

Lansarea programului

Dupa incarcare, programul solicita un cod de protectie la care se raspunde folosind matricea din Anexa 1. Se pun patru intrebari de forma 'ENTER THE COLOUR IN SQUARE X-XX (W,Y,G,R)?'. Dupa ce se raspunde corect, apare mesajul de copyright insotit de un mesaj sonor si compilatorul este gata de lucru.

Comenzi directe pentru compilator

Comenzile directe sint precedate de caracterul '*', pentru a fi distincte de comenzile uzuale. Ele sint urmatoarele:

- *C - Compileaza programul Basic
- *R - Lanseaza codul obiect
- *N - Distruge programul Basic si codul obiect, fara a afecta compilatorul
- *S - Salveaza codul obiect
- *I - Alege sursa pentru textul sursa (Ram, Tape, Microdrive)
- *O - Alege suportul pentru codul rezultat (R, T, M)
- *B - Salveaza compilatorul (implementata numai la unele versiuni)
- *Q - Revine in sistemul de operare normal, distrugind compilatorul

Directive de compilare

Pe langa comenzile directe, compilatorul mai recunoaste si o serie de directive de compilare, ce se introduc in textul sursa al programului Basic.

Toate aceste directive de compilare se introduc dupa declaratiile REM urmate de caracterul '!', care prefixeaza orice directiva de compilare.

- REM ! P CODE - are ca efect producerea de catre compilator a unui cod intermediar ce se situeaza ca viteza de executie intre Basic si codul masina propriu-zis, dar este mai scurt decit acesta (aceasta directiva este implicita)
- REM ! MACHINE CODE - codul rezultat este cod masina
- REM ! INT i,j,l,a(5,10) - declara variabilele intregi, deci pe 2 bytes
- REM ! AUTORUN - pusa la inceputul programului, produce autolansarea acestuia dupa incarcare
- REM ! %..... - BLAST-ul poate coexista cu alte extensii Basic care folosesc declaratiile REM. Aceasta directiva transmite linia interpretorului, pentru a putea fi interceptata de eventualele extensii si tratata ca atare

Extensii Basic

BLAST-ul introduce in Basic-ul Sinclair standard o serie de noi elemente, ce sint compilate foarte usor si au o viteza ridicata de executie.

Noile comenzi se introduc tot prin linii REM, de data aceasta urmate de caracterul '&'. Ele sint urmatoarele:

- REM & BREAK ON
- REM & BREAK OFF
- REM & WHILE conditie
- REM & WEND
- REM & REPEAT
- REM & UNTIL conditie
- REM & DOKE - POKE pe 2 bytes
- REM & DEEK - PEEK pe 2 bytes
- REM & CALL adr <,p1,p2,....,pn> - parametrii sint optionalii si se separa prin virgule; pot fi atit valori pe 2 bytes cit si pointeri, caz in care vor aparea

in lista sub forma &n; parametrii sint stocati
 intr-o tabela adresata prin IX: (IX+0),
 (IX+1) primul, etc.
 REM & DEF nume procedura <(p1,p2,...,pn)>
 corp procedura
 =
 REM & ENDPROC - de remarcat faptul ca corpul procedurii nu
 se prefixeaza cu declaratii REM, iar procedurile
 Pot fi recursive.

BLAST TOOLKIT
 OXFORD COMPUTER SYSTEMS SOFTWARE (C)1985

Impreuna cu BLAST-ul se livreaza un TOOLKIT care ajuta la
 editarea programelor Basic si poate salva text sursa sub format
 'Include', pentru compilarea programelor mari.

