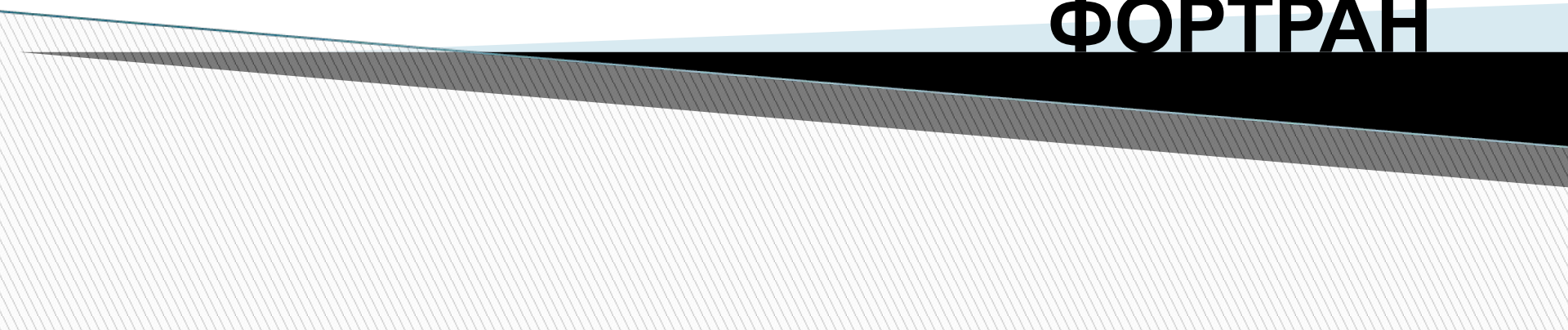


Лекция 3
ПРОГРАММИРОВАНИЕ.
ОСНОВНЫЕ ЭЛЕМЕНТЫ ЯЗЫКА
ФОРТРАН



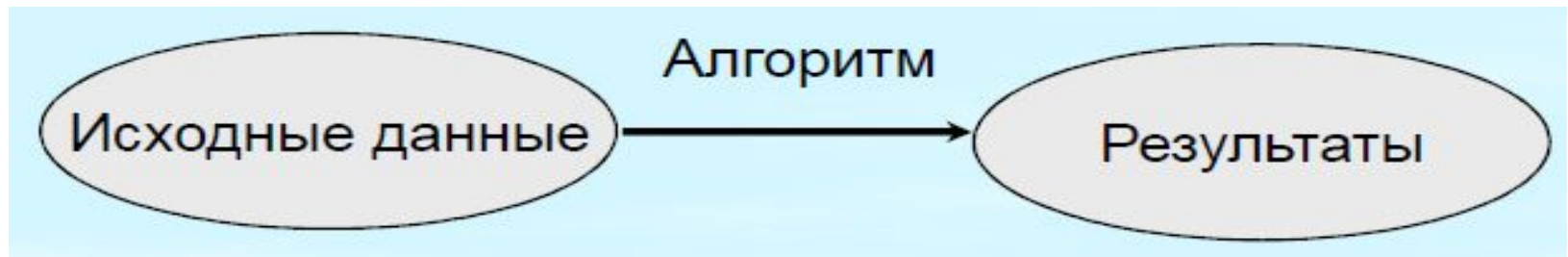
Литература

1. Пудан Л.Я. Информатика. Основы алгоритмизации и программирования на ФОРТРАНЕ, 2002. – Изд-во ТГАСУ.
2. Мудров А.Е. Численные методы для ПЭВМ на языках Бейсик, Фортран и Паскаль. Издательство: РАСКО, 1991
3. Бартеньев О.В. Фортран для студентов, 1999
4. Соловьев П.В. FORTRAN для персонального компьютера, 1991

Решение задач с помощью компьютера



Алгоритм – точное описание процесса обработки исходных данных *исполнителем*, ведущее от исходных данных к результату.



Исполнитель – человек или компьютер, понимающий язык, на котором записан алгоритм.

