

4. Stavka BASE

Oblika stavka: BASE = naslov

Stavek spremeni naslov spremenljivke, uporabljene v programu. Vse spremenljivke, ki so uporabljene po tem stavku, si sledijo v neprekinjenem zaporedju od tega naslova naprej. V programu je lahko več BASE stavkov.

Če bi s stavkom BASE hoteli spremenljivko shraniti na že uporabljeni naslov, bi interpretator preprečil napako in stavek bi bil ignoriran. Na ekranu se bo izpisalo opozorilo:

** WARNING ** VARIABLES UNBOUND

5. Stavka DIM

Oblika stavka: DIM A(N),C(X,Y)...

kjer je:

A(N) vektor A, ki ima N vrstic;


C(X,Y) matrika C, ki ima X vrstic in Y stolpcev.

Polja morajo biti določena s stavkom DIM še preden so uporabljena v programu. Indeksi polja so lahko od 1 do 255.

6. Stavka REM

Oblika stavka: REM komentar

Stavek je uporabljen za pisanje komentarjev v programu. Če interpretator prepozna stavek REM, izpiše komentar, nato prične obdelovati naslednjo vrstico. Vsi stavki, ki v isti vrstici sledijo stavku REM, so ignorirani.

Izdaja						23	Stran	J	K	Identifikacijska številka
Št. obvestila										00E0003PC
 Iskra Elektromehanika Kranj						Arhiv	Namesto identifikacijske številke			

Obr. 6-456


7. Stavak END

Ta stavak povzroči, da se program neha izvajati. Po tem stavku programer lahko spet uporablja ukaze.

8. Stavka TRCON in TRCOFF

če želimo pred izvajanjem stavka v programu izpis tega stavka, dosežemo to s stavkom TRCON. S stavkom TRCOFF tako izvajanje z izpisovanjem prekinemo.

Prenos tretjim osebam in uporaba v nedogovorjene namene nista dovoljena.

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						24				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

9. STAVKI ZA PRIREJANJE

9.1. Stavki LET

Oblika stavka: LET spremenljivka = izraz
 ali
 spremenljivka = izraz

Interpreter izračuna izraz, rezultat pa shrani na naslovu spremenljivke.

9.2. Stavki POKE

Oblika stavka: POKE(naslov)= vrednost


željena vrednost bo shranjena na naslovu, ki je definiran kot argument stavka POKE. Naslov in vrednost sta lahko izraza, ki ju interpreter najprej izračuna. če je izračunana vrednost, ki naj bo shranjena, večja od 255, bo interpreter shranil le desni byte števila.

9.3. Prirejanje nizov

Oblika stavka: ime\$=izraz

 LET ime\$=izraz

Rezultat izraza je prirejen spremenljivki ime\$. če je rezultat daljši, kot je definirana dolžina spremenljivke, bodo shranjeni le prvi znaki (kolikor dopušča spremenljivka).

Izdeja						25	Stran	J	K	Identifikacijska številka
Št. obvestila										QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

10. STAVKI ZA KONTROLO

10.1 Stavki CALL

Oblika stavka: CALL <addr>

Stavek kliče podprogram v strojni kodi na naslovu <addr>, ki je lahko v desetiškem ali šestnajstiškem zapisu. Če ima program na koncu ukaz RTS, se bo izvajanje programa nadaljevalo na stavku, ki sledi stavku CALL.

10.2. Stavki FOR/NEXT

Oblika stavka: FOR <I>=<izraz> TO <izraz> STEP <izraz>
NEXT <I>

Stavek FOR/NEXT uporablja <I> za števec. Zanka je omejena s stavkom NEXT. Če korak povečanja, ki ga določa <izraz> ob stavku STEP, ni določen, bo interpretator povečal števec za 1. števec ne sme biti vektor ali matrika.


Interpretator bo izvajal zanko vsaj enkrat. V programu je lahko vnezdenih največ 20 zank.

Npr.: FOR I=20 TO Z/4 STEP K*3
FOR I=50 TO 30 STEP -1

10.3. Stavki GOSUB/RETURN

Oblika stavka: GOSUB <št>
RETURN

Stavek kliče podprogram, ki se začne z vrstico zaporedne številke <št>. Zadnji stavek podprograma mora biti stavek RETURN. V programu je lahko vnezdenih največ 16 podprogramov.

Izdaja						26	Stran	J	K	Identifikacijska številka
Št. obvestila										OCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

če številka vrstice v stavku ON ERROR GOTO ni definirana, se bo izpisala številka napake, izvajanje programa pa bo končano.

Ko je napaka odkrita, lahko s pomočjo funkcije ERR in stavka PRINT izpišemo številko napake. To številko najdemo v poslavju SEZNAM NAPAK.

Ko je napaka odkrita, lahko s pomočjo funkcije ERR in stavka PRINT izpišemo številko napake. To številko najdemo v poslavju SEZNAM NAPAK.

10.7. Stavka ON/GOTO in ON/GOSUB

Oblika stavkov: ON <izraz> GOTO <št1>, <št2>, ...

ON <izraz> GOSUB <št1>, <št2>, ...

kjer so <št1>, <št2>, ... zaporedne številke vrstic.

Interpreter najprej izračuna izraz. Rezultat izraza pove, s katero vrstico programa se bo izvajanje nadaljevalo.

10.8. Stavka STOP

Oblika stavka: STOP


Stavek prekinje izvajanje programa. Na ekranu se bo izpisalo sporočilo:

STOP <številka vrstice>

10.9. Stavka PAUSE

Oblika stavka: PAUSE

S pomočjo tega stavka prekinemo izvajanje programa. Z ukazom CONT (glej poslavje 3.) izvajanje programa nadaljujemo, kjer smo ga prekinili.

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						28				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

11. VHODNO/IZHODNI stavki

Vsi vhodno/izhodni stavki uporabljajo vmesni pomnilnik. Ta lahko vsebuje do 128 znakov. Posebna uporaba vmesnega pomnilnika BUF\$ je opisana v poslavju 2.3.2.

11.1. Stavek INPUT

Oblika stavka: INPUT <S1>,<S2>,<S3>,...<SN>

kjer so <S1>,<S2>,<S3>,...<SN> imena spremenljivk.

S pomočjo tega stavka vtipkamo podatke za spremenljivke. Na ekranu se najprej izpiše vraščaj. Ko je vtipkan zadnji znak spremenljivke, pritisnemo tipko CARRIAGE RETURN. Celoštevilčne spremenljivke lahko ločimo tudi z vejico ali presledkom, nize pa samo z vejico.

Če vtipkamo kontrolni znak CONTROL X, bo zadnja vrstica brisana. S kontrolno tipko <— (puščica v levo) pomaknemo kazalec za eno mesto nazaj.

Spremenljivke <S1>,...,<SN> so lahko celoštevilčne, BCD ali nizi znakov. Če se zgodi napaka, bo spremenljivka imela vrednost 0 oz. niz bo prazen.

11.2. Stavek PRINT

Stavek uporabljamo za izpis spremenljivk na ekran ali printer. Oblika stavka je:

PRINT spremenljivke


ali

PRINT #I,spremenljivke

I je lahko :

-0 : izpis na ekran

-1 : izpis na printer

Izdaja						Lst	Stran	J	K	Identifikacijska številka
Št. obvestila						29				QCE 000 3PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		


11.4. Stavka: PRECISION

Oblika stavka: PRECISION=I

Celo število I določa natančnost izračuna BCD funkcij (I mest). I leži v mejah od 1 do 18.

Če natančnost s tem stavkom ni določena, so števila avtomatično obdelana na 9 mest natančno.

Prenos tretjim osebam in uporaba v nedogovorjene namene nista dovoljena.

