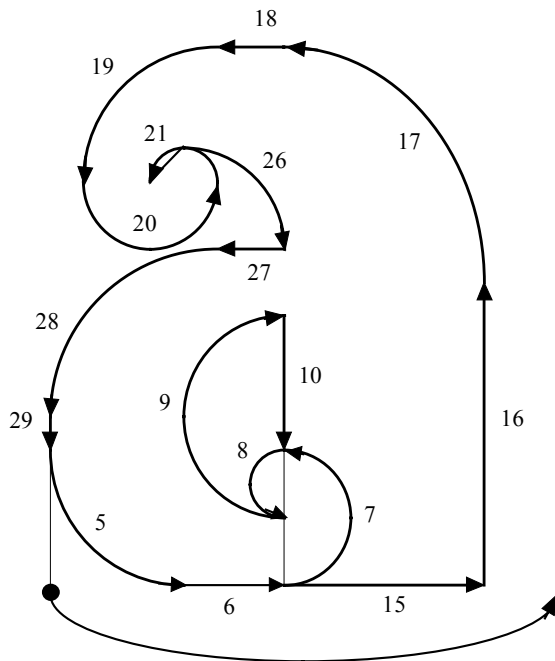
1) программа, отображающая на экране символа при помощи простейших кодов



- \*160,58,a
1. 002,
  2. 024,
  3. 001,
  4. 00A,2,042,
  5. 003,2,
  6. 030,
  7. 00A,2,064,

8. 00A,1,024,
9. 00A,3,-064,
10. 004,2,
11. 02C,
12. 002,
13. 02C,
14. 001,
15. 030,
16. 054,
17. 00A,3,002,
18. 018,
19. 00A,2,022,
20. 00A,1,044,
21. 003,2,
22. 00A,1,004,
23. 002,
24. 012,
25. 001,
26. 00A,3,-022,
27. 028,
28. 00A,5,022,
29. 004,2,
30. 01C,
31. 002,
32. 02C,
33. 070,
34. 0

2) программа отрисовывания символа  
с помощью последовательности дуг



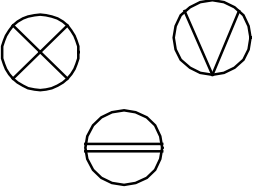
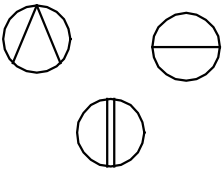
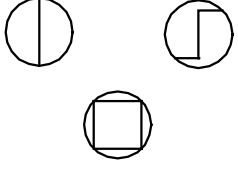
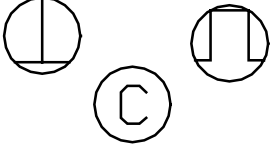
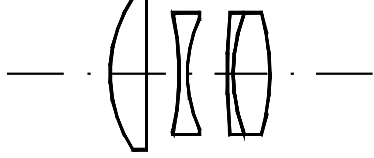
- \*160,75,a
1. 002,
  2. 024,
  3. 001,
  4. 003,2,
  5. 00D,4,-4,54,
  6. 3,0,0,
  7. 0,4,128,

8. 0,-2,128,
9. 0,6,-128,
10. 0,-4,0,
11. 0,0,
12. 002,
13. 04C,
14. 001,
15. 00D,6,0,0,
16. 0,10,0,
17. -6,6,54,
18. -2,0,0,
19. -4,-4,54,
20. 4,0,128,
21. -2,0,128,
22. 0,0,
23. 002,
24. 012,
25. 001,
26. 00D,3,-3,-54,
27. -2,0,0,
28. -10,-10,54,
29. 02C,
30. 0,0,
31. 004,2,
32. 002,
33. 02C,
34. 070,
35. 0



### Приложение 3. Задание 3. Создание программ в кодах для сложных форм

Таблица 3. Данные для задания 3 по вариантам

№	Задание	Иллюстрация
1.	Создать программы в кодах для отображения знаков покрытия поверхности.	
2.	Создать программы в кодах для отображения знаков покрытия поверхности.	
3.	Создать программы в кодах для отображения знаков покрытия поверхности.	
4.	Создать программы в кодах для отображения знаков покрытия поверхности.	
5.	Создать программу в кодах для отображения картинки “оптическая схема” .	
6.	Создать программу в кодах для отображения картинки “глаз” .	
7.	Создать программу в кодах для отображения картинок “отрицательная линза”, “положительная линза”, “мениск” .	
8.	Создать программу в кодах для отображения картинки “плоское зеркало” .	

