

PRESEK

List za mlade matematike, fizike, astronome in računalnikarje

ISSN 0351-6652

Letnik 14 (1986/1987)

Številka 3

Strani 162-169

Vladimir Batagelj, Boris Horvat in Tomaž Pisanski:

LOGO NA RAČUNALNIKU SPECTRUM – SOLI/LCSI SINCLAIR LOGO

Ključne besede: računalništvo, programiranje, programski jeziki, logo.

Elektronska verzija: <http://www.presek.si/14/831-Batagelj.pdf>

© 1986 Društvo matematikov, fizikov in astronomov Slovenije

© 2010 DMFA - založništvo

Vse pravice pridržane. Razmnoževanje ali reproduciranje celote ali posameznih delov brez poprejšnjega dovoljenja založnika ni dovoljeno.

RAČUNALNIŠTVO

LOGO NA RAČUNALNIKU SPECTRUM - SOLI/LCSI Sinclair Logo

Uvod

Med jeziki, namenjenimi za uvod v programiranje, zasluži prav gotovo posebno pozornost jezik logo. Želva grafika in možnost krmiljenja "igrač-robotov" omogočata uvajanje v programiranje že v prvih razredih osnovne šole in celo v vrtcu. Zato je logo po krivici dobil sloves programskega jezika "za otroke". Kratek uvod v logo je pripravil Gregor Pavlič v knjigi "Z računalnikom v matematiko" (DZS).

Tolmače za logo najdemo na večini osebnih računalnikov, tudi na ZX Spectrumu in Commodore-64. Žal pa zanje ni pripravljenih slovenskih navodil. Zato smo se pri Preseku odločili, da pripravimo kratka pregleda praukazov jezika logo na obeh računalnikih. V tej številki objavljamo priročnik za logo na ZX Spectrum.

Za vsak ukaz so podani naslednji podatki:

- črka F na začetku vrstice označuje, da ukaz vrača neki objekt kličočemu ukazu (v jedru ukaza je ukaz OUTPUT); če črke F ni, se ukaz obnaša kot pravi ukaz (spreminja okolje),
- polno ime ukaza,
- število in vrsta vhodov ukaza (parametrov),
- okrajšano ime ukaza (če obstaja),
- kratek opis delovanja ukaza,
- znaki +, *, o in - na koncu vrstice primerjajo logo na obeh računalnikih
 - + ukaz je enak za oba računalnika,
 - * obstaja enakovreden ukaz na drugem računalniku pod drugim imenom,
 - o ukaz lahko na drugem računalniku enostavno sestavimo iz obstoječih ukazov,
- na drugem računalniku tega ukaza ni.

Ukazi so razdeljeni smiselno v skupine, ki se za oba računalnika v glavnem ujemajo. Posamezni ukaz lahko nastopa tudi v več skupinah.

Logo za računalnik Sinclair Spectrum so razvili pri SOLI/LCSI (Les Systems d'Ordinateur Logo Internationale / Logo Computer Systems Inc.) in predstavlja razmeroma popolno narečje jezika logo. Omogoča delo tako s kasetnikom kot tudi z mikrotračnikom ter uporabo tiskalnika.

Tolmač za logo vnesemo s kasetnika v računalnik z ukazom

```
LOAD "logo"
```

oziroma z mikrotračnika z ukazom

```
LOAD *"m";1;"logo"
```

Ko je program vnešen, se pojavi na zaslonu napis:

WELCOME TO SINCLAIR LOGO
(c) SOLI/LCSI 1984 VER 1,6

Na koncu spodnje vrstice je stalno podatek o načinu vnosa znakov:

l - male črke C - velike črke E - način E

Na začetku smo v osnovnem (sprotnem) načinu dela in lahko začnemo ukazovati.

Pri tem sta nam v pomoč pozornika:

? - označuje osnovni (sprotni) nivo
> - označuje način TO (sestavljanje novih ukazov)

kadar pa smo v urejevalniku, je v spodnji vrstici napis:

LOGO EDITOR (c) SOLI / LCSI

Logo zapustimo z ukazom `BYE`, vanj pa se ponovno vrnemo z basiškim ukazom `RUN`; seveda medtem ne smemo pokvariti dela pomnilnika, v katerem se nahaja logo.

Osnove in posebnosti

Logo pozna naslednje vrste objektov:

- besede "a3 25 TRUE "
- sezname [] [FO 50 [2R 5B]]
- cela števila (-99999999 .. 99999999)
- realna števila

najmanjše: 1E-39 predela v 0, 2E-39 predela narobe, 1E-38 je v redu
največje: okrog 1.7014118346E+38

Barve določamo z njihovimi številčkami, ki so enake kot v basicu:

0 - črna	4 - zelena
1 - modra	5 - sinja
2 - rdeča	6 - rumena
3 - vijoličasta	7 - bela

Kote merimo v stopinjah.

Z znakom `E` bomo v nadaljevanju označevali znak za prehod v način `E` (Caps Shift & Symbol Shift), s `Cs` pa tipko Caps Shift.

0. Urejanje ukazov

EDIT	ime	ED	začni urejanje ukaza	ime
Cs	5	< ← >	en znak levo	
Cs	6	< ↓ >	naslednja vrstica	
Cs	7	< ↑ >	prejšnja vrstica	
Cs	8	< → >	en znak desno	

E Cs 5	E < + >	začetek vrstice
E Cs 6	E < ↓ >	konec zaslona
E Cs 7	E < ↑ >	začetek zaslona
E Cs 8	E < - >	konec vrstice
E B	(Begin)	začetek besedila
E E	(End)	konec besedila
E Y	(Yield)	zbriši do konca vrstice in spravi
E R	(Resume)	vