



# ОСНОВЫ ПРОГРАММИРОВАНИЯ

# ПЛАН:

1. Система программирования
2. Языки программирования низкого и высокого уровней



# 1. Система программирования

**Система программирования – комплекс инструментальных программных средств, предназначенный для работы с программами на одном из языков программирования.**

▪





# Этапы создания программы

Алгоритм

программист

Текст программы на  
каком-либо языке  
программирования

транслятор

Машинный код

Очистить экран  
Ввести a  
Вычислить площадь  
Вывести результат

```
CLS  
INPUT a  
s=a*a  
PRINT s  
END
```

```
001011101010  
010101100101  
010111010110  
101101111010
```

начало

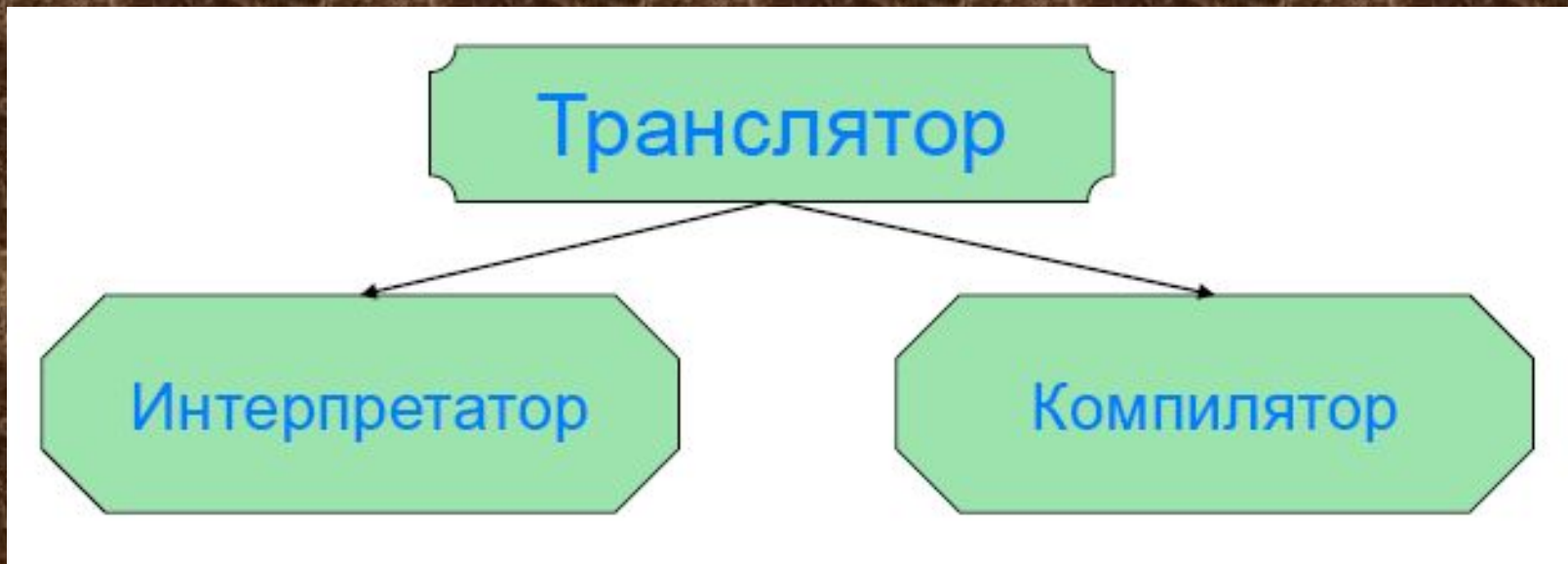
Очистить экран

## Состав системы программирования:

- транслятор (компилятор или интерпретатор);
- интегрированная среда разработки;
- средства создания и редактирования текстов программ;
- библиотеки стандартных подпрограмм и функций;
- отладочные программы;
- «дружественная" к пользователю диалоговая среда;
- многооконный режим работы;
- мощные графические библиотеки;
- утилиты для работы с библиотеками;
- встроенный ассемблер;
- встроенная справочная служба;
- другие специфические особенности.



**Транслятор** (англ. translator — переводчик) — это программа-переводчик. Она преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд. Трансляторы реализуются в виде компиляторов или интерпретаторов.



**Компилятор** (англ. compiler — составитель, собиратель) читает всю программу целиком, делает ее перевод и создает законченный вариант программы на машинном языке, который затем и выполняется.

Откомпилированные программы работают быстрее.





**Интерпретатор** (англ. interpreter — истолкователь, устный переводчик) переводит и выполняет программу строка за строкой.

После того, как программа откомпилирована, исходная программа и компилятор больше не нужны. В то же время программа, обрабатываемая интерпретатором, должна заново переводиться на машинный язык при каждом очередном запуске программы.

Интерпретируемые программы проще исправлять и изменять.



