

4. Програмирање

Компјутерска програма: Множество команди со кои компјутерот извршува одредена работа.

Системски програми:

- оперативни системи
- контролни програми и
- услужни програми

Апликативни програми или кориснички програми.

Програмирање е процес на пишување програма, а луѓето кои пишуваат програми се нарекуваат програмери.

Програмирањето е дел од процесот на решавање некоја задача на компјутер.

Овој процес се состои од повеќе фази:

- поставување на задачата
- дефинирање постапка за решавање на задачата
- пишување програма
- тестирање на програмата

4.1. Програмски јазици

Јазиците претставуваат основно средство за комуникација помеѓу луѓето. Тој се претставува со знаци, со симболи или со говор.

Видови јазици:

- природни и
- вештачки.

Со вештачките јазици се остварува комуникацијата:

Човек – машина или машина – машина.

Програмски јазици

Поделба на програмските јазици:

- машински
- симболички
- виши и
- високи програмски јазици.

Машински јазик

Машинскиот јазик е јазик на компјутерот. Се состои од одреден број **машински инструкции**, кои се изразуваат со 0 и 1.

бинарен систем → хексаден систем

Пример на програма во машински јазик:

B8D616
8ED8
A00000
02060100
BB1080
B44C
CD21

Симболички јазик (асемблер)

Симболичкиот јазик се состои од кратки едноставни зборови (симболи) со кои се означуваат машинските инструкции. На пример, MOV (move), ADD (add) итн.

Пример на програма во симболички јазик:

```
16D7:0000    MOV    AX,16D6
16D7:0003    MOV    DS,AX
16D7:0005    MOV    AL,[1000]
16D7:0008    ADD    AL,[0001]
16D7:000C    MOV    BX,8010
16D7:000F    MOV    [BX],AL
16D7:0011    MOV    AH,4C
16D7:0013    INT    21
```

Специјални системски програми – **преведувачи** (компајлери)

Симболичките јазици се машинско–ориентирани јазици, бидејќи во зависност од процесорот секој компјутер или група компјутери има свој сопствен симболички јазик.

Виши програмски јазици

Вишите програмски јазици се развиваат кон крајот на 50-тите години на дваесетиот век, со цел програмирањето да се приближи кон сè побројните корисници.

Предности на вишите програмски јазици:

- програмирањето значително се олеснува,
- овозможува брзо и ефикасно запишување на алгоритмите во компјутерски програми,
- не е потребно да се знаат техничките својства на компјутерот,
- не зависат од компјутерот на кој се извршува програмата,
- многу брзо и лесно се учат,
- овозможува размена на програми меѓу корисниците.

Вишите програмски јазици имаат своја **азбука** составена од цифри, букви и специјални знаци (интерпункциски знаци и знаци за аритметички и логички операции).

Со комбинација на букви од англискиот јазик се формираат елементарни конструкции на јазикот – **зборовите**, коишто го сочинуваат **речник на јазикот**.

На пример, INPUT, DO, READ, OPEN, IF, FOR, REPEAT, STOP, PRINT итн.

Комбинации од зборовите на јазикот формираат **реченици** или **наредби** (инструкции). Тие имаат строго дефинирани правила на конструкција (синтакса), строго дефинирано значење (семантика) и предизвикуваат строго одредена акција во компјутерот.

Примери:

```
IF N=10 GOTO 50  
REPEAT UNTIL KEYPRESSED  
MULTIPLY A BY B GIVING C
```

Изворна програма (анг. source program) – програма напишана во виш програмски јазик.

Преведувач (анг. compiler)

Извршна програма (анг. executive program)



Извршната програма се сместува како датотека на некоја надворешна меморија, од каде се повикува (активира) кога треба да се изврши.

Понекогаш, се користи и директно извршување на изворната програма, без да биде преведена во извршна. Тоа се прави со системски програми наречени **интерпретери**. Тие не прават датотеки во машинска форма, туку секвенциално (последователно) ги извршуваат наредбите од програмата.

Својства на вишите програмски јазици:

- машински независни јазици,
- проблемски ориентирани јазици.

Хронолошко појавување на некои поважни програмски јазици.

година	јазици		
1945	машински јазик		
1950	асемблер		
1955	FORTRAN		
1960	ALGOL	COBOL	LISP
1965	Basic		
1970	Pascal	PROLOG	
1980	C		
1985	SQL		
1990	C++		
1995	Java	Haskell	
2000	C#		

Класификација на вишите програмски јазици според областа на примена:

Јазици со општа намена	FORTRAN, COBOL, Basic, Pascal. C/C++, Java, C#
Јазици за вештачка интелигенција	PROLOG, LISP, FORT, COMIT и др.
јазици за системско и паралелно програмирање	ADA, MODULA-2 и др.
скриптни јазици	Perl, Tcl, JavaScript, Python и др.
специјални јазици	RPG, SQL, Perl, Python и др.

Пример: Програми напишани во Basic, FORTRAN, Pascal и C, за наоѓање збир на првите 10 природни броеви

```

10 SUMA=0
20 N=1
30 WHILE N<=10
40     SUMA=SUMA+N
50     N=N+1
60 WEND
70 PRINT SUMA
80 END
    
```

```

SUMA=0
N=1
WHILE N<=10
    SUMA=SUMA+N
    N=N+1
ENDWHILE
WRITE(*,30) SUMA
FORMAT(I10)
STOP
    
```

30

```

PROGRAM Suma;
VAR N,Suma: integer;
BEGIN
    Suma=0;
    N=1;
    WHILE N<=10 DO
        BEGIN
            Suma:=Suma+N;
            N:=N+1;
        END;
    WriteLn(Suma);
END.

```

```

main ()
{
    int n,suma;
    suma=0;
    n=1;
    while (n<=10)
    {
        suma+=n;
        n+=1;
    }
    print(“%d”,suma);
}

```

Изворните програми, најчесто, имаат наставка која е кратенка од името на програмскиот јазик.

.BAS, .FOR, .PAS, .C, итн.

Класификација на вишите програмски јазици според начинот на кој се обработуваат податоците:

- императивни и
- декларативни.

Императивниот начин е наредбен начин. Главна карактеристика на програмите коишто функционираат на овој начин е користењето на променливи, наредби и процедури.

влезни податоци → наредби → резултати

Декларативниот начин е описен начин, со кој се опишува како да се дојде до резултатот.

Императивните јазици се делат на:

- **процедурални** (Basic, FORTRAN, Pascal, C, Ada, Modula, и др.) и
- **објектно–ориентирани** (C++, Java, C#, Delphi, Simula, и др.).

Во процедуралните јазици процедурите со кои се обработуваат податоците се одвоени, додека кај објектно–ориентираниите јазици процедурите се скриени во самите објекти.