

Введение

Уважаемый читатель! Прежде всего предупредим о том, чем эта книга не является. Книга, которую вы держите в руках, — не учебник по одному из популярных языков программирования. Прочитав ее, вы не станете профессионалом в С# или Java, использующим полученные навыки для поиска наиболее выгодных предложений на рынке труда. Книга не предназначена также для обучения программированию как таковому — методам написания эффективных алгоритмов, построения пользовательских интерфейсов, доступа к базам данных и пр., хотя косвенно эти темы в ней освещаются. Цель книги другая. В ней делается попытка взглянуть с разных точек зрения на интереснейшую предметную область — языки программирования по состоянию на 2013 год. Если можно так выразиться, применить стереоскопический взгляд.

В настоящее время насчитывается около восьми тысяч языков программирования, причем одни не похожи на другие. Вы сможете найти в книге упоминания и примеры из языков, о которых широкие круги программистской общественности либо мало осведомлены вообще, либо забыли, но которые при этом не перестают быть интересными и даже поучительными. Автор будет считать свою задачу выполненной, если ему удастся расширить кругозор читателя, а уж если читатель сможет извлечь из книги практическую пользу — с лихвой (в частности, книга содержит введение в программирование на Лиспе и Прологе, которые, по убеждению автора, сейчас незаслуженно редко применяются в нашей стране в практических проектах).

Во введении можно долго рассуждать об исторической ретроспективе предмета, его связи со смежными дисциплинами, значимости для жизни современного общества и т. д. Все эти аспекты важны, но, как представляется автору, в самом начале лучше погрузить читателя в суть того, что ему предстоит изучать. Получить представление о предмете может помочь набор примеров — программ на различных

семантически эквивалентных языках (подробнее о том, что такое синтаксис и семантика, будет рассказано далее), которые делают одно и то же, будучи исполненными на ЭВМ, оснащенной соответствующими средствами. Выглядят программы на этих языках по-разному. Следуя примеру Лоуренса Теслера [1, с. 76], используем для иллюстрации не банальный пример «Здравствуй, мир!», а программу, имеющую (условно) прикладное значение — подсчитывающую сумму нечетных чисел, входящих в последовательность целых чисел. Итак, перейдем к примерам.

Программа на языке BASIC:

```
10 DIM T(100)
20 INPUT N
30 FOR I=1 TO N
40 INPUT T(I)
50 NEXT I
60 GOSUB 110
70 PRINT "СУММА НЕЧЕТНЫХ=" S
80 GOTO 200
110 REM подпрограмма
120 S=0
130 FOR I=1 TO N
140 IF NOT ODD(T(I)) THEN GOTO 160
150 S=S+T(I)
160 NEXT I
170 RETURN
200 END
```

Программа на языке COBOL:

```
DATA DIVISION.
WORKING-STORAGE SECTION.
01 NUMERIC-VARIABLES USAGE IS COMPUTATIONAL.
    02 NUMBERS PICTURE 9999 OCCURS 100 TIMES INDEXED BY I.
    02 N PICTURE 999.
    02 SUM PICTURE 99999.
    02 HALFC PICTURE 9999.
    02 MODC PICTURE 9.
PROCEDURE DIVISION.
EXAMPLE
    MOVE 23 TO NUMBERS (1)
```

```

MOVE 34 TO NUMBERS (2)
MOVE 7 TO NUMBERS (3)
MOVE 9 TO NUMBERS (4)
MOVE 11 TO NUMBERS (5)
MOVE 5 TO N
PERFORM SUMNECH.

```

SUMNECH

```

MOVE 0 TO SUM
PERFORM ANALIS-1 VARYING I FROM 1 BY 1 UNTIL I>N

```

ANALIS-1

```

DIVIDE 2 INTO NUMBERS (I) GIVING HALFC REMAINDER MODC
IF MODC IS EQUAL TO 1 ADD NUMBERS(I) TO SUM.