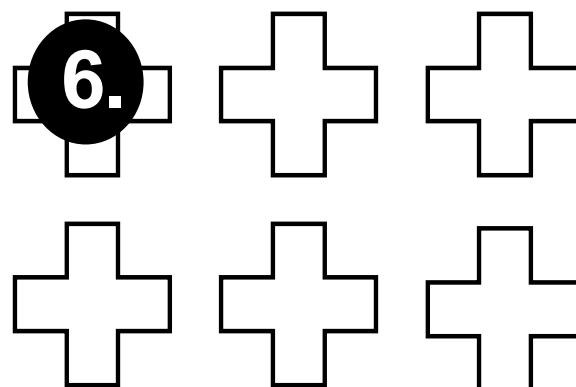
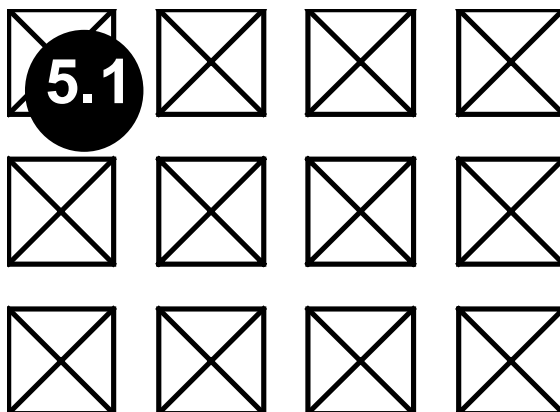
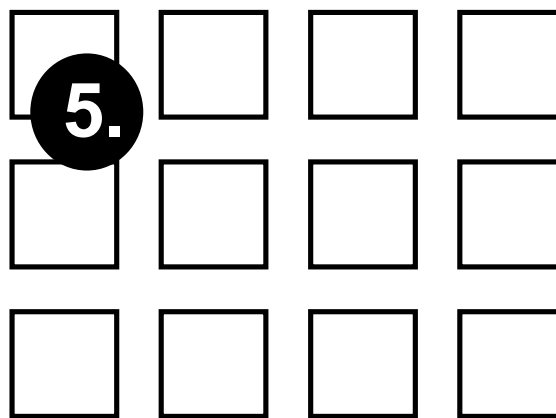
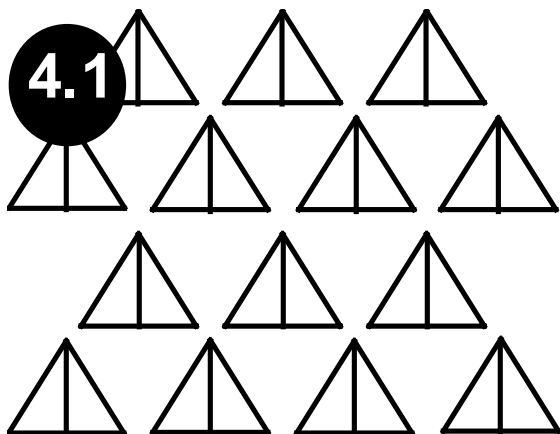
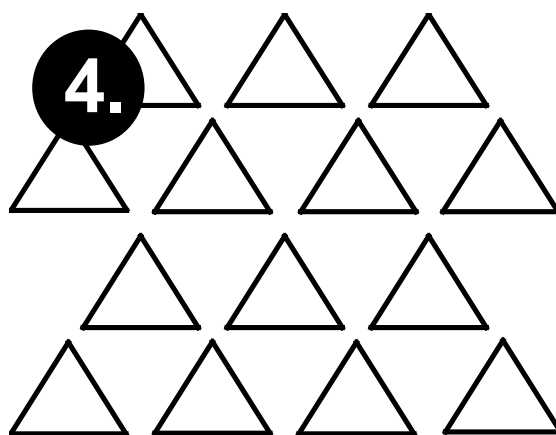
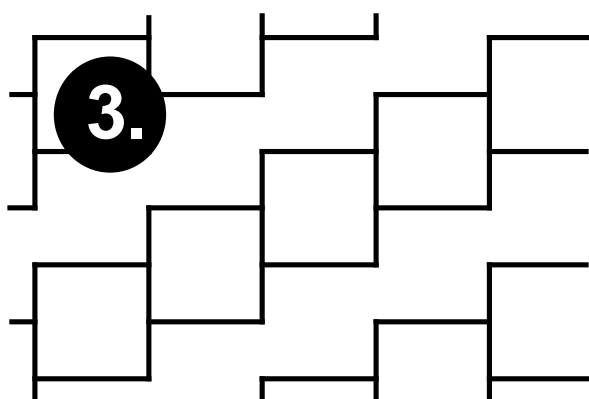
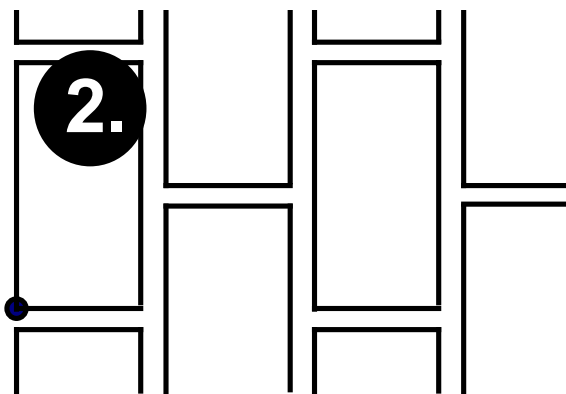
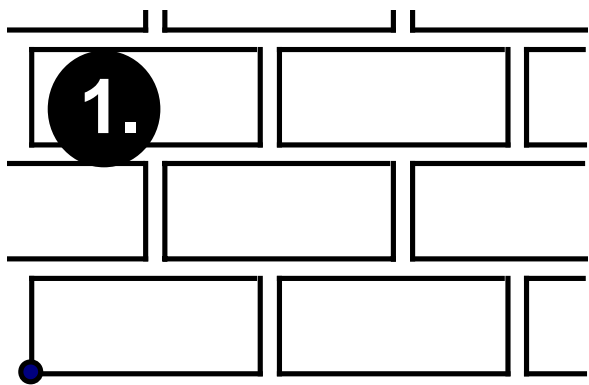
<b>9.</b>	Создать программу в кодах для отображения картинki “вогнутое зеркало” .
<b>10.</b>	Создать программу в кодах для отображения картинki “выпуклое зеркало” .
<b>11.</b>	Создать программу в кодах для отображения картинki “светоделительный кубик”, “плоскопараллельная пластинка” .
<b>12.</b>	Создать программу в кодах для отображения картинki “пентапризма”, “призма Дове” .
<b>13.</b>	Создать программу в кодах для отображения картинki “призма AP-90”, “призма БР-180” .
<b>14.</b>	Создать программу в кодах для отображения картинki “Телескопическая схема Кеплера”, “Телескопическая схема Галилея” .

