

# Языки и системы программирования

Лекция 1.  
Основные понятия

# Алгоритм

**Алгоритм** – понятное и точное предписание исполнителю выполнить конечную последовательность действий, приводящих от исходных данных к искомому результату.  
О.Ю. Заславская, И.В. Левченко

**Algorithmi** – латинское написание имени Мухаммеда аль-Хорезми (787-850), выдающийся математик средневекового Востока

# Свойства алгоритма

- дискретность;
- результативность;
- точность;
- понятность;
- массовость;

# Программа - это

Последовательность команд, которые предназначены для реализации заданного алгоритма компьютером.

О.Ю. Заславская, И.В. Левченко

Совокупность машинных кодов и данных, загружаемых в оперативную память компьютера и обрабатываемая процессором.

В.В. Фаронов

# Этапы решения задачи на ЭВМ

1. Постановка задачи.
2. Формализация задачи.
3. Построение алгоритма.
4. Составление программы на языке программирования.
5. Отладка и тестирование программы.
6. Проведение расчетов и анализ полученных результатов.

# Транслятор

**Транслятор** (англ. *translator* - переводчик) – преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.

Трансляторы реализуются в виде **компиляторов** или **интерпретаторов**.

# Компилятор

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

# Интерпретатор

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

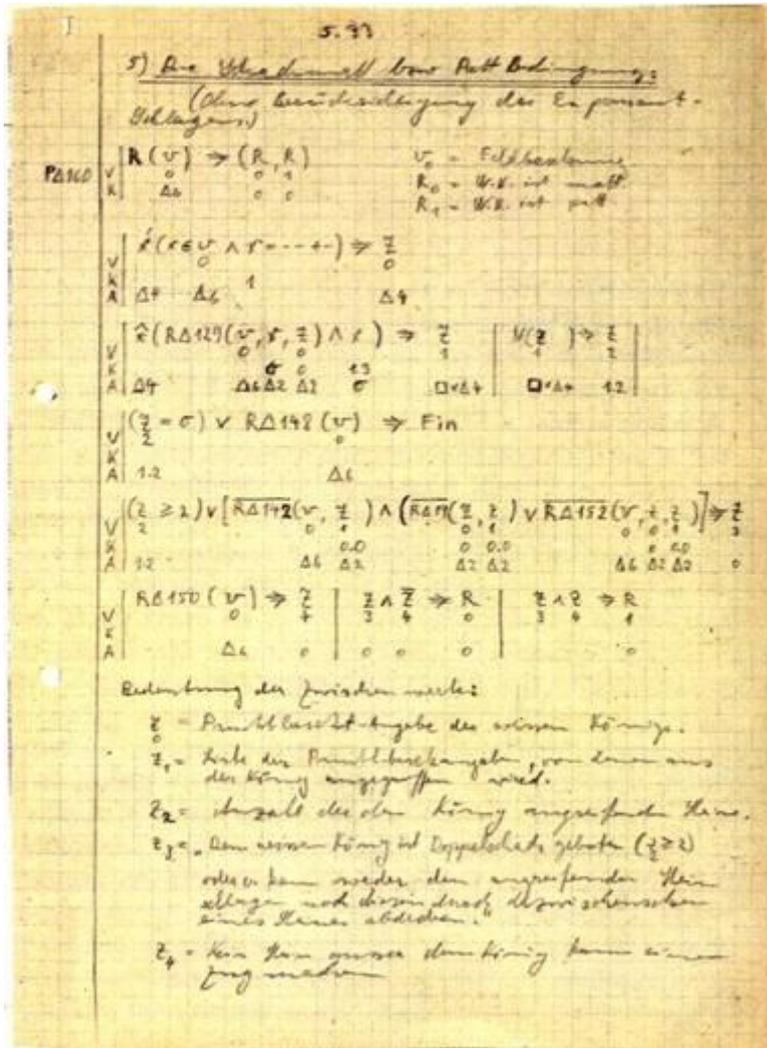
Откомпилированные программы работают быстрее, но интерпретируемые проще исправлять и изменять.

Паскаль реализуется с помощью компилятора,

Бейсик реализуется с помощью интерпретатора.

## 3.2. Языки и системы программирования

### Предыстория



Первая попытка создать высокоуровневый язык программирования принадлежит гениальному Конраду Цузе (конец 1940-х годов), разработавшему **Planalcalculus** (планировщик вычислений).

«Planalcalculus родился исключительно как результат теоретической работы, без всякой связи с тем, появится или нет в обозримом будущем машины, подходящие к программам на Planalcalculus».

Фрагмент рукописи Цузе с шахматной программой на языке Planalcalculus

## 3.2. Языки и системы программирования

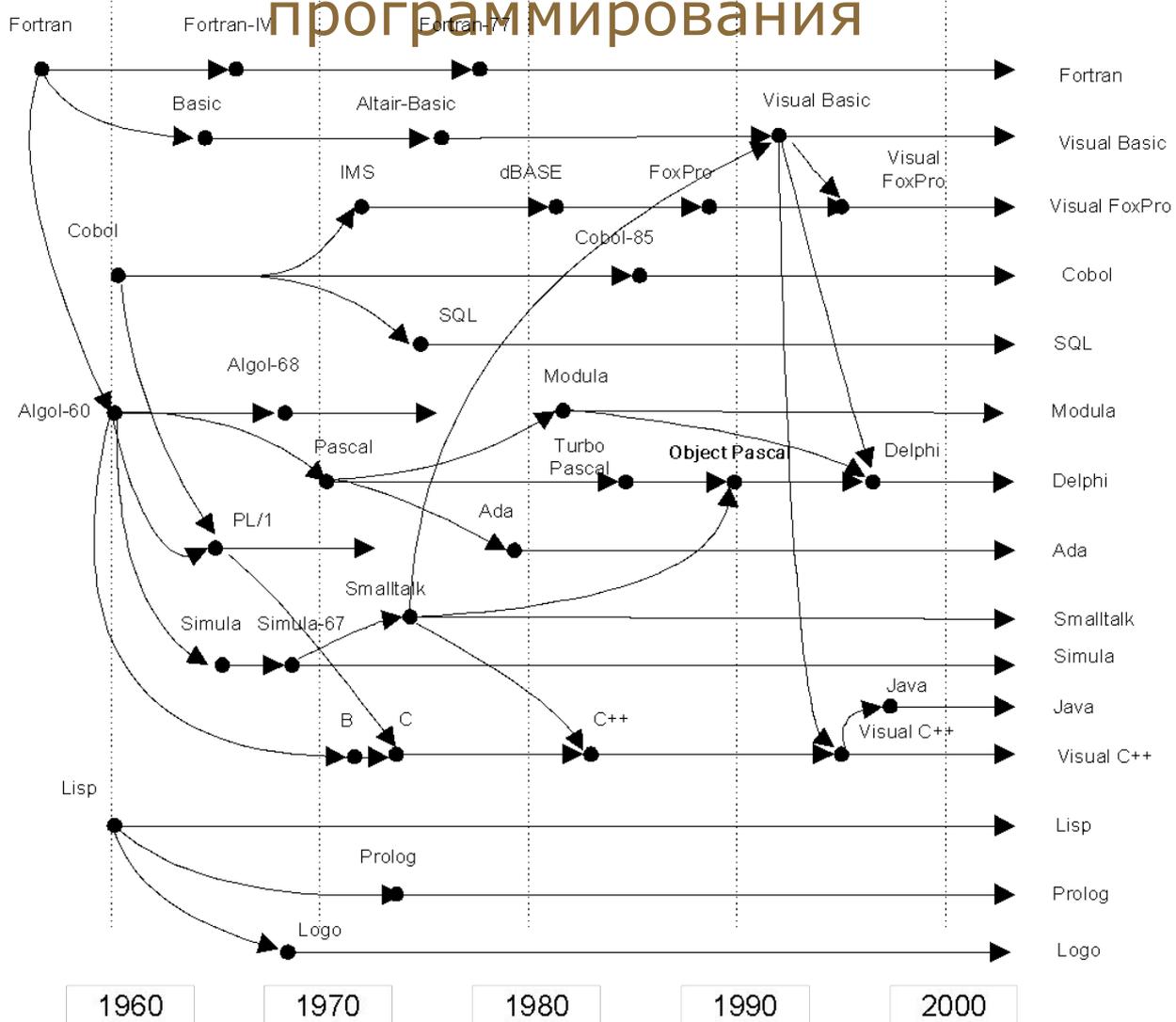
### Предыстория



Михаил Романович Шура-Бура и А.П. Ершов – создатели первых отечественных систем автоматизации программирования для ЭВМ «БЭСМ» и «Стрела»  
(1954-1956 годы)



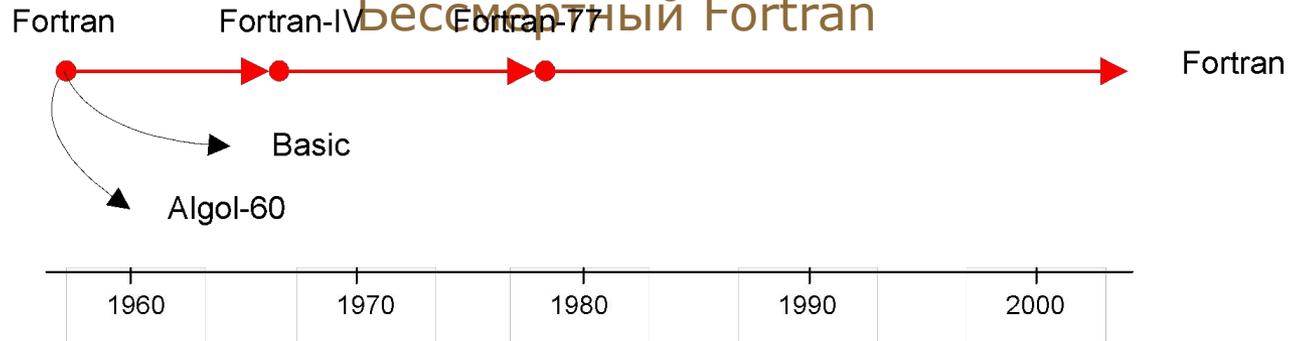
## 3.2. Языки и системы программирования



Родословная основных высокоуровневых языков программирования

## 3.2. Языки и системы программирования

### Бессмертный Fortran



**Fortran = FORMula TRANslator**

Первый высокоуровневый язык программирования Fortran был разработан в фирме IBM под руководством Джона Бэкуса (Backus, John; р. 1924).