```

Программа на языке APL:

```

▽ СУМ←СУМНЕЧЕТ ЧИСЛА
▽ СУМ←+(2|ЧИСЛА)/ЧИСЛА

```

вызов: СУМНЕЧЕТ 2 3 3 4 7 9

Программа на языке Форт:

```

: СУМНЕЧЕТ
0 SWAP 0
DO
  SWAP DUP 2 MOD
  IF +
  ELSE DROP
  THEN
LOOP

```

Вызов: 2 3 3 4 7 9 СУМНЕЧЕТ

Программа на языке Лисп:

```

(DEFUN СУМНЕЧЕТ(ЧИСЛА)
  (COND
    ((NULL ЧИСЛА) 0)
    ((ODD (CAR ЧИСЛА)) (+ (CAR ЧИСЛА)(СУМНЕЧЕТ(CDR ЧИСЛА))))
    (T (СУМНЕЧЕТ (CDR ЧИСЛА)))))

```

Программа на языке ассемблера микропроцессора Motorola 68000:

```

СУМНЕЧЕТ MOVE.L (A7)+,A2 Адрес возврата из стека в A2
          MOVE.L (A7)+,A1 Адрес первого числа => A1
          MOVE.W (A7)+,D1 Заслать n в D1
          CLR.W D2 Обнулить D2
          JMP СЧЕТЧИК Перейти в конец цикла n=0?
ЦИКЛ     BTST 0,1(A1) Если число по адресу A1 четное...
          BEQ.S СЛЕД ...перейти к метке СЛЕД
          ADD.W (A1),D2 ...иначе прибавить число к D2
СЛЕД     ADDQ.W #2,A1 Взять в A1 адрес следующего числа
СЧЕТЧИК DBF D1,ЦИКЛ Уменьшить D1,пока не -1 => на ЦИКЛ
          MOVE.W D2,-(A7) Занести сумму нечетных в стек
          JMP (A2) Перейти по адресу возврата
  
```

Программа на языке Пролог:

```
sumnech([X|Xs],S):-odd(X),sumnech(Xs,S1),S is S1+X.
```

```
sumnech([X|Xs],S):-sumnech(Xs,S),\+ odd(X).
```

```
sumnech([],0).
```

```
odd(X):-integer(X),X rem 2 =:= 1.
```

Программа на визуальном языке программирования российской разработки HiAsm (рис. 1).

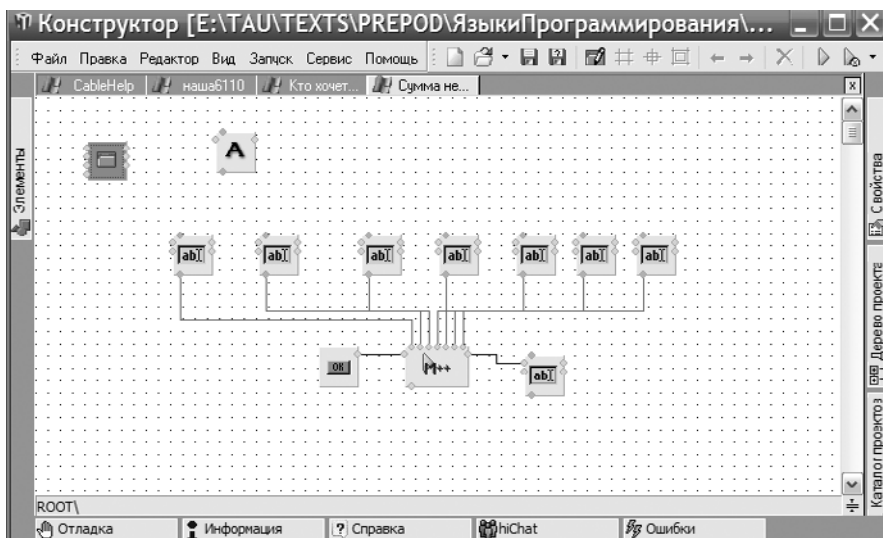


Рис. 1

Автор надеется: читатель не без интереса просмотрел приведенные программы и обратил внимание на то, что они заметно различаются по длине, стилю и внешнему виду вообще...

Целью настоящей книги является дать читателю представление о «ландшафте» предметной области, относящейся к языкам программирования ЭВМ, описать некоторые базовые понятия, обрисовать историю данной предметной области и ее перспективы, позволить ему сравнить подходы, применяющиеся в разных языках — Лиспе, ассемблере, визуальных языках или Си. Есть даже отдельный раздел об *эзотерических языках*. Из этого вытекает особенность — в обзорной работе, говоря обо всем понемногу, мы не можем раскрыть глубины и показать все особенности конкретного языка. Читатель, который ставит перед собой цель действительно глубоко изучить какой-либо из рассматриваемых в настоящей книге языков, должен обратиться к специализированным учебникам (часть из них приводится в списке литературы).

Желательно — но не обязательно! — наличие у читателя начального опыта программирования на каком-либо языке.

ЗАМЕЧАНИЕ

Во введении и далее по тексту книги приняты следующие обозначения. *Ключевые понятия*, на которые стоит обратить внимание, набраны курсивом. Для примеров программ используется моноширинный шрифт.