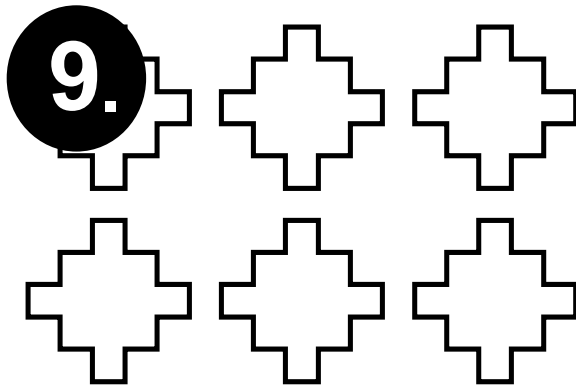
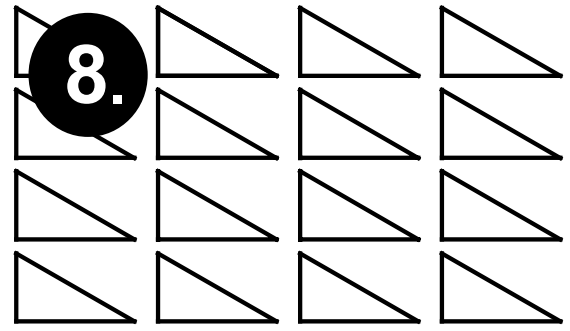
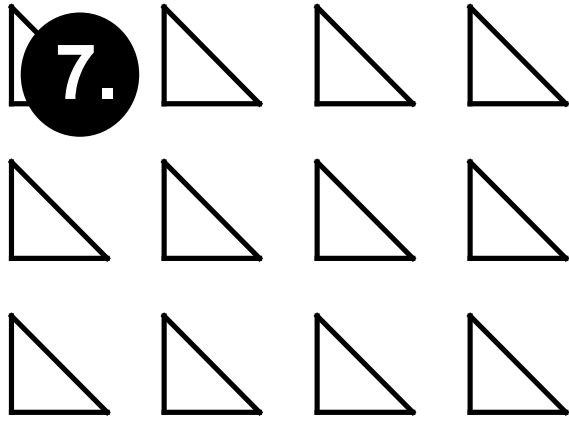
## Приложение И. Задание 4. Создание стилей линий

Таблица И. Данные для задания 4 по вариантам

№	Задание
1.	—      — —      —      — — —
2.	— — —      — — —      — — —
3.	— — —      -      — — —      -
4.	—      -      -      —      -      -      —      -      -
5.	—      -      —      —      -      —
6.	—      -      —      -      —      -
7.	— — — — — — — — — — — — — — — —
8.	—      -      —      -      —      -
9.	— — — — — — — — — — — — — — — —
10.	—      -      —      -
11.	— — — — — — — — — — — — — — — —
12.	—      - - - -      —      - - - -
13.	— — — — — — — — — — — — — — — —
14.	—      -      —      -      —      -

# Приложение К. Задание 5. Создание стилей штриховок





## Приложение Л. Пример программы на AutoLISP

### 1. Подпрограмма отображает указатель к поверхности линзы (детали)

```
;-----  
; sf - размер шрифта надписи  
; n_step - количество повторений  
;-----  
(defun surf(sf n_step)  
; начальные установки для ответов  
  (setq nnou "Н") (setq nnoul "н")  
; установка начального шага для повторений  
  (setq nls_step 1) (setq answ yyes)  
; создание имени для нового стиля  
  (setq name_st(strcat "s_" (rtos sf)))  
; создание стиля текста  
  (command "style" name_st "TXT" sf "0.8" "15" "" "" "")  
; вывод подсказки  
  (prompt "\nУказатель поверхности : ")  
; пока не отказались от повторения  
  (while (and (/= answ nnou) (/= answ nnoul))  
; имя поверхности  
    (setq surf_nam (getstring "\nИмя поверхности :"))  
; отключение режимов привязки  
    (command "osnap" "none")  
; место для надписи  
    (setq p_nar (getpoint "\nГде расположить надпись ?"))  
    (command "text" p_nar "" surf_nam)  
; указатель на поверхность  
    ($vekt 1)  
; наращивание счетчика шагов  
    (setq nls_step(1+ nls_step))  
)  
)
```

## 2. Подпрограмма вычерчивания самого указателя (стрелки)

```
;-----  
;   pr - признак острия, 1 - стрелка, 2 - база.  
;-----  
(defun $vekt (pr / p_end)  
;привязка к конечной точке примитива  
  (command "osnap" "end")  
  (setq p_end (getpoint "\nУкажите точку конца стрелки :"))  
;привязка к ближайшей точке  
  (command "osnap" "nea")  
  (setq p_nar (getpoint "\nУкажите точку острия стрелки :"))  
;база или стрелка  
  (if (= pr 1)  
    (progn  
      (command "dim")  
; задание острия стрелки в виде открытой стрелки  
      (command "DIMblk1" "_open"); вид стрелок  
      (command "DIMblk2" "_open"); вид стрелок  
      (command "exit")  
      (command "dimstyle" "s" "rus_open" "y")  
    )  
    (progn  
      (command "dim")  
; задание острия стрелки в виде базы  
      (command "DIMblk1" "_DATUMFILLED"); вид стрелок  
      (command "DIMblk2" "_DATUMFILLED"); вид стрелок  
      (command "exit")  
      (command "dimstyle" "s" "rus_base" "y")  
    )  
  )  
; вывод стрелки  
  (command "_leader" p_nar p_end "" "" "n")  
)
```