Comenzi referitoare la o linie

*E n - Edit
 *C n - Copy
 *D n - Delete
 *M n - Move

Comenzi pentru blocuri de linii

*C <line range>,n - Copy
 *D <line range> - Delete
 *M <line range,>n1,n2 - Move
 *R <line range,>n1,n2 - Renumber

Alte comenzi

*T - Trace on
 * - Trace off
 *K - Kill (sterge declaratiile REM ce nu sint urmate de
 caracterele !,& si %)
 *W <line range,> nume - salveaza blocul delimitat
 *B nume - Salveaza in format Include
 *Q - Quit
 *V - Listeaza variabilele de sistem, spatiul liber, etc.
 *L - Listeaza variabilele programului la un moment dat
 *J n1,n2 - concaterneaza liniile desemnate
 *G, *A - similar cu *T si *S, dar nu mai asteapta apasarea
 unei taste

ANEXA 1 - CODURI PROTECTIE

ABCDE_FGHIJ_KLMNO_PQRST_UVWXYZ

40	RYGRY	GGRYG	WGGYG	RYGYG	YGWYGG
39	YGRYW	GYGGR	GYRGY	GGRGR	GYGRGR
38	GYGGY	RRGYG	YRGYW	YRYRG	RGRGYG
37	RGYGR	YGGRY	WYRGY	GWGYG	YGYGGR
36	YRGGY	GRYGG	GRGYG	RGRGW	GYGRGY
35	GGYYG	WYWYG	RYRGR	GYGYR	YGWRYG
34	RYGRR	YGYRY	GYYGY	YGYRG	GRYGYG
33	YGRYG	RYGGR	GGGRG	RYGYG	YGWYRG
32	RGYGY	GRYRY	WGGYR	GWYGY	RGYRYG
31	GYWGR	YGRGG	RYGRY	GGYRG	YRGGRG
30	YRGYG	RYRYR	GGRYG	YRYRY	WYGYGW
29	RYRGW	GGGYR	YWYGR	GGRYR	GGWGRY
28	GRGYG	GYRGY	GYGYG	RRYGY	RRYRGG
27	RYGRY	RRYRG	GGWGR	YGRWR	GYRGRY
26	YRGGY	GYRGY	RYGRY	RGRGG	YWYGYR
25	GYRYG	WGYRG	GRYRG	RGGYR	GYGRGY
24	RGGRG	YGRYW	GRGGR	GWYGG	RGYGYR

23	GGGYR	GRYGG	YYRYG	RGGWY	GRRYGY
22	YWYRY	GYGGY	GRYRG	YRYGR	RGYWYG
21	GGRGR	YRYWG	RYGYG	GYRRY	GRRGRY
20	RGYYG	YGYGR	YRYWY	RRGYG	RYGWYR
19	YGRGG	WYGYG	RGWGW	GGWGW	GGYRRG
18	RYGRG	YRGRG	YWYGY	RGRYG	YGRGRY
17	YWGYR	GRGYG	GYGWR	YRYGY	GYGGYR
16	GGGRG	RYRGR	GRGYG	GGGGG	RGRYRG
15	YRGYR	YWYGR	RYRRY	RGRYR	YRGWYG
14	GYRRG	GYGGR	GWGGR	GYGWG	YGYGGY
13	GGYGY	GYWYG	YGWYG	RGYGY	GYGYRG
12	RYRGG	RGGRY	RGWGW	YGRGW	YRGWGY
11	YRYWG	YGRYW	YGGYG	RYGYG	RYRYGR
10	RGGYY	GRGRY	GRYWY	GRGGW	GWGRGY
09	GYGRG	RYGGG	YRGYG	GYRYG	RYGYYG
08	RGWYR	YGRYR	RGRGR	YGYGR	YGRGRW
07	YRGGY	WGYGG	YGYRY	YWGYR	GRYRYG
06	GYRYR	GGGRG	GRRGR	GGYRY	WYGYRG
05	YWGGG	YYWGY	WGGRG	YRGYG	YGYGYR
04	GGYRG	RGGYR	GYRGY	WYGGY	RGWYRG
03	RYRRY	GRYRY	GRGYG	GRRGR	GYGRGY
02	YGGGR	GYGGR	YGRGR	YRYRY	RGYRYG
01	WYRYG	YWYGY	GRYRG	WYGYG	GRYGGW

ABCDE_FGHIJ_KLMNO_PQRST_UVWXYZ

AMCSOFT TIMISOARA 04.04.1990