## Интегрированная среда разработки (IDE) –

система программных средств, используемая программистами для разработки программного обеспечения.

Состав:

- текстовый редактор;
- компилятор и/или интерпретатор;
- средства автоматизации сборки;
- отладчик.



**Язык****IDE**

Си/C++

Borland C++, C++ Builder ...

Бэйсик

Turbo Basic, Visual Basic ...

Паскаль

Delphi, Turbo Pascal ...

ActionScript

Adobe Flash, Adobe Flash  
Builder ...

Универсальные IDE

Visual Studio, Komodo, Eclipse



Отладчики – специальные средства, позволяющие исследовать внутреннее поведение программы.

Возможности:

- пошаговое исполнение программы с остановкой после каждой команды (оператора);
- просмотр текущего значения любой переменной или нахождение значения любого выражения;
- установка в программе «контрольных точек», т.е. пошаговое выполнение программы.



## **2. Языки программирования низкого и высокого уровней**

Языки программирования делятся на:

- **машинноориентированные языки** (языки низкого уровня): Автокоды, Ассемблеры – позволяют управлять вычислительным процессом напрямую, при помощи машинных команд,
- **языки программирования высокого уровня.**

# Машинный язык (40-50 годы XX в.)

Программы на машинном языке – очень длинные последовательности единиц и нулей, являлись машинно-зависимыми (т.е. для каждой ЭВМ необходимо было составлять свою программу)

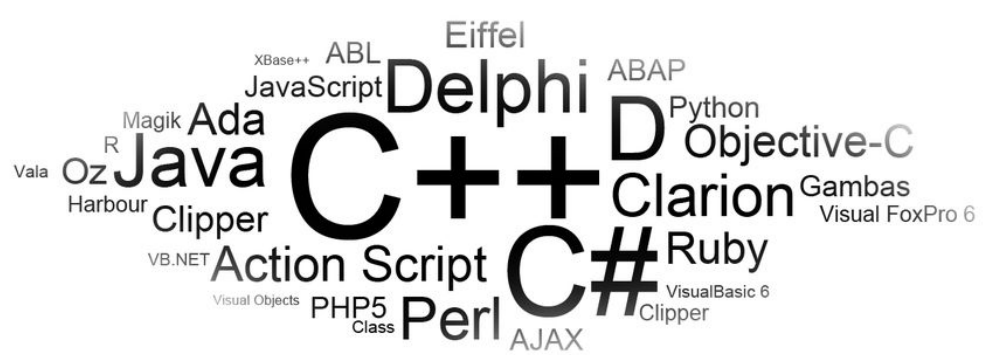




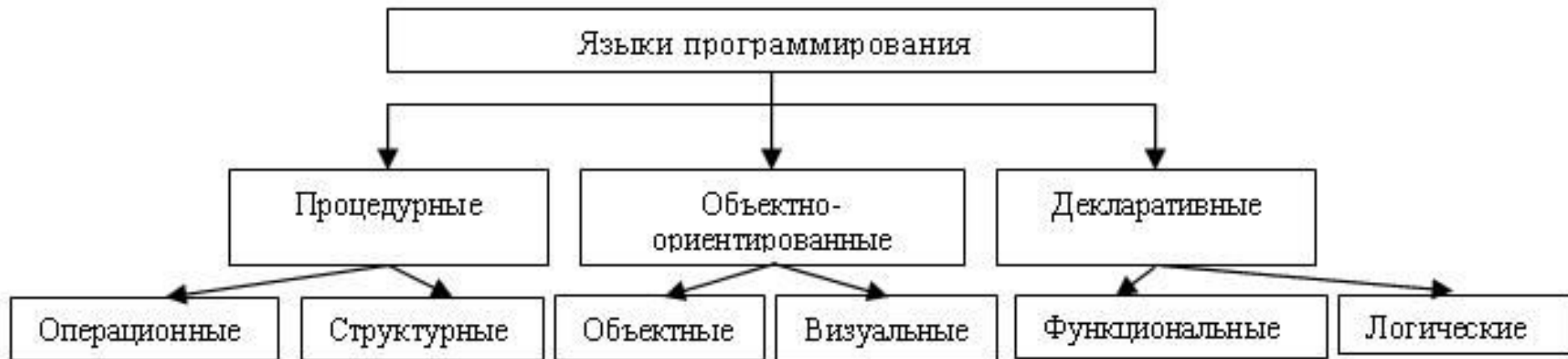
# Ассемблер (начало 50-ых годов XX в.).