## **Приложение М. Задание 6. Создание программы параметрического чертежа на AutoLISP**

1. Формирование чертежа ирисовой диафрагмы (вставка в сборочный чертеж или отдельный вид).
2. Формирование чертежа призмы (любой) (вставка в сборочный чертеж и отдельный вид).
3. Формирование чертежа линзы.
4. Формирование сборочного чертежа склейки.
5. Формирование и подбор пружинящих колец и планок и построение чертежа.
6. Формирование стилей текста и размеров для дальнейшего использования.
7. Формирование чертежа корпуса детали.
8. Чертеж промежуточных колец и резьбовых колец.
9. Формирование чертежа щелевой диафрагмы (вставка в сборочный чертеж или отдельный вид).
10. Формирование спецификации для сборочного чертежа.
11. Формирование таблицы параметров на чертеже оптической детали.
12. Выбор размера из стандартного ряда.
13. Подбор допусков и посадок для конкретного вала или отверстия.
14. Расчет и формирование окулярной резьбы.



## ЛИТЕРАТУРА

- 1.Автокад: справочник команд. - Казань: ГАРМОНИЯ комьюникейшнз, 1994.
- 2.Аугер В. AutoCAD 11. - К.: Торгово-издательское бюро ВНУ, 1993.
- 3.Бугрименко Г.А. АВТОЛИСП - язык графического программирования в системе AutoCAD. - М.: Машиностроение, 1992.
- 4.Бугрименко Г.А., Лямке В.Н., Шейбокене Э.-К.С. Автоматизация конструирования на ПЭВМ с использованием системы AutoCAD. - М.: Машиностроение, 1993.
- 5.Гладков С.А. Программирование на языке AutoLISP в системе САПР AutoCAD. - М.: ДИАЛОГ-МИФИ, 1991.
- 6.Кречко Ю.А. AutoCAD: программирование и адаптация. - М.: ДИАЛОГ-МИФИ, 1995.
- 7.Кречко Ю.А., Полищук В.В. Автокад 13: новые возможности. В 2-х ч. - М.: ДИАЛОГ-МИФИ, 1996.
- 8.Кречко Ю.А., Полищук В.В. Автокад: курс практической работы. - М.: ДИАЛОГ-МИФИ, 1996.
- 9.Справочник конструктора оптико-механических приборов. Под ред. В.А. Панова. - Л.: Машиностроение, 1980.
- 10.AutoCAD. Полезные рецепты. Под ред. М.И. Кнеллера. - М.: Радио и связь, 1994.
- 11.AutoLISP. Программирование в AutoCAD 14 / Кудрявцев Е.М. - М.: «ДМК», 1999.

# СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ.....</b>	<b>3</b>
РЕКОМЕНДАЦИИ ПРЕПОДАВАТЕЛЮ.....	5
РЕКОМЕНДАЦИИ СТУДЕНТУ .....	5
<b>1. ОСНОВЫ РАБОТЫ С УНИВЕРСАЛЬНОЙ ПРОГРАММОЙ АВТОМАТИЗИРОВАННОГО КОНСТРУИРОВАНИЯ AUTOCAD.....</b>	<b>7</b>
1.1. ОБЗОР СОВРЕМЕННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В ОБЛАСТИ CAD/CAE/CAM .....	7
1.2. ОСНОВНЫЕ ПОНЯТИЯ AUTOCAD.....	9
1.2.1. Примитивы .....	9
1.2.2. Системы координат .....	9
1.2.3. Единицы измерения и масштаб .....	9
1.2.4. Вид.....	10
1.2.5. Слой.....	10
1.2.6. Чертёж.....	10
1.3 ИНТЕРФЕЙС ПРОГРАММЫ.....	11
<b>2. СОЗДАНИЕ ИЗОБРАЖЕНИЙ.....</b>	<b>14</b>
2.1. КОМАНДЫ ОБЩЕГО НАЗНАЧЕНИЯ .....	14
2.1.1. Способы задания точек в поле чертежа.....	14
2.1.1. Команды вычерчивания примитивов.....	15
2.1.2. Установка режима привязки .....	19
2.2. СВОЙСТВА ПРИМИТИВОВ .....	20
2.2.1. Цвет .....	20
2.2.2. Тип линии .....	20
2.2.3. Масштабирование линий .....	21
2.3. ШТРИХОВКА.....	22
2.4. РАБОТА С ТЕКСТОМ .....	23
<b>3. РАБОТА ПО МОДЕРНИЗАЦИИ ЧЕРТЕЖА .....</b>	<b>25</b>