Izdaja						Lst	Stran	J	K	Identifikacijska številka
Št. obvestila						31				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

12. STAVKI ZA DELO Z DISKETO

ID BASIC uporablja olajšave, ki jih nudi operacijski sistem FDOS-3 pri delu z disketo. Način dela je kanalsko orientiran. To pomeni, da moramo prirediti datoteki številko kanala, preden jo prvič uporabimo. Na razpolago je 10 kanalov (od 0 do 9). Zato lahko beremo in vpisujemo največ 10 datotek naenkrat.

Imena datotek morajo ustrezati pravilom operacijskega sistema. Ime je sestavljeno iz ene do osem črk. Če želimo uporabiti podaljšek imena, moramo za zadnjo črko imena napisati piko, šele potem podaljšek. Ta ima lahko do tri znake.

12.1. Stavka OPEN

Oblika stavka: OPEN #filnum><definicija datoteke>

kjer je:

filnum številka kanala od 0 do 9, ki jo hočemo prirediti datoteki (velja samo znotraj tega programa);


<definicija datoteke> ime, podaljšek imena in disketna enota.

Stavka uporabljamo za odpiranje datotek na disketi. Datoteke odpiramo z istim ukazom za pisanje ali branje. Vsaka datoteka mora imeti drugo številko kanala.

Datoteka ne more biti odprta za branje in pisanje istočasno. OPEN stavka bo najprej poiskal datoteko v imeniku datotek na določeni enoti. Če bo datoteko našel, jo bo odprl za branje, drugače pa za pisanje.

S pomočjo funkcije STATUS se usotovi, ali je datoteka odprta za pisanje ali branje. Če se med izvajanjem stavka OPEN zgodi napaka, lahko s pomočjo funkcije ERR usotovimo številko napake in s tem tudi vrsto napake.

Datoteko moramo odpreti, preden želimo delati z njo.

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						32				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

Stavek OPEN je edini stavek, ob katerem moramo navesti ime datoteke in številko enote. Namesto imena datoteke je lahko izraz, ki ime določi.

Npr.:

OPEN #2,"TEST.TXT:1"

ali

A\$="TEST"

OPEN #2,A\$+" .TXT:1"

V zgornjem primeru je ob OPEN stavku definirana datoteka z imenom TEST.TXT na enoti 1.

12.2. Stavek CLOSE FILES

Oblika stavka: CLOSE FILES

Stavek zaključi vse datoteke, ki so bile v programu uporabljene in še niso zaključene.


12.3. Stavek CLOSE

Oblika stavka: CLOSE #filnum

kjer je filnum številka kanala datoteke.

S tem stavkom zaključimo datoteke, ki smo jih že brali ali vpisovali. Po tem ukazu ima lahko isto številko kanala druga datoteka.

Npr.: CLOSE #2

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						33				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

12.4. Stavki READ

Če želimo brati podatke z diskete, uporabimo stavki READ. Pred tem mora biti datoteka že pripravljena za branje. Oblika stavka je:

```
READ #filnum, spremenljivke
READ spremenljivke
```

Spremenljivke v stavku so lahko vseh tipov. S filnum določimo kanal datoteke s katere beremo podatke. Če filnum izpustimo, beremo podatke, definirane s stavkom DATA.

Podatke bere računalnik najprej v vmesni pomnilnik, potem jih šele priredi navedenim spremenljivkam v podanem vrstnem redu. Spremenljivke so lahko celoštevilčne, BCD ali nizi. Lahko so enodimenzionalne, vektorji ali matrike.

Število spremenljivk, navedenih ob stavku READ, se mora ujemati s številom podatkov v enem zapisu datoteke. Dolžine zapisov datoteke so lahko različne. Če je v zapisu več podatkov, kot je navedenih spremenljivk ob stavku READ, bodo podatki, ki še ostanejo v tem zapisu, ignorirani.