Работа над языком началась в 1954 г., первая реализация для IBM 704 в выполнена в 1957 г.

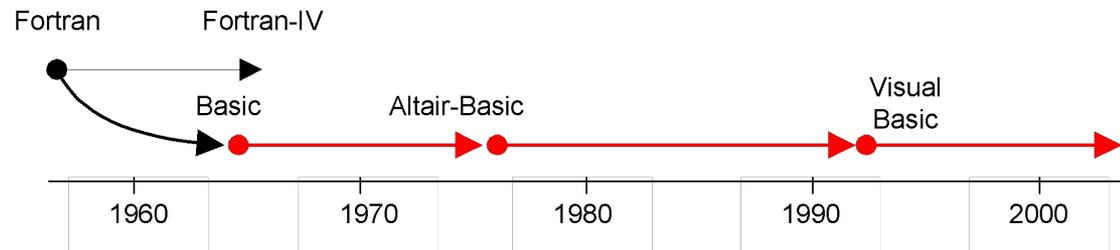
## 3.2. Языки и системы программирования Бессмертный Fortran

```
C      MAIN PROGRAM
101  FORMAT(208)
102  FORMAT(//'N=' ,15, 5X, 'R=' , 15
      1//6X, 'M' , 5X, PROB)
103  FORMAT(18, F14.10)
201  READ(1,101) N, IR
      WRITE(3,102) N, IR
      IF(N) 202, 202, 203
202  STOP
203  IF(IR) 202, 202, 204
204  M=0
      P=COMBF(N,M)*COMBF(IR-1,N-M-1)
      1/COMBF(N+IR-1,IR) ...
```

Фрагмент программы на языке Fortran

## 3.2. Языки и системы программирования

### Basic – язык для начинающих

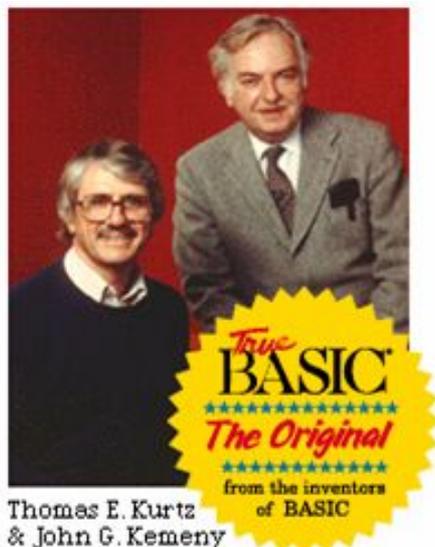


**BASIC = Beginners All-purpose Symbolic Instruction Code**

Язык Basic был разработан в 1964 г. в Дармутском колледже в г. Хановере (Dartmouth College, Hanover), штат Нью-Хемпшир

## 3.2. Языки и системы программирования

### Basic – язык для начинающих



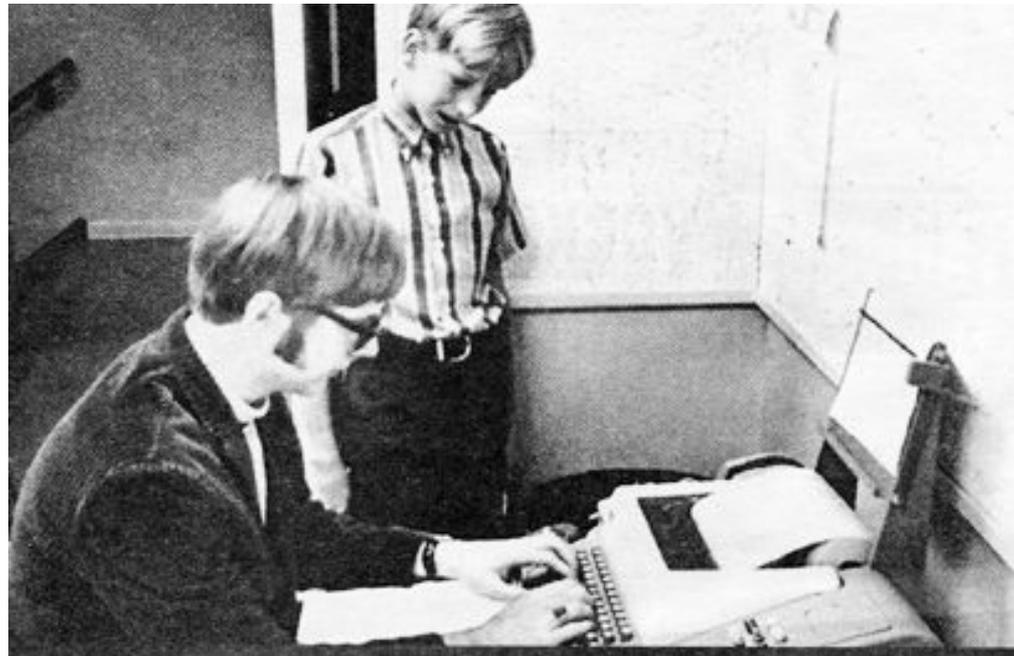
Авторы языка Basic.  
Стоит Джон Кемени  
(Kemeny, John G.; 1926-1993),  
сидит Томас Курц  
(Kurtz, Thomas E.; р. 1928)

```
10 dim A(5)
20 for i=1 to 5
30 input A(i)
40 next i
50 if i=5 then goto 140
60 if A(i)<=A(i+1) then goto 90
70 i=i+1
80 goto 130
90 z=A(i)
100 A(i)=A(i+1)
110 A(i+1)=z
120 i=1
130 goto 50
140 for i=1 to 5
150 print A(i)
160 next i
```

Простейшая  
программа на  
языке Basic

## 3.2. Языки и системы программирования

Basic – язык для начинающих



Будущие создатели Microsoft Пол Аллен (Allen, Paul; р. 1954) и Билл Гейтс (Gates, William; р. 1955) познакомились с Бэйсиком, работая в компьютерном классе школы в Сиэтле (снимок 1968 г.)

## 3.2. Языки и системы программирования

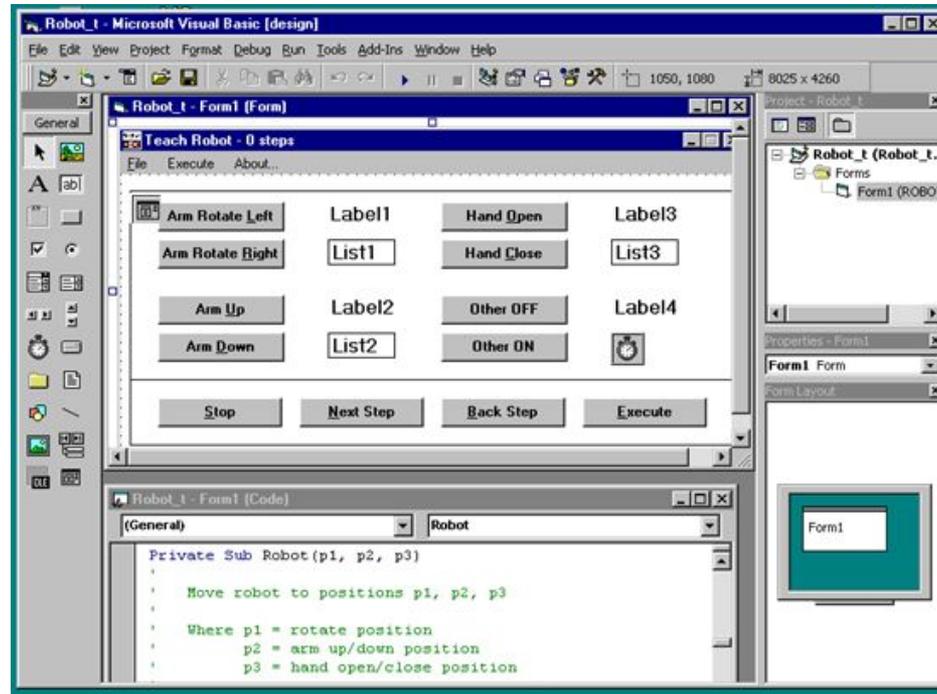
Basic – язык для начинающих

Начав с Бэйсика, компания Microsoft превратилась в крупнейшую софтверную империю, а Билл Гейтс – стал самым богатым человеком на планете