3.1. ПЕРЕЧЕНЬ КОМАНД РЕДАКТИРОВАНИЯ .....	26
3.2. РЕЖИМЫ ВЫБОРА ОБЪЕКТОВ ДЛЯ РЕДАКТИРОВАНИЯ .....	29
<b>4. СЛОИ.....</b>	<b>31</b>
<b>5. БЛОКИ .....</b>	<b>32</b>
5.1. СОЗДАНИЕ БЛОКОВ .....	32
5.2. ВСТАВКА БЛОКОВ .....	32
<b>6. ПРОГРАММИРОВАНИЕ В КОДАХ .....</b>	<b>33</b>
6.1. ФАЙЛ С ТЕКСТОМ ПРОГРАММЫ .SNP .....	33
6.2. ОСНОВНЫЕ КОДЫ .....	33
6.3. ВСПОМОГАТЕЛЬНЫЕ КОДЫ .....	34
6.4. СОЗДАНИЕ ШРИФТА.....	37
6.4. КОМПИЛИРОВАНИЕ ПРОГРАММЫ.....	37
6.5. СЧИТЫВАНИЕ И ЗАГРУЗКА ФОРМ.....	37
6.6. ВСТАВКА ФОРМЫ .....	37
<b>7. СТИЛИ ЛИНИЙ.....</b>	<b>38</b>
7.1. ФАЙЛ С ТЕКСТОМ ПРОГРАММЫ ACAD.LIN .....	38
7.2. СОЗДАНИЕ ТИПА ЛИНИЙ.....	38
<b>8. ШТРИХОВКИ .....</b>	<b>40</b>
8.1. ФАЙЛ С ТЕКСТОМ ПРОГРАММЫ ACAD.PAT .....	40
8.2. ОПИСАНИЕ ШТРИХОВОК.....	40
<b>9. ПРОГРАММИРОВАНИЕ НА AUTOLISP .....</b>	<b>42</b>
9.1. СИНТАКСИС ЯЗЫКА .....	42
9.1.1. Переменные AutoLISP .....	42
9.1.2. Типы данных AutoLISP.....	43
9.1.3. Выражения AutoLISP .....	43
9.2. ОБЗОР ФУНКЦИЙ AUTO LISP .....	45
9.2.1. Математические функции.....	45

9.2.2. Логические функции AutoLISP .....	47
9.2.3. Использование функций GET для ввода данных.....	47
9.2.4. Вывод данных в командную строку.....	48
9.2.5. Условное ветвление программ .....	49
9.2.6. Организация циклов .....	50
9.2.7. Вызов команд AutoCAD из программы на AutoLISP.....	50
9.2.8. Файловый ввод/вывод .....	50
<b>9.3. ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ПРИ СОЗДАНИИ</b>	
<b>ПАРАМЕТРИЧЕСКОГО ЧЕРТЕЖА.....</b>	<b>52</b>
9.3.1. Вставка рамки.....	52
9.3.2. Ввод основных параметров.....	52
9.3.3. Опрос пользователя по начальным установкам.....	53
9.3.4. Вычерчивание.....	53
<b>10. ОСОБЕННОСТИ РАБОТЫ В СРЕДЕ AUTOCAD 14 ВЕРСИИ. 54</b>	
10.1. ВВЕДЕНИЕ. РАБОТА С ЧЕРТЕЖОМ. СОЗДАНИЕ, СОХРАНЕНИЕ.....	54
10.1.1. Сохранение .....	54
10.1.2. Вывод панелей инструментов на экран .....	54
10.2. РЕЖИМЫ РАБОТЫ В СРЕДЕ AUTOCAD .....	55
10.3. СВОЙСТВА ОБЪЕКТА.....	55
10.4. УВЕЛИЧЕНИЕ ЧЕРТЕЖА И ЕГО ЧАСТЕЙ .....	56
10.5. УСТАНОВКА ПРИВЯЗОК К ТОЧКАМ ЧЕРТЕЖА.....	56
10.6. ОСОБЕННОСТИ МОДИФИЦИРОВАНИЯ ЭЛЕМЕНТОВ ЧЕРТЕЖА .....	56
10.7. РАБОТА С ТЕКСТОМ .....	57
10.8. СЛОИ ЧЕРТЕЖА .....	58
10.9. РАЗМЕРЫ, ИХ ПРОСТАНОВКА И ИСПОЛЬЗОВАНИЕ .....	58
10.10. БЛОКИ.....	59
10.10.1. Создание блока.....	59
10.10.2. Вставка блока .....	59
10.11. ТРЕХМЕРНОЕ МОДЕЛИРОВАНИЕ .....	60
10.11.1. Пространство и координаты .....	60