Pravila za branje različnih podatkovnih tipov z diskete:

tip naslednje spremenljivke
v zapisu:

tip naslednje
spremenljivke v
stavku READ:

	celoštevilen	BCD	niz	prazen
celoštevilen	celoštevilen	napaka	celoštevilen, napaka če niz vsebuje numerične znake	
BCD	BCD	BCD	BCD, če je niz sestavljen iz BCD števil	napaka
niz	niz znakov	niz znakov	niz znakov	0

Prenos tretjim osebam in uporaba v nedogovorjene namene nista dovoljena.

Izdeja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						34				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

12.5. Stavka WRITE

Oblika stavka: WRITE #filnum,<S1>,<S2>,<S3>,...

kjer so <S1>,<S2>,<S3>,... imena spremenljivk ali izrazov.

Imena in izrazi so ločeni z vejicami.

Pred uporabo tega stavka mora biti datoteka pripravljena za pisanje s pomočjo stavka OPEN. Potem lahko na disketo zapišemo podatke, ki so zapisani z ASCII znaki. Podatki, ki so zapisani z enim stavkom WRITE, so en zapis. Ta lahko vsebuje različno število znakov, največ pa 128. Podatki so za vsako spremenljivko med seboj avtomatično ločeni z vejico. Na koncu zapisa je poseben znak (v heksadecimalnem zapisu \$0D).

Npr.:

WRITE #2,25,-400,"BESEDA",N,25.3E-3

12.6. Stavka DATA

Če je podatkov, ki jih uporabljamo v programu, malo, jih lahko definiramo kar v programu. V ta namen uporabimo stavka DATA. Oblika tega stavka je:


DATA podatki

Podatki so lahko numerični ali alfanumerični. Ločeni so z vejicami. Zapišemo jih v takem zaporedju, kot jih kasneje želimo prebrati. Prav tako kot s stavkom WRITE, kreiramo s stavkom DATA en zapis.

12.7. Stavka CHAIN

Oblika stavka: CHAIN < program >

Stavek povzroči izvajanje navedenega programa. Program, ki ga želimo izvajati na tak način, mora biti zapisan na disketi v strojni kodi. Interpreter ga bo najprej zapisal v pomnilnik na naslov, ki je določen programu. Zato se lahko zgodi, da bo program na istem naslovu, kjer je bil prej zapisan interpretator.

Izdeja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						35				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

Paziti je treba, da se to ne zгоди, če želimo po izvajanju programa vrniti kontrolo interpretatorju.

12.8. Stavak RESTORE

Oblika stavka: RESTORE #filnum,...,#filnum

kjer je filnum številka kanala datoteke.

S tem stavkom zapremo datoteko, ki je bila oderta za branje ali pisanje, in jo ponovno odpremo za branje podatkov.

Npr.: RESTORE #3

12.9. Stavak SCRATCH

Oblika stavka: SCRATCH #filnum,...,#filnum

kjer je filnum številka kanala datoteke.

S tem stavkom zapremo datoteko oderto za branje ali pisanje, in jo ponovno odpremo za pisanje podatkov.

Npr.: SCRATCH #2


12.10. Stavak KILL

Oblika stavka: KILL <ime programa>

Program s tem stavkom zbršemo z diskete. Brisanega programa se ne da ponovno kreirati, zato moramo ta stavak previdno uporabljati.

Npr.: KILL "TEST"

KILL A#


Izdaja						List	Stran	J	K	Identifikacijska številka		
St. obvestila						36				QCE0003PC		
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke				
								<table border="1"> <tr> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table>				

13. POSEBNI STAVKI

Nekaterih stavkov interpretator zaradi narave dela ne more izvajati, prevajalnik pa lahko. Interpreter jih zato obdela kot stavke za komentar (podobno kot stavek REM) in izpiše svarilo. Stavki so vključeni v interpretator zato, da lahko z interpreterjem pripravimo celoten izvorni program, ki sa kasneje hočemo prevajati s prevajalnikom.