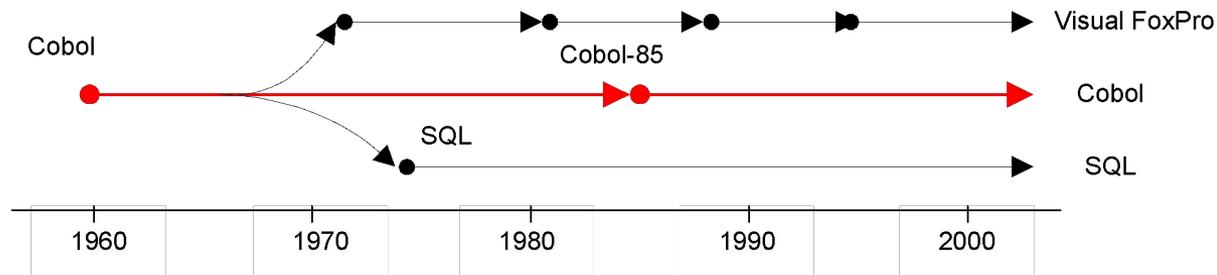
Штаб - квартира  
корпорации Microsoft в  
Редмонде (пригород  
Сиэтла)

## 3.2. Языки и системы программирования Basic – язык для начинающих



На протяжении нескольких десятилетий Visual Basic оставался фирменный языком компании Microsoft. В начале 1990-х годов он стал объектным и приобрел средства визуального проектирования

## 3.2. Языки и системы программирования Cobol – язык для бухгалтеров



### **COBOL = COMmon Business-Oriented Language**

На фото: разработчики языка Cobol у шуточного обелиска, присланного в их адрес в качестве намека на безнадежно медленную работу, способную похоронить саму идею. Справа внизу – Грейс Хоппер

## 3.2. Языки и системы программирования Cobol – язык для бухгалтеров

### Основные свойства языка Cobol:

- независимость программ от оборудования;
- независимость программ от данных;
- сложные структуры данных;
- синтаксис,  
к естественному английскому языку.

приближенный

## 3.2. Языки и системы программирования Cobol – язык для бухгалтеров

### 1010 IDENTIFICATION DIVISION.

1020 PROGRAM-ID "EXAMPLE".

### 1030 ENVIROMENT DIVISION.

1040 INPUT-OUTPUT SECTION.

1050 FILE-CONTROL.

1060 SELECT CD ASSIGN TO "SYS010" UNIT-RECORD 2540R.

1070 SELECT TT ASSIGN TO "SYS009" UTILITY 2400.

### 1080 DATA DIVISION.

1090 FILE SECTION.

1100 FD CD DATA RECORD IS C

1110 LABEL RECORDS ARE OMITTED.

1120 01 C.

1130 02 C1 PICTURE 9(4).

1140 02 C2 PICTURE 9.

1150 02 C3 PICTURE X(70).

...

**Программа на Коболе  
(начало)**

## 3.2. Языки и системы программирования Cobol – язык для бухгалтеров

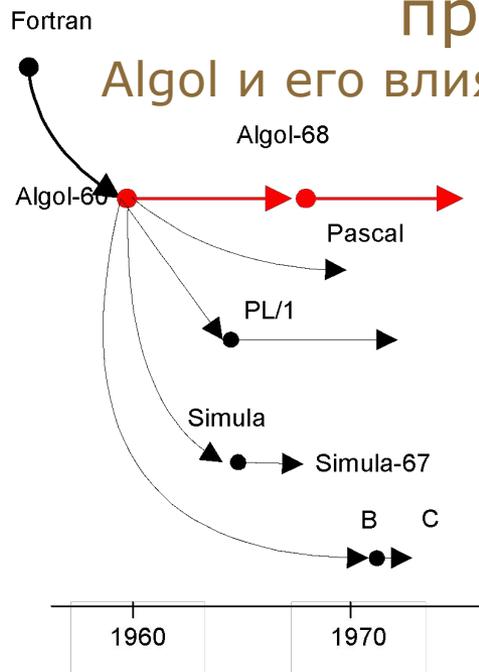
### 1290 PROCEDURE DIVISION.

1300 P1. OPEN INPUT CD, OUTPUT TT.  
1310 P2. READ CD, AT END GO TO P3.  
1320 MOVE C1 TO D1.  
1330 MOVE C2 TO D2.  
1340 MOVE C3 TO D3.  
1350 ADD C1, C2, GIVING D4.  
1360 WRITE T FROM D.  
1370 GO TO P2.  
1380 P3. CLOSE SD, TT.  
1390 STOP RUN.

**Программа на Коболе (окончание)**

## 3.2. Языки и системы программирования

### Algol и его влияние на языки программирования



#### **ALGOL = ALGORitmic Language**

В 1958 году в Цюрихе (Швейцария) состоялась международная конференция, предложившая проект нового универсального международного языка программирования Algol-58. В 1960 году на парижской конференции была принята окончательная версия под названием

Algol-60.

На снимке: участники парижской конференции голосуют за Алгол-60.

## 3.2. Языки и системы программирования

### Algol и его влияние на языки программирования

#### Основные свойства языка Algol-60:

- машинная независимость;
- формальный синтаксис;
- описание переменных и блочная структура;
- рекурсия

#### Нормальная форма Бэкуса-Наура (БНФ)

<цифра> ::= 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0

<целое без знака> ::= <цифра> | <цифра> <целое без знака>

## 3.2. Языки и системы программирования

Algol и его влияние на языки программирования

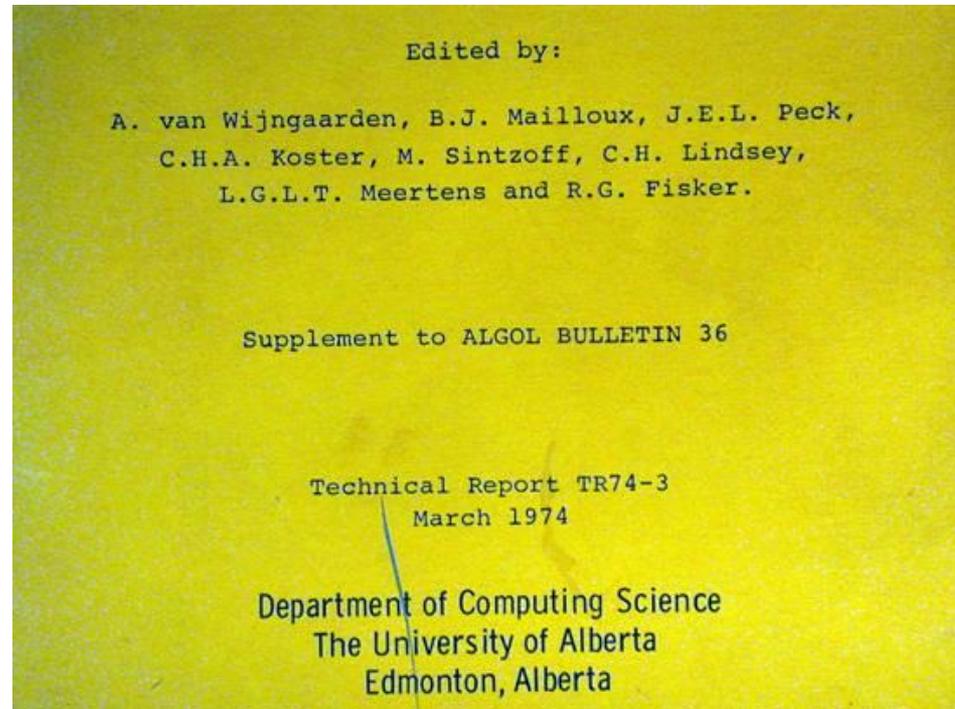
```
begin  
  integer i, n;  
  real s;  
  real array x[1:n];  
  s:=0;  
  for i:=1 step 1 to n do  
    s:=s+x[i];  
  s:=s/n  
end
```

Простейшая программа на Алголе-60, вычисляющая среднее арифметическое  $n$  чисел.

Синтаксис Алгола-60 сформировал стандарт для всех последующих языков программирования

## 3.2. Языки и системы программирования

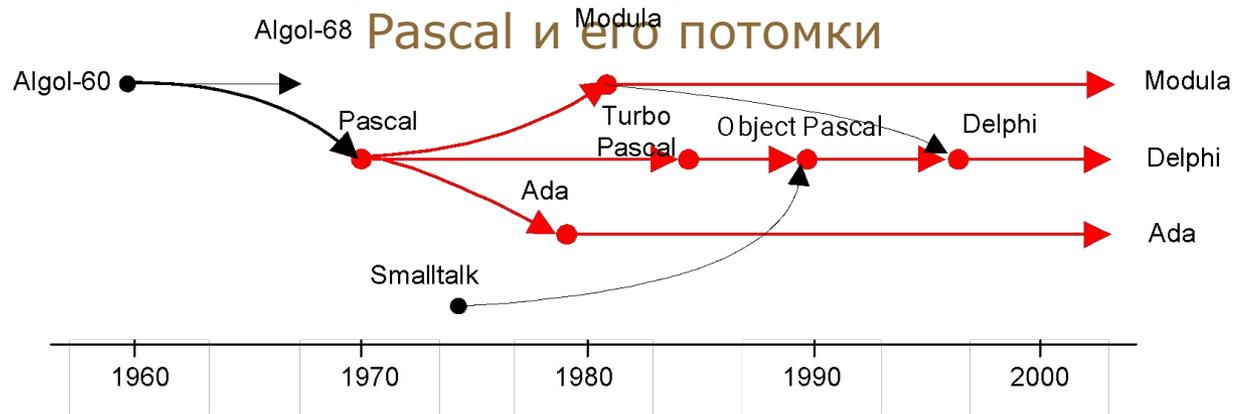
### Algol и его влияние на языки программирования



В результате многолетней переработки Алгола-60 комитетом [IFIP](#) появился язык [Алгол-68](#) (пересмотренное сообщение под ред. А. ван Вейнгаардена (A. van Wijngaarden) и др. опубликовано в 1975 г.)