10.11.2. 3D-объекты: тела, поверхности.....	60
10.11.3. Способы получения 3D объектов.....	60
10.11.4. Модифицирование 3D - объектов.....	61
10.11.5. Операции над объектами в двухмерном сечении.....	61
10.12. ПРИНТЕРЫ. ПЛОТТЕРЫ.....	61
10.12.1. Вывод чертежей на принтер/плоттер.....	62
10.13. СОЗДАНИЕ ЧЕРТЕЖА-ПРОТОТИПА.....	63
10.14. РАБОТА СО ССЫЛКАМИ НА ЧЕРТЕЖИ И КАРТИНКИ. РАБОТА С AUTOCAD В СЕТИ.....	63
10.14.1. Применение ссылок.....	63
10.14.2. Установка ссылок.....	63
10.14.3. Работа в сети.....	64
<b>ПРИЛОЖЕНИЯ.....</b>	<b>65</b>
Приложение А. СПОСОБЫ ПОСТРОЕНИЯ ОКРУЖНОСТИ.....	65
Приложение Б. СПОСОБЫ ПОСТРОЕНИЯ ДУГИ.....	66
Приложение В. ЗАДАНИЕ 1. ВЫПОЛНЕНИЕ ЧЕРТЕЖА ЛИНЗЫ.....	67
Приложение Г. ЗАДАНИЕ 2. ВЫПОЛНЕНИЕ ЭСКИЗА ЧЕРТЕЖА ПРИЗМЫ С ХОДОМ ЛУЧЕЙ.....	69
Приложение Д. ПРИМЕРЫ ПРОГРАММ ФОРМ В ОСНОВНЫХ КОДАХ.....	71
Приложение Е. ПРИМЕРЫ ОПИСАНИЯ ДУГ.....	72
Приложение Ж. ПРИМЕРЫ ПРОГРАММЫ С ИСПОЛЬЗОВАНИЕМ РАЗЛИЧНЫХ КОДОВ.....	73
Приложение З. ЗАДАНИЕ 3. СОЗДАНИЕ ПРОГРАММ В КОДАХ ДЛЯ СЛОЖНЫХ ФОРМ.....	75
Приложение И. ЗАДАНИЕ 4. СОЗДАНИЕ СТИЛЕЙ ЛИНИЙ.....	77
Приложение К. ЗАДАНИЕ 5. СОЗДАНИЕ СТИЛЕЙ ШТРИХОВОК.....	78
Приложение Л. ПРИМЕР ПРОГРАММЫ НА AUTOLISP.....	80
Приложение М. ЗАДАНИЕ 6. СОЗДАНИЕ ПРОГРАММЫ ПАРАМЕТРИЧЕСКОГО ЧЕРТЕЖА НА AUTOLISP.....	82



Толстоба Надежда Дмитриевна  
Компьютерное конструирование  
оптических приборов  
Учебное пособие

В авторской редакции

Компьютерное макетирование

Н.Д. Толстоба

Зав. редакционно-издательским отделом

Н.Ф. Гусарова

Лицензия ИД № 00408 от 05.11.99

Подписано к печати 05.04.2000

Отпечатано на ризографе. Тираж 100 экз. Заказ № 219.