Единственным языком, понятным компьютеру является язык **машинных команд. Он реализуется аппаратно: каждую команду выполняет некоторое электронное устройство.**

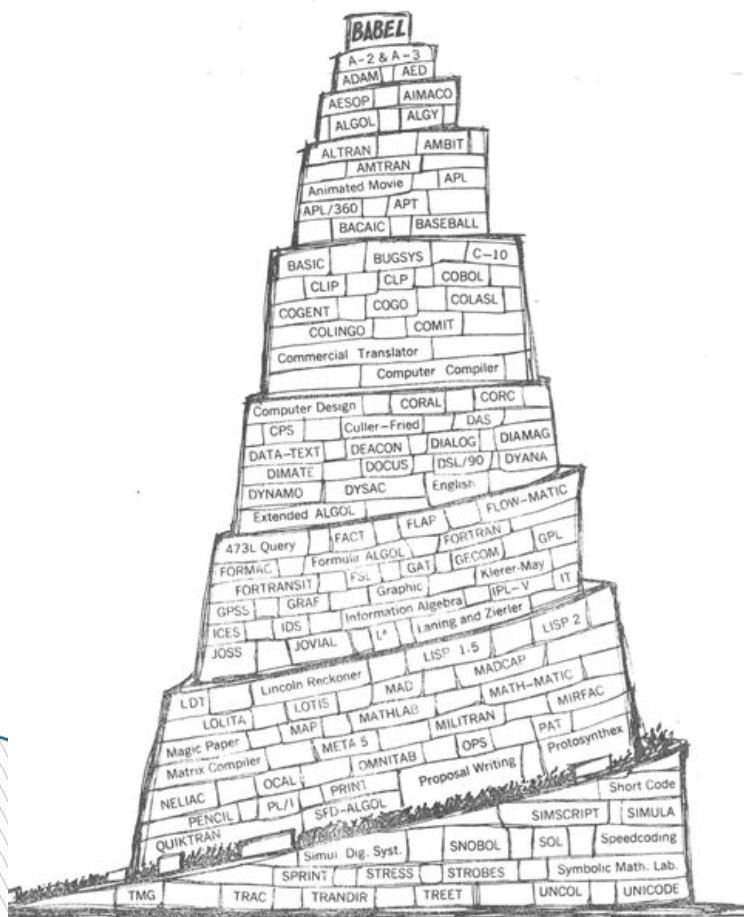
С появлением первых компьютеров стали развиваться формализованные **алгоритмические языки программирования, понятные человеку-программисту. Эти языки не ориентированы на конкретный компьютер, а предназначены для записи алгоритмов.**

В настоящее время в мире существует несколько сотен реально используемых языков программирования. Для каждого есть своя область применения

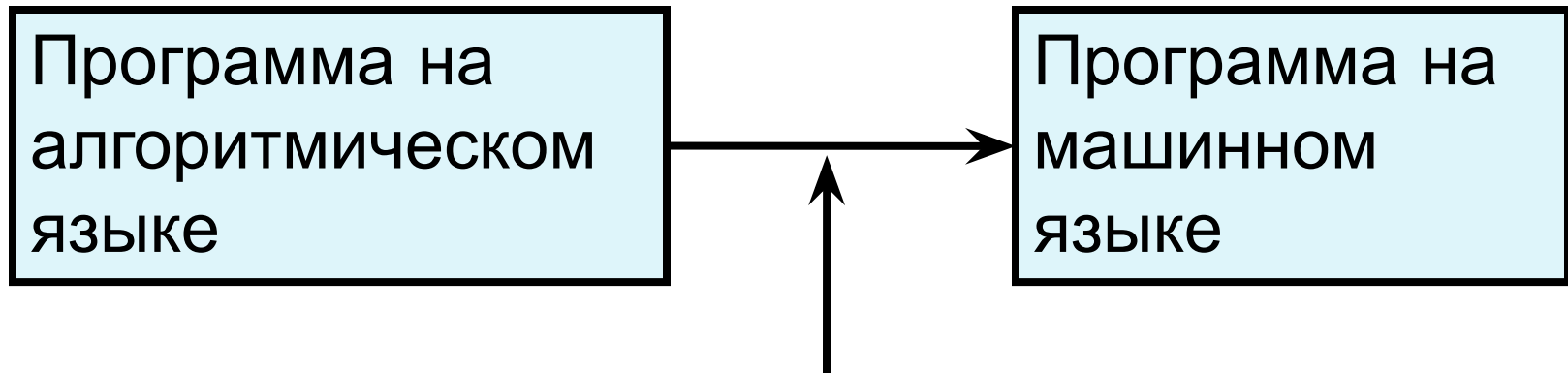
Наиболее активный период разработки языков и систем программирования приходится на 1960-е годы.

«Вавилонская башня»
языков программирования

За это десятилетие в мире родилось более тысячи разнообразных языков, как универсальных, так и специализированных, но выжили и доросли до XXI века немногие, в том числе бессмертные **Fotran, Basic, Algol, Cobol, Simula, Lisp** и их потомки.



Трансляция и компиляция программ



Компиляция (трансляция) или интерпретация программы на алгоритмическом языке в программу на машинном языке выполняется автоматически соответствующими программами **компиляторами (трансляторами) или интерпретаторами**

Программа – интерпретатор (Бейсик, Паскаль) анализируют отдельные части программы (исходного кода) и могут выполнять их без создания исполняемого *.exe файла.

Программа – компилятор (Фортран) переводит в объектный код лишь завершённую программу (исходный код), дополнительно необходима компоновка(сборка) и создание исполняемого *.exe файла, который и выполняет программу.

Основные элементы ФОРТРАНА

1. АЛФАВИТ ФОРТРАНА

a) прописные и строчные латинские буквы **A B C ... Z** и знак подчеркивания **_** (регистр букв не различается);

Замечание: русские буквы нельзя использовать в операторах (можно только в комментариях или символьных константах и переменных).

b) арабские цифры от **0** до **9**;

Основные элементы ФОРТРАНА

с) специальные знаки:

+	'...', "..." – апострофы и двойные кавычки
–	□ – символ пробела
*	:
/ и //	; – отделяет операторы, если они записываются в одной строке бланка
()	! – начало строки комментариев
=	& (амперсанд) – знак продолжения строки оператора (максимальная длина 132 символа)
. и ,	

Основные элементы ФОРТРАНА

2. КОНСТАНТЫ И

ПЕРЕМЕННЫЕ

Опр. Константы – это величины, значение которых не изменяется в процессе выполнения программы.

Опр. Переменные - это данные, значения которых при выполнении программы **могут изменяться**. Каждая переменная имеет **имя и значение**.

В программе константы представлены в **явной** форме. Например, $\pi = 3.14159$.

Переменные с именами a , x , y и их значения:
 $a = 3.7$, $x = -67$, $y = 0.087$.

Правило выбора имен

Имя составляется из букв **латинского алфавита** (прописных или строчных), цифр и символа подчеркивания **_**. **Первой** в имени может быть только **буква**.

Запрещается использование в имени символа **пробел**.

Примеры правильных имен:

X a Z6 pq9 k3 t_23

Примеры неправильных имен:

27q – не начинается с буквы;

***5** – не начинается с буквы и содержит символ *;

K-9 E.G. – содержит символы, не являющимися буквами и цифрами;

A 78 - имя содержит пробел.¹¹

Типы данных

Тип значения переменной определяет *тип переменной* и, наоборот, *тип переменной* определяет *тип значения переменной*.
Поэтому различают переменные следующих типов:

- *Целый (INTEGER),*
- *Вещественный (REAL),*
- *Комплексный (COMPLEX),*
- *Логический (LOGICAL),*
- *Символьный (CHARACTER).*

Целый тип (INTEGER)

- это целые числа (со знаком или без знака).

Например, **-55** , **+67**, **100**.

Для целых констант/переменных используется **точное** двоичное представление - поразрядная запись цифр числа в ячейке памяти.

Арифметические операции с такими константами проводятся с получением **целочисленного результата** (в этом особенность работы с целыми числами в Фортране).

Например, **4/5=0**, **9/4=2** (в примерах знак / определяет операцию деления).

Вещественный тип (REAL)

используется для записи действительных чисел
– чисел с дробной десятичной частью.

Константы такого типа могут быть записаны:

1. в обычном виде с целой и дробной *десятичной* частью (со знаком или без знака),
например, **93.678** **-5.76** **-9.0** (или **-9.**)
.934 (или **0.934**).

форма F – форма с фиксированной *точкой*

2. в *экспоненциальной форме* или в *форме E*.

$$\pm q \cdot 10^{\pm N}$$

q - мантисса числа,
 $\pm N$ - *целый* порядок.

Вещественный тип (REAL)

В Фортране такие числа записываются в одну линию, и основание степени **10** заменяется символом **E** (знак умножения между **q** и **E** не ставится)

$$\pm qE \pm N$$

Примеры записи чисел в форме E:

Обычная запись	Экспоненциальная форма	Форма E
-12.564	$- 1.2564 \cdot 10^1$	- 1.2564E 1
0.000345	$3.45 \cdot 10^{-4}$	3.45E -4
-5678.6543	$-5.6786543 \cdot 10^3$	-5.6786543D 3

Символьный и логический типы

Символьные константы (тип **CHARACTER**)

– это любая последовательность символов, заключенных в апострофы или двойные кавычки, например:

'LAB'

"STUDENT"

'45_rty'

Логические константы (тип **LOGICAL**)

могут принимать только два значения: **.TRUE.** (Истина) и **.FALSE.** (Ложь).

Два способа задания переменных

1). правило умолчания (*REAL, INTEGER*)

Если имя переменной *начинается* с одной из букв **I, J, K, L, M, N**, (или **i, j, k, l, m, n**), то переменная имеет тип **целый** (ее значение – число целое).

В противном случае тип переменной **вещественный** (ее значение - число с дробной десятичной частью).

целый тип: **k6 NT mas JY IC4 L2001**

вещественный тип: **F_L x Y1 r123 PT5**

Два способа задания переменных

2). явное задание типа данных

Описание типа переменных с помощью *явного объявления типа. Примеры:*

INTEGER XC, R6, NY ! переменные целые

REAL LH, m9, NB, xx ! переменные
вещественные

При объявлении типа переменных можно, кроме типа, объявлять и *начальные значения переменных (инициализация):*

INTEGER :: A=490, b=-2

REAL :: PI=3.14159, M=0.5

Основные элементы ФОРТРАНА

3. ОПЕРАЦИИ

Операции определяют действия над данными.

Арифметические операции:

Название операции	Символ операции
Сложение	+
Вычитание	—
Умножение	*
Деление	/
Возведение в степень	**

Основные элементы ФОРТРАНА

4. ВЫРАЖЕНИЯ

Порядок вычисления выражения определяется определенным **приоритетом** операций и скобок:

- Вычисления в скобках;
- Обращение к функции;
- Возведение в степень ******;
- Умножение или деление *****, **/** ;
- Сложение или вычитание **+**, **-** .

Правило определения типа AB

- Если в состав выражения входит хотя бы **одна вещественная величина**, то выражение имеет **вещественный тип**.
Результат - число вещественное.
- Если в выражении только **целочисленные данные**, то результат – число целое и выражение имеет **тип целый**.

Основные элементы ФОРТРАНА

5. ВСТРОЕННЫЕ ФУНКЦИИ

Правило: *аргумент функции*
обязательно записывается в скобках.

abs(x) - МОДУЛЬ $|x|$

sqrt(x) - Корень квадратный \sqrt{x}

sin(x) - СИНУС (АРГУМЕНТ В РАДИАНАХ)

Примеры арифметических выражений (АВ):

$\frac{5}{6} * t$	<p>5.0/6.0*t целые числа переведены в вещественные, так как 5/6=0</p>
$\sin^2 43^0$	<p>sin(43*3.14159/180)**2</p>
$\sqrt[7]{\frac{f+g}{t}}$	<p>((f+g)/t)**(1./7.)</p>
$\cos^3 \sqrt{ \sin Z }$	<p>cos(sqrt(abs(z)))**3</p>
\tan^2	<p>tan(x**2)**2</p>