Ассемблер обеспечивает возможность применения символических имен в исходной программе и избавляет программиста от утомительного труда (неизбежного при программировании на языке машинных команд) по распределению памяти компьютера для команд, переменных и констант.

```
0400 2073FE JSR #FE73      s~
0403 A200   LDX ##0         "□
0405 BD8004 LDA $480,X     =□λ
0408 F006   BEQ $410      p✓
040A 2075FE JSR #FE75      u~
040D E8     INX         h
040E D0F5   BNE $405      Pu
0410 00     BRK         □
0411 B9     *=$480
0480 48     'H         H
0481 45     'E         E
0482 4C     'L         L~
0483 4C     'L         L
0484 4F     'O         O
0485 00     $0         □
0486 67     !
```



# Схема классификации языков программирования высокого уровня





**Процедурное программирование** - представляет собой последовательность команд, определяющих алгоритм решения задачи.

Основная идея - использование памяти для хранения данных.

Основная команда - присвоение, с помощью которой определяется и меняется память компьютера.

Программа производит преобразование содержимого памяти, изменяя его от исходного состояния к результирующему.



# Языки процедурного программирования

*Операционные языки (привязанные к конкретной архитектуре ЭВМ):*

- Фортран (начало 50-х г.) - первый компилируемый язык для программирования научно-технических задач;
- Кобол (конец 60-х г.) - для решения задач обработки больших объемов данных, хранящихся на различных носителях данных;
- Алгол (1960 г.) – многоцелевой расширенный язык, в котором впервые введены понятия «блочная структура программы» и «динамическое распределение памяти»;
- BASIC (середина 60-х г.) - характеризуется простотой освоения и наличием универсальных средств для решения научных, технических, экономических и игровых задач.

## *Структурные языки (не привязанные к конкретной архитектуре ЭВМ):*

- PL-1 (1963-1966гг.) - хорошо приспособлен для исследования и планирования вычислительных процессов, моделирования, решения логических задач, разработки систем математического обеспечения.
- PASCAL (1968-1971гг.) - популярный для ПК, в основу которого положен подход от общей задачи к частным (более простым и меньшим по объему).
- АДА (1979 г) - назван в честь первой программистки Ады Лавлейс. Его отличает модульность конструкций.
- СИ (начало 70-х г.) - первоначальный его вариант планировался как язык для реализации операционной системы Unix вместо языка Ассемблера. Особенность языка - это то, что различия между выражениями и операторами сглаживаются, что приближает его к функциональным языкам программирования.
- Модула (1980 г.) - для профессиональных системных программистов.



**Объектно-ориентированное программирование (ООП)** — это метод программирования, при использовании которого главными элементами программ являются объекты.

Объединение данных и свойственных им процедур обработки в одном объекте, называется инкапсуляцией и является одним из важнейших принципов ООП.



# Языки ООП



## Объектные языки:

- C++ ,
- Java.

Визуальные языки (середина 90-х г.) - интерфейсная часть программного продукта создается в диалоговом режиме, практически без написания программных операторов:

- Visual Basic,
- Delphi,
- C++ Builder,
- Visual C++,
- Object PAL,
- dBase.

Язык VBA (Visual Basic for Application) – язык приложений Microsoft Office (Excel, Word, Power Point и др), который соблюдает основной синтаксис языка и правила программирования языков Basic – диалектов, что позволяет создавать макросы для автоматизации выполнения некоторых операций и графический интерфейс пользователя, интеграцию между различными программными продуктами.

Языки программирования для компьютерных сетей являются интерпретируемыми. Интерпретаторы для них распространяются бесплатно, а сами программы – в исходных текстах. Такие языки называются скрипт – языками.

- **Perl** (1987 г.) язык для обработки больших текстов и файлов и расшифровывается, как язык для практического извлечения данных и составления отчетов. С помощью него можно создать скрипт, который открывает один или несколько файлов, обрабатывает информацию и записывает результаты.

- **PHP** (1995-1997гг.) обладает средствами доступа к БД и используется создателями динамических сайтов во всем мире.

- **Tcl/Tk** (конец 80-х г.) состоит из мощных команд, предназначенных для работы с абстрактными нетипизированными объектами и позволяет создавать программы с графическим интерфейсом.

- **VRML** (1994 г.) создан для организации виртуальных трехмерных интерфейсов в Интернете.

- **XML** - с 1996 г. идет работа над созданием универсального языка структуры документов. Может стать заменой языка HTML.

Декларативные языки программирования – это функциональные и логические языки программирования.

**Функциональное программирование**- это способ составления программ, в которых единственным действием является вызов функции. Программа, написанная на функциональном языке, представляет собой последовательность описания функций и выражений. Выражение вычисляется сведением сложного к простому. Все выражения записываются в виде списков.

**Логическое программирование** - это программирование в терминах логики.

## Языки декларативного программирования

### Язык функционального программирования:

Лисп (1959 г) - позволяет обрабатывать большие объемы текстовой информации.



### Язык логического программирования:

Пролог (1973 г.) - язык искусственного интеллекта.

Программа на языке Пролог строится из последовательности фактов и правил, затем формулируется утверждение, которое Пролог пытается доказать с помощью правил. Язык сам ищет решение с помощью методов поиска и сопоставления, которые в нем заложены.