Posebni stavki so:

ON NMI GOTO	GEN
ON IRQ GOTO	IRQ ON
ORG	IRQ OFF
OPT	RETI
STACK	TASK
SWITCH	

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						37				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

14. FUNKCIJE Z DISKETAMI

Rezultat teh funkcij je vedno celo število.

14.1. Funkcija EOF

Oblika funkcije: EOF (#filnum)

Vrednost funkcije je 1, če smo pri branju datoteke po zadnjem zapisu želeli prebrati še en zapis, sicer pa je vrednost funkcije 0. S filnum definiramo kanal datoteke, ki jo želimo kontrolirati z funkcijo EOF.

Npr.: IF EOF (#4)=1 THEN 550

14.2. Funkcija FREE

Oblika funkcije: FREE (<št>)

kjer je <št> številka enote siskesa diska.

Vrednost funkcije je število prostih sektorjev na željeni enoti.

Npr.: A=FREE(1)


14.3. Funkcija FILSIZ

Oblika funkcije: FILSIZ (#filnum)

Vrednost funkcije je trenutna dolžina datoteke v sektorjih. Filnum pove kanal datoteke.

Npr.: IF FILSIZ (#4)=35 THEN 440

Prenos treh oseb in uporaba v nedogovorjene namene nista dovoljena.

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						38				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

14.4. Funkcija STATUS

Oblika funkcije: STATUS (#filnum)


Vrednost funkcije je lahko:

- 0 = datoteka, definirana s filnum, ni oderta;
- 1 = datoteka, definirana s filnum, je oderta za branje;
- 2 = datoteka, definirana s filnum, je oderta za pisanje.

Npr.: ON STATUS(#8) GOTO 470,120

IF STATUS(#1)=1 THEN 610


N2=STATUS(#5)

Izdaja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						39				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		


SEZNAM NAPAK

številka napake (ERR)	opis napake
4	sintaktična napaka
5	ime spremenljivke ni definirano
6	ni enačaja
7	nedoločena zaporedna številka vrstice
8	ni oklepaja
9	napaka v numeričnem izrazu ali pričakovan je bil numerični izraz
10	ni številke vrstice ali številka vrstice je napačna
11	ni števila
12	stavki FOR/NEXT niso pravilno vnezdeni
13	preveč različnih spremenljivk v programu
15	ni relacijskega operatorja ali nepravilna uporaba relacijskega operatorja
17	na koncu niza ni narekovaja
18	števec v stavku FOR/NEXT ni celo število ali v stavku FOR/NEXT ni števca
19	polje je že definirano
20	napaka pri definiranju polja (oklepaja ni ali pa jih je preveč)
21	napaka pri definiranju polja (indeks je 0 ali večji od 255)
23	polje ni definirano

Prenos tretjim osebam in uporaba v nedogovorjene namene nista dovoljena.

izdaja						list	Stran	J	K	Identifikacijska številka
št. obvestila						40				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

- 24 napaka pri indeksiranju polja
- 29 ni niza
ali
napaka v izrazu z nizi
- 30 preveč stavkov FOR/NEXT aktivnih
(največ 20)
- 34 ni GOTO ali GOSUB
- 36 napaka pri branju diskete
ali
napaka pri pisanju na disketo
- 37 RETURN brez klica podprograma
- 38 za spremenljivko ni prostora
- 39 nepravilen tip
- 41 nepravilna oblika števila na vhodu
- 62 prekoračitev pri množenju
- 63 deljenje z 0
- 64 napaka pri pretvarjanju
- 65 deljenje z 0 pri BCD številih
- 66 kvadratni koren ali logaritem negativnega števila
- 67 argument inverznega sinusa, kosinusa ali $\exp(x)$ prevelik
- 68 premajhno BCD število
- 69 preveliko BCD število

Izdeja						List	Stran	J	K	Identifikacijska številka
Št. obvestila						41				QCE0003PC
 Iskra Elektromehanika Kranj						Arhiv		Namesto identifikacijske številke		