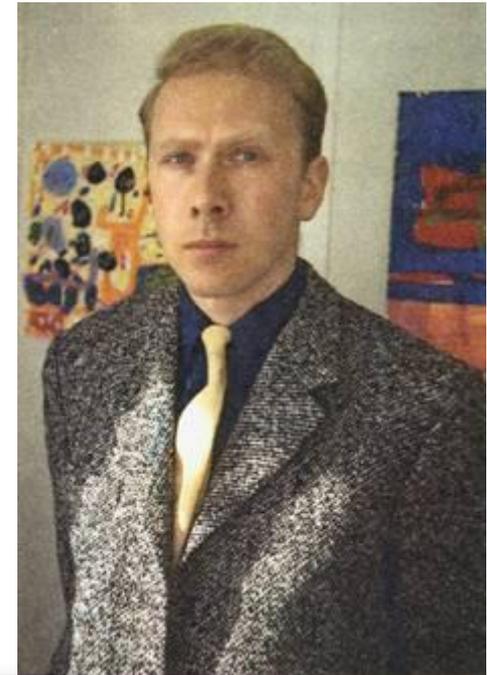
## 3.2. Языки и системы программирования

### Pascal и его потомки



Член комитета по Алголу-68 Никлаус Вирт (Wirth, Niklaus; р. 1934) был против принятия переусложненного стандарта.

В знак доказательства своей правоты он разработал в 1971 г. простой и ясный алголоподобный язык, предназначенный прежде всего для обучения студентов в Федеральном техническом университете в Швейцарии. В честь изобретателя первой вычислительной машины Вирт назвал язык **Паскалем**.



## 3.2. Языки и системы программирования Pascal и его потомки

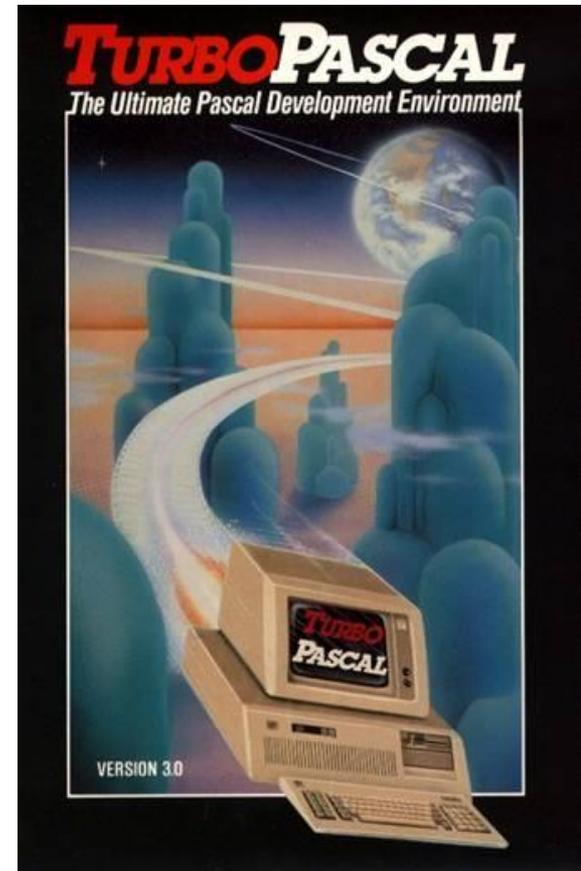
```
var
    i, n: integer;
    s: float;
    x: array[1..n] of real;
begin
    s:=0;
    for i:=1 to n do
        s:=s+x[i];
    s:=s/n
end.
```

Программа на Паскале, вычисляющая среднее  
арифметическое  $n$  чисел

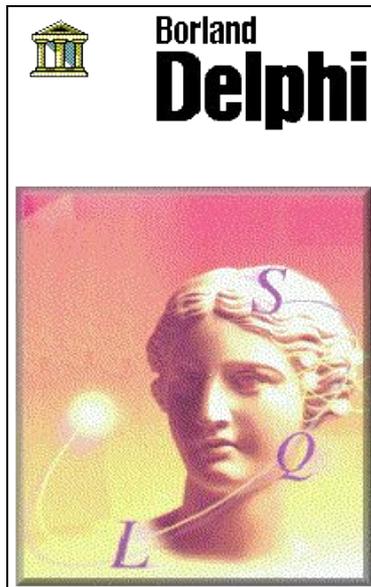
## 3.2. Языки и системы программирования Pascal и его потомки



Новую жизнь языку Pascal дал Филипп Кан (Kahn, Philippe; р. 1938) – создатель компилятора Turbo Pascal для IBM PC и основатель компании Borland (1984 г.)

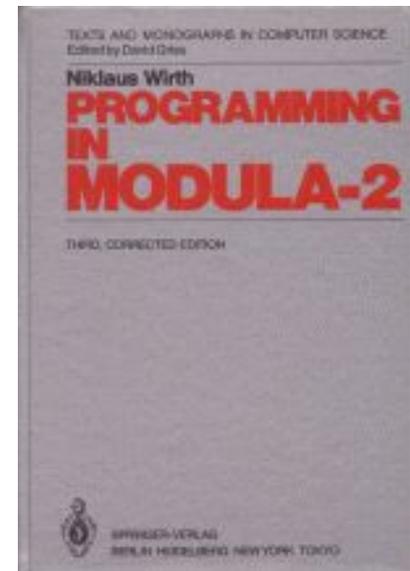


## 3.2. Языки и системы программирования Pascal и его потомки



Среда разработки Delphi фирмы Borland объединила передовые достижения технологии программирования: объектное расширение языка Pascal, визуально-событийное проектирование, модульное структурирование и отдельная компиляция.

В отличие от учебного Паскаля, язык программирования Modula-2, предложенные Никлаусом Виртом, изначально предназначался для профессионального применения



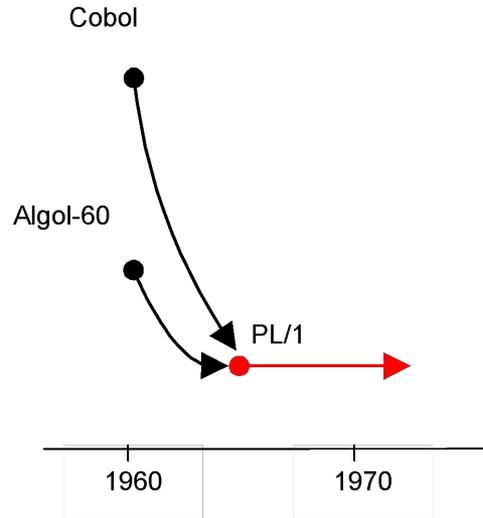
## 3.2. Языки и системы программирования Pascal и его потомки

В 1975 году Министерство обороны США приняло решение разработать стандартный язык для программирования сложных и ответственных военных приложений. Был объявлен широкий международный конкурс, в котором приняли участие 15 групп разработчиков. В результате нескольких туров в мае 1979 года выявился победитель — французская фирма С.И.И., руководитель проекта Жан Ихбиа (Ichbiah, Jean).

Снимок сделан на II конференции по истории языков программирования, 1993 г.



## 3.2. Языки и системы



### PL/1 = Programming Language One

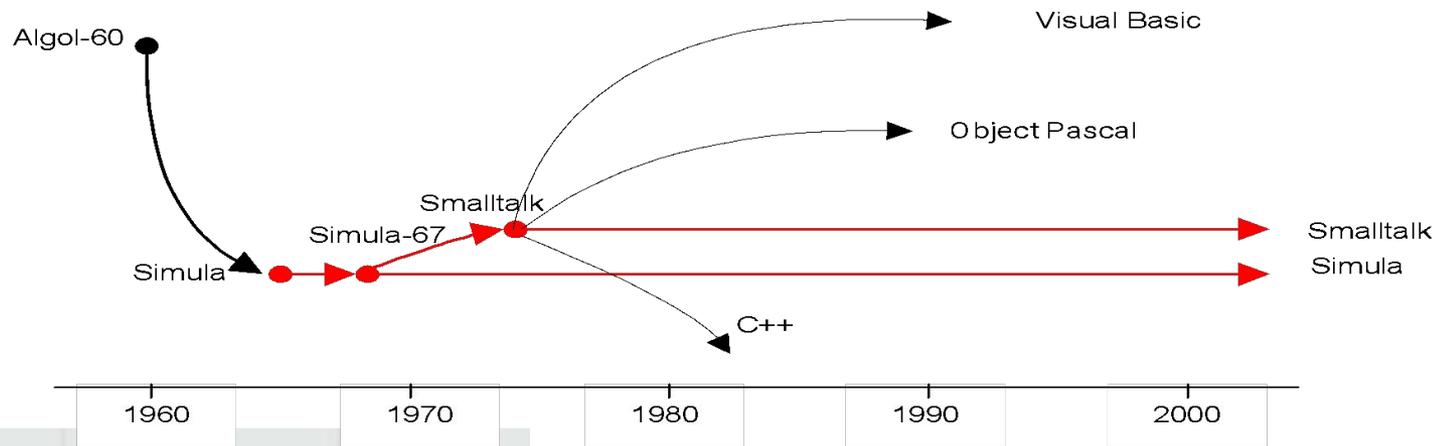
Язык PL/1 был частью амбициозного проекта IBM S/360, он создавался в спешке и представлял собой механическую смесь идей из многих языков. Критики сравнивали его с елкой со множеством украшений.

```
EXAMPLE: PROCEDURE OPTIONS (MAIN);  
ON ENDFILE (SYSIN) GO TO ENDING;
```

```
P1:   GET LIST (A, B, C);  
      D = B*B - 4*A*C;  
      E = -B/(A+A);  
      IF D<0 THEN DO;  
        X1, X2 = E;  
        Y1 = SQRT(-D)/(A+A);  
      END;  
      ELSE DO;  
        R = SQRT(D)/(A+A);  
      ...  
      Y1 = 0;  
      END;  
      Y2 = -Y1;  
      PUT LIST (X1, Y1, X2, Y2);  
      GO TO P1;
```

```
ENDING;;  
END EXAMPLE;
```

## 3.2. Языки и системы программирования Simula и Smalltalk – революция в программировании



### Simula = SIMULAlation

За разработку языка Simula Кристен Нигорд (Nygaard, Kristen; 1926-2002), на снимке слева, и Оле-Йохан Дал (Dahl, Ole-Johan; 1931-2002) были удостоены высшей награды компьютерного сообщества – медали Тьюринга

## 3.2. Языки и системы программирования Simula и Smalltalk – революция в программировании

```
|a|  
a := Array new: 5.  
1 to: 5 do: [:i | a at: i put:  
  (Prompter prompt: 'Введите элемент массива') asNumber].  
a := a asSortedCollection.  
a do: [:i | Transcript putAll: i printString].
```

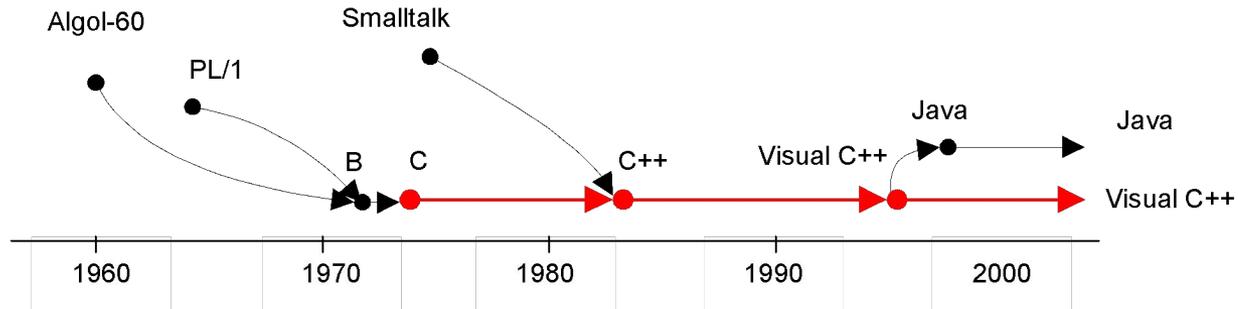
**Простейшая программа  
на Smalltalk, вычисляющая  
среднее арифметическое  
пяти чисел**



**Алан Кей**

## 3.2. Языки и системы программирования

### С – язык для профессионалов



Язык **Си (C)** был создан Деннисом Ричи (Ritchie, Dennis M.; р. 1941) в 1973 году в Bell Labs в ходе разработки операционной системы UNIX. Он развивал язык **Би (B)**, который основывался на созданном в Кембриджском университете языке **BCPL** (от **Basic Combined Programming Language**), который в свою очередь был потомком Алгола-60

## 3.2. Языки и системы программирования

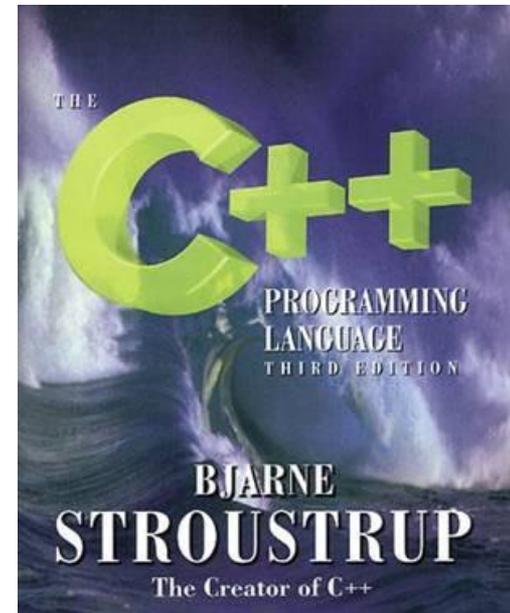
### C – язык для профессионалов

```
float A[5];
for(int i=0;i<5;i++)scanf("%f",&A[i]);
i=0;
while(i<4){
    if(A[i]<=A[i+1])i++;
    else{
        z=A[i];
        A[i]=A[i+1];
        A[i+1]:=z;
        i=0;
    }
};
for(i=0;i<5;i++)printf("%f\n",A[i]);
```

Текст на языке C отличается лаконичностью

## 3.2. Языки и системы программирования

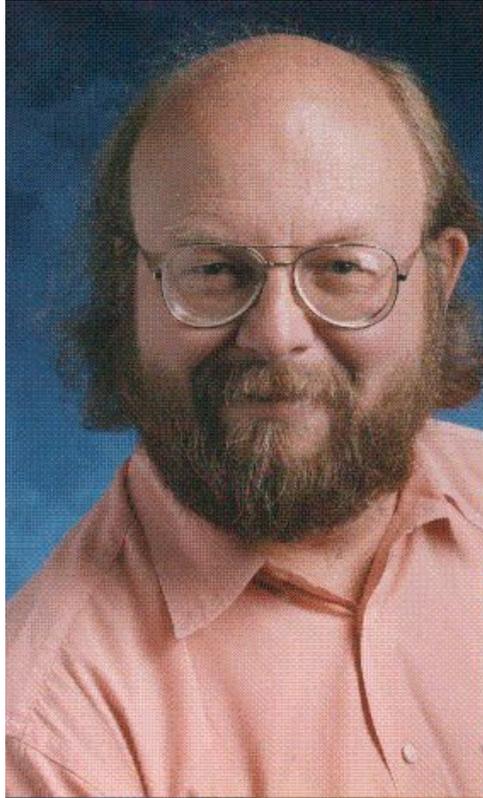
С – язык для профессионалов



Бьярн Страуструп (Stroustrup, Bjarne; р. 1950) ввел в язык С объекты и превратил его в C++

## 3.2. Языки и системы программирования

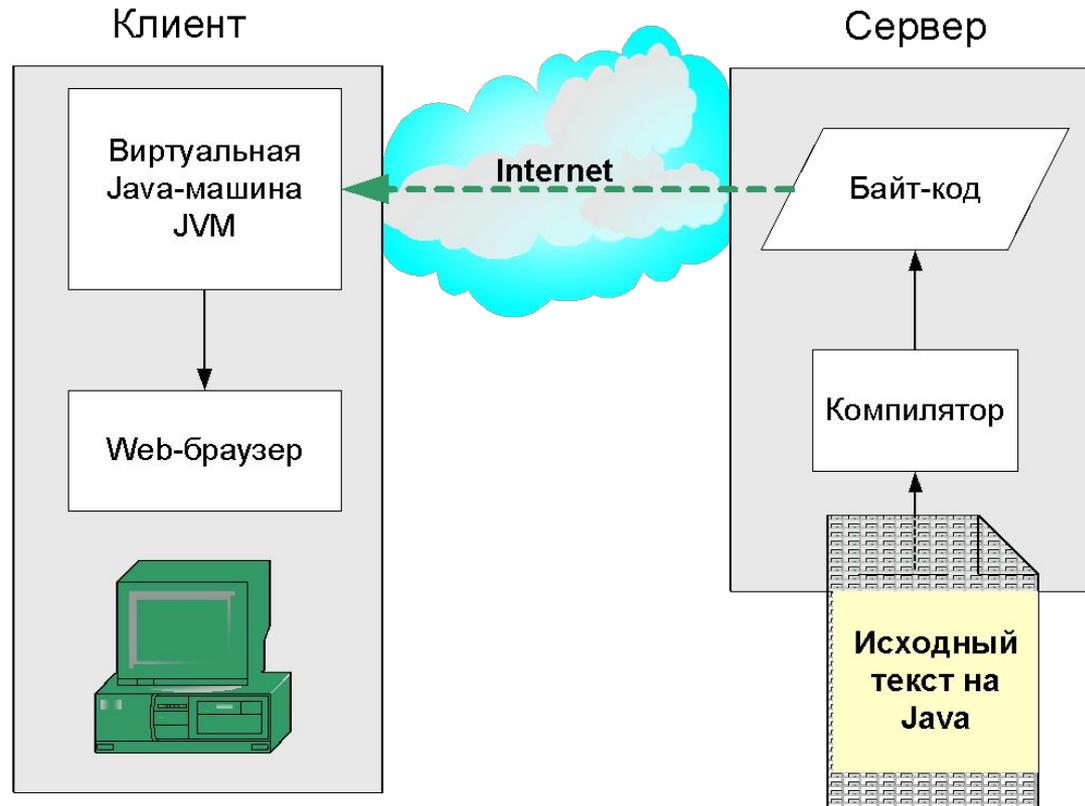
Java – дитя интернета



В 1995 г. фирма Sun Microsystems представила язык **Java** для программирования в интернете. Он возник в ходе реализации проекта **Oak** («Дуб»), целью которого было создание системы программирования бытовых микропроцессорных устройств.

Джеймс Гослинг (Gosling, James) – автор Java.

## 3.2. Языки и системы программирования Java – дитя интернета



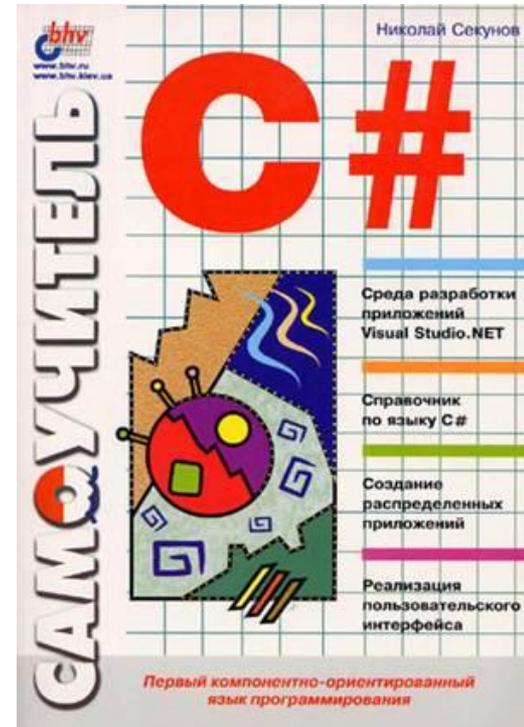
Java - ТЕХНОЛОГИЯ

## 3.2. Языки и системы программирования

### Java – дитя интернета

```
class test
{
  int i, n;
  float s;
  float x[n];
  public static void main( String
args[] )
  {
    n = 10;
    s = 0;
    for( i=1; i<=n; i++)
    {
      s = s + x[i-1];
      s = s / n;
    }
  }
}
```

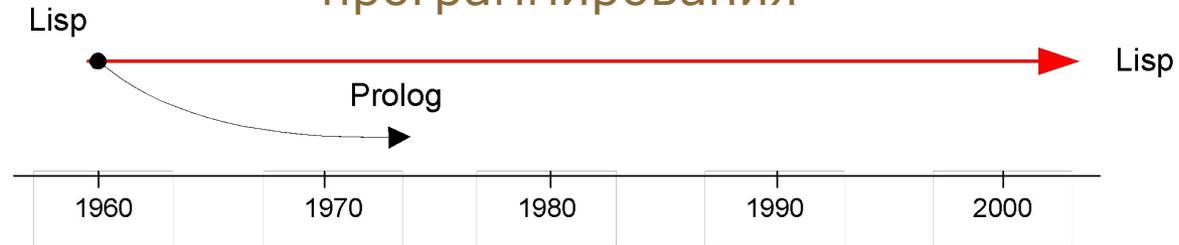
Язык Java основан на C++



В качестве альтернативы Java  
корпорация Microsoft предложила язык  
С# (Си-шарп)

## 3.2. Языки и системы программирования

Долгожитель Lisp – инструмент функционального  
программирования



Дж. Маккарти и А.П. Ершов  
Снимок 1975 г.

### **Lisp = LISt Processing**

Язык Lisp создан в 1960 году Джоном Маккарти (McCarthy, John; р. 1927 ) в Массачусетском технологическом институте на теоретическом фундаменте лямбда-исчисления, предложенного еще в 1930 году известным американским логиком Алонзо Черчем.

## 3.2. Языки и системы программирования

### Долгожитель Lisp – инструмент функционального программирования

```
(setq L `(8 5 13 11 10))  
(defun sum (L)  
  (cond ((null L) '0)  
        (t (add (car L) (sum (cdr L)))))  
  )  
)  
(div (sum L) '5)
```

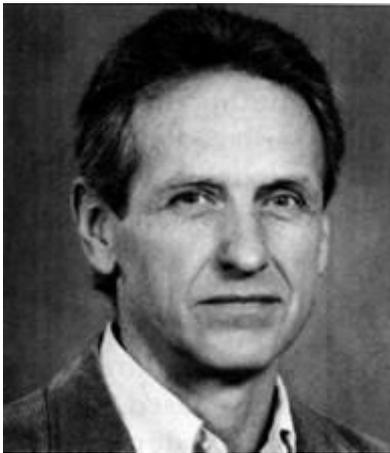
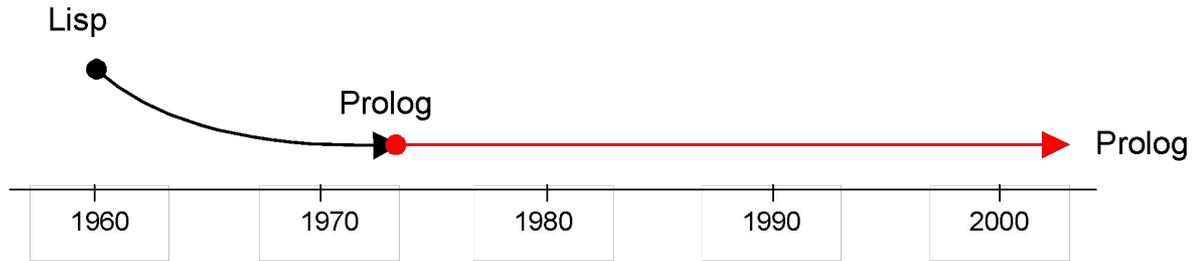
#### **Примитивы:**

cond — условная функция, проверяющая с помощью функции null пустоту списка;  
add — суммирование аргументов;  
car — извлечение первого элемента из списка;  
cdr — извлечение остатка списка (без первого элемента).

Программа на Lisp имеет специфический вид из-за обилия скобок. За это студенты прозвали его «**Lots of Infuriating & Silly Parenthesis**» - «Множество раздражающих и глупых скобок»

## 3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения



**Prolog = PROgramming for LOGic**

Теоретические основы языка были разработаны Робертом Ковальским (Kowalski, Robert) в Эдинбургском университете (Шотландия) в конце 1960-х годов

Первая практическая реализация языка осуществлена Аленом Кольмари (Colmerauer, Alain) в Марсельском университете (Франция) в 1972 г.



## 3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения

### Факты:

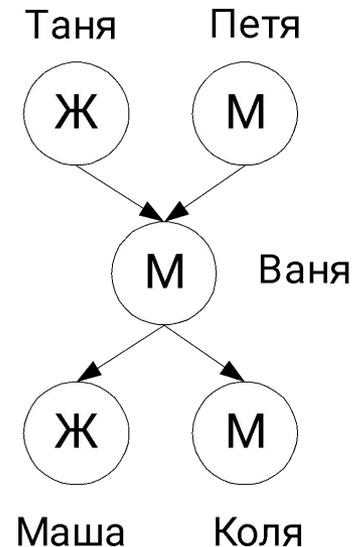
муж (петя), муж (ваня),  
муж (коля), жен (таня), жен (маша),  
мать (ваня, таня), отец (ваня, петя),  
отец (маша, ваня), отец (коля, ваня).

### Правила вывода:

родитель (X, Y) :— отец (X, Y)  
родитель (X, Y) :— мать (X, Y)  
дед (X, Y) :— родитель (X, Z), отец (Z, Y)  
брат (X, Y) :— муж (Y), родитель (X, Z),  
родитель (Y, Z), X<>Y

### Примеры диалога:

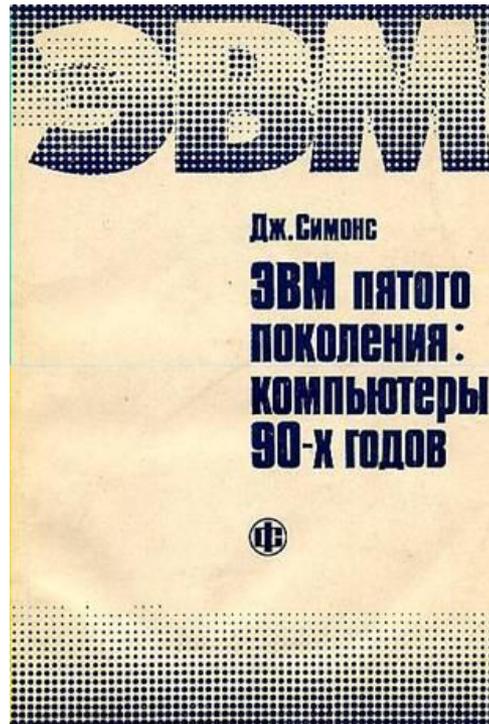
GOAL> дед (коля, X) *Кто дед Коли?*  
X = Петя  
GOAL> брат (маша, X) *Кто брат Маши?*  
X = Коля



Описание предметной области семейных отношений на языке Prolog

## 3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения



Проект ЭВМ V поколения – японский вызов мировой компьютерной индустрии, брошенный в начале 1980-х годов

## 3.2. Языки и системы программирования

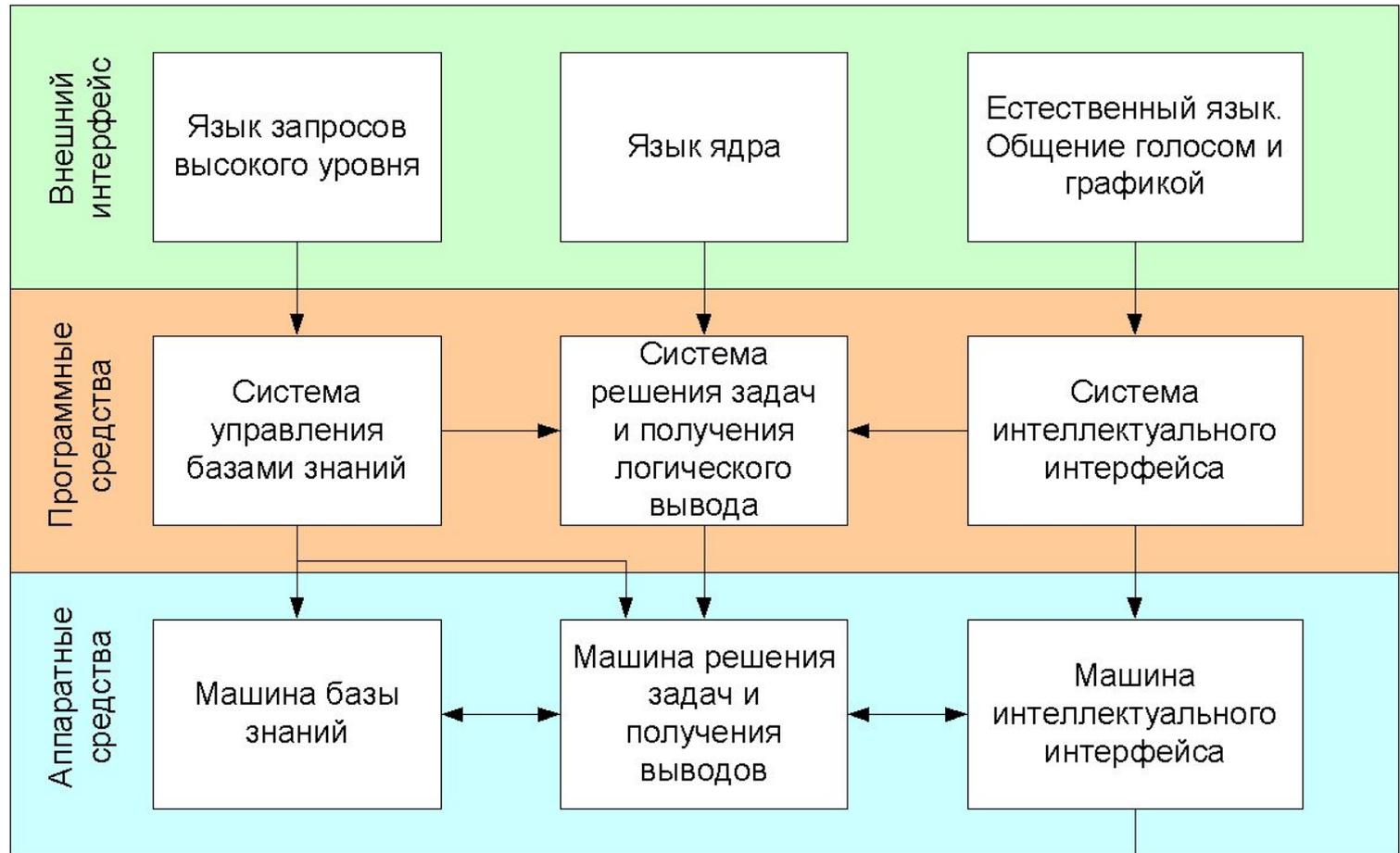
Prolog – несостоявшаяся мечта ЭВМ V поколения

### **Концептуальные отличия ЭВМ V поколения:**

- новая технология производства микросхем, знаменующая переход от кремния к арсениду галлия, и дающая возможность на порядок повысить быстродействие основных логических элементов;
- новая архитектура (не фон-неймановская);
- новые способы ввода-вывода информации — распознавание и синтез речи и образов;
- отказ от традиционных алгоритмических языков программирования (Фортран, Алгол и т. п.) в пользу декларативных;
- ориентация на задачи искусственного интеллекта с автоматическим поиском решения на основе логического вывода.

## 3.2. Языки и системы программирования

Prolog – несостоявшаяся мечта ЭВМ V поколения



Структура ЭВМ V поколения

К сетям ЭВМ  
V поколения

## 3.2. Языки и системы программирования

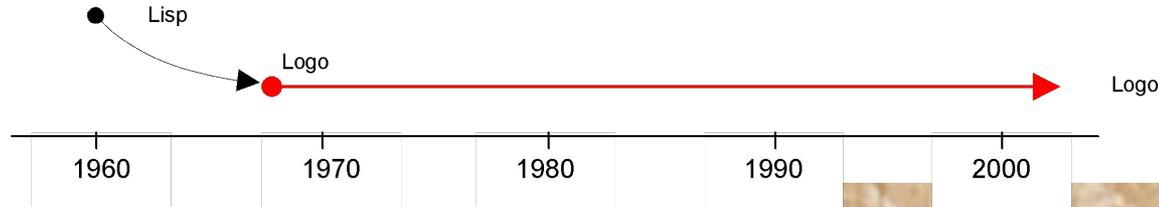
Prolog – несостоявшаяся мечта ЭВМ V поколения



В качестве основного языка ЭВМ V поколения  
предполагалось использовать Prolog

## 3.2. Языки и системы программирования

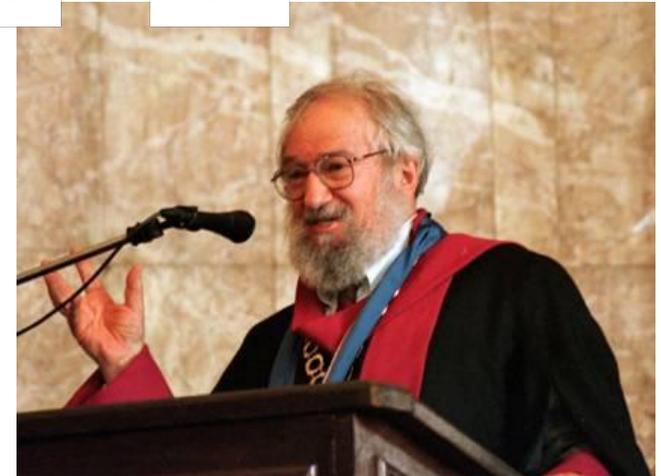
### Logo – язык для самых маленьких



Язык **Logo**, изобретен в 1967 г. в MIT выдающимся математиком и педагогом Сеймуром Пейпертом (Papert, Seymour; p. 1928).

Пейперт в 1958-1963 годах работал в Женеве у знаменитого психолога Жана Пиаже (Piaget, Jean), где занимался детьми и природой их мышления.

Идейной основой Logo является язык Lisp



На фото: Сеймур Пейперт получает степень почетного доктора Софийского университета (1999 г.)

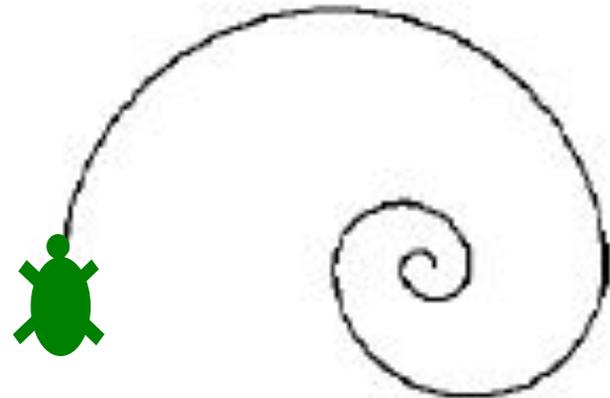
## 3.2. Языки и системы программирования

Logo – язык для самых маленьких

```
это дуга :шаг :число_шагов  
повтори :число_шагов  
  [вперед :шаг направо 10]  
Конец
```

Цикл

```
это спираль :шаг  
если :шаг < 1 [стоп]  
дуга :шаг 18  
спираль :шаг / 2  
конец
```



Процедура с параметром

---

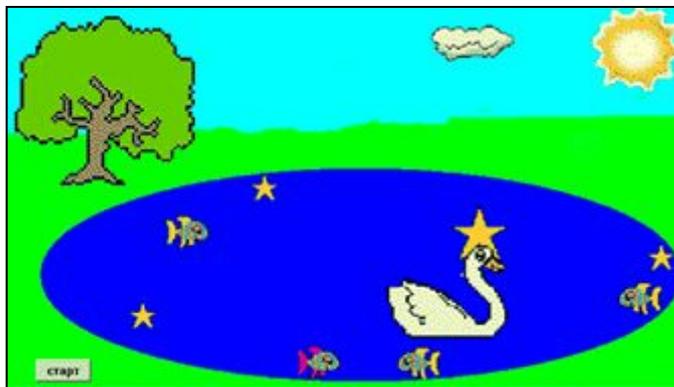
Рекурсия

Программа на Logo управляет черепашкой, оставляющей видимый след. С помощью зрительных образов интерпретируются все базовые структуры программирования

## 3.2. Языки и системы программирования

### Logo – язык для самых маленьких

Концепция Logo учитывает детскую психологию и рассчитана на обучение школьников, начиная с младших классов



Новейшие реализации Logo используют принципы объектно-ориентированного программирования. В программе Юли Гладких, 9 лет, черепашка в форме лебедя плавает по озеру.

## 3.2. Языки и системы программирования

### **Основные парадигмы программирования:**

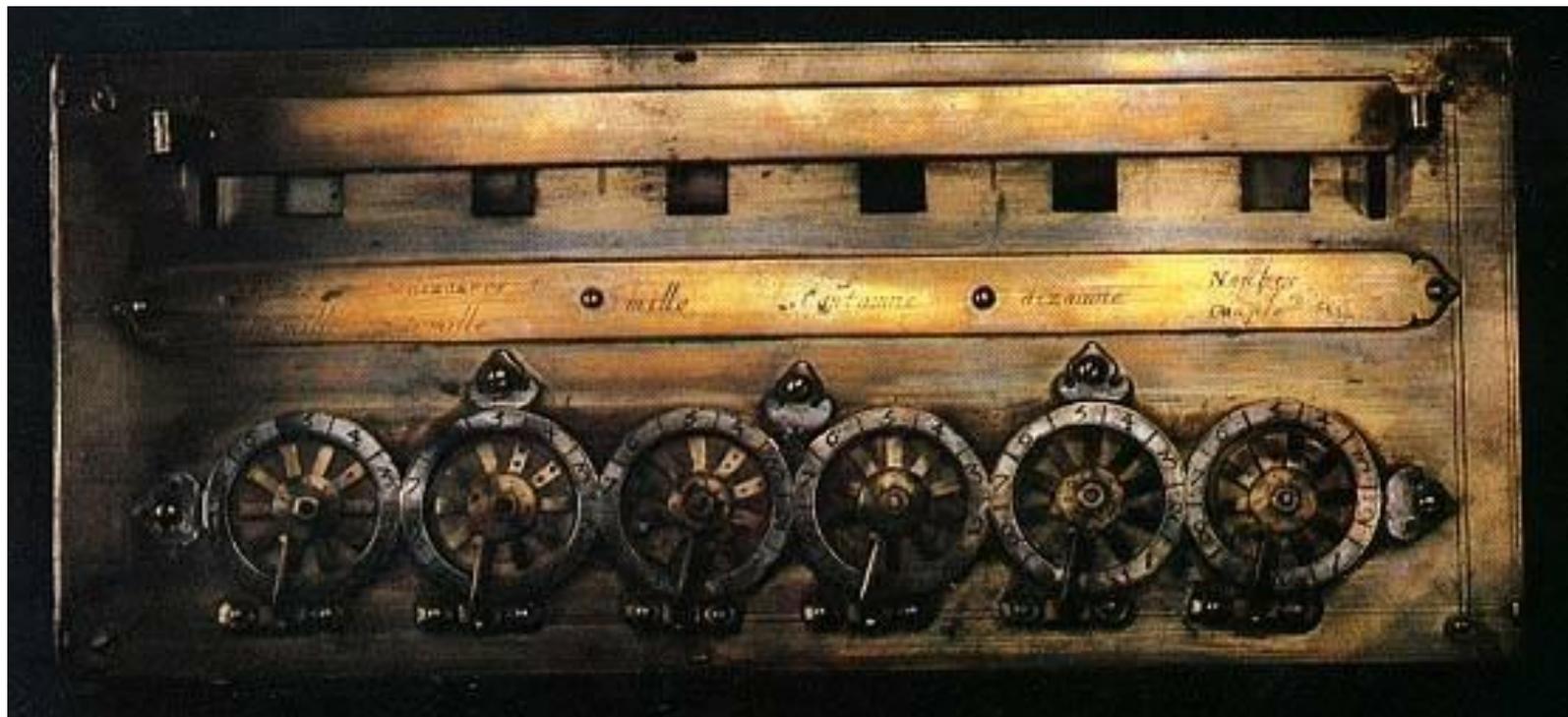
- процедурное программирование (Fortran, Basic, Cobol, Algol, Pascal, Ada, C, Logo, FoxPro);
- объектно-ориентированное программирование (Simula, Smalltalk, Object Pascal, C++, Java, C#);
- визуально-событийное программирование (Visual Basic, Delphi, Visual C++, Visual Java, Visual FoxPro);
- функциональное программирование (Lisp);
- логическое программирование (Prolog).

## 1.4. Суммирующая машина Паскаля



Блез Паскаль (Pascal, Blaise; 1623-1662)

## 1.4. Суммирующая машина Паскаля



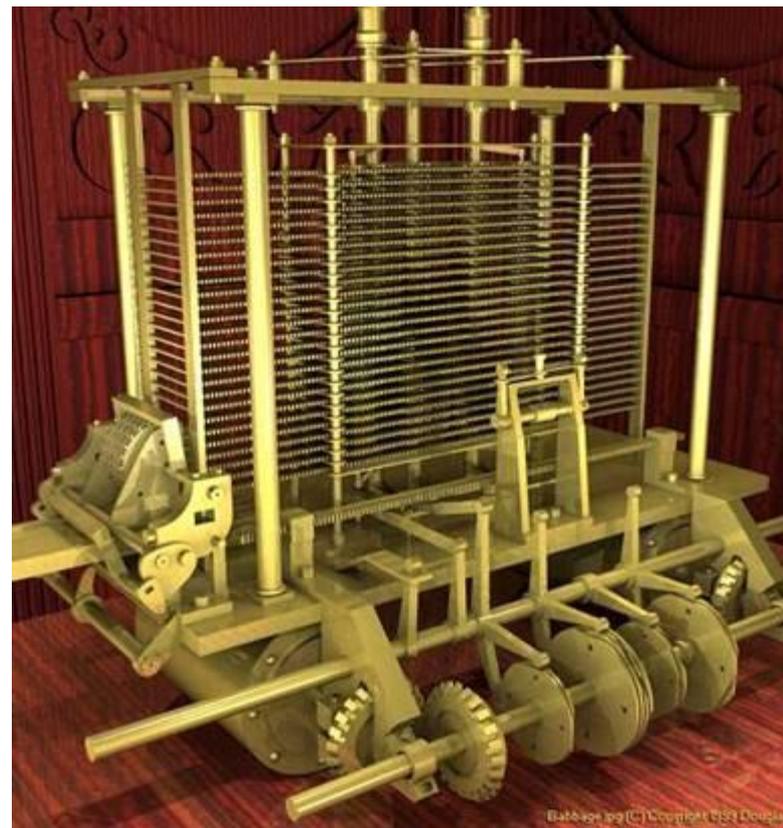
Паскалина (1642 г.) Вид спереди

1.6. Принцип программного управления.  
Вычислительные машины Бэббиджа



Чарльз Бэббидж (Charles Babbage, 1791-1871)

## 1.6. Принцип программного управления. Вычислительные машины Бэббиджа



Вычислительные машины  
Бэббиджа 1820-1832 г.  
(фрагменты, реконструкция)

## 1.6. Принцип программного управления. Вычислительные машины Бэббиджа



Первый в истории программист графиня Ада Лавлейс, урожденная Байрон  
(Lovelace, Ada Augusta; 1815-1852)

## 1.8. Сложные электромеханические и релейные машины – предвестники ЭВМ



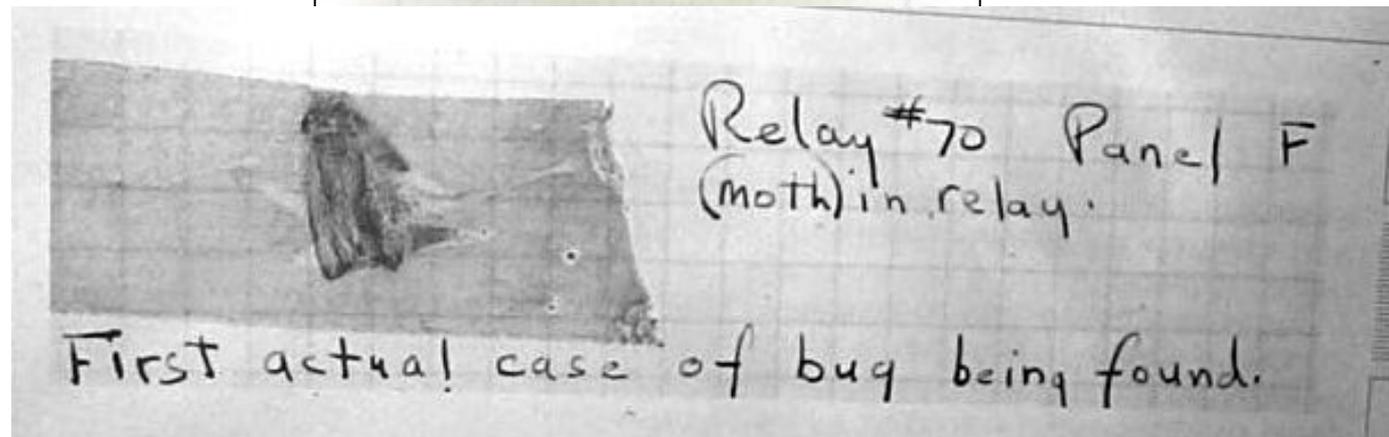
Младший лейтенант...



...адмирал

Руководитель группы программистов Mark-1  
Грейс Хоппер (Hopper, Grace; 1906-1992)

## 1.8. Сложные электромеханические и релейные машины — предвестники ЭВМ



Запись 9.09.45 в рабочем журнале компьютера Mark: «Реле #70 панель F. Мотылек в реле.

Первый достоверный случай обнаружения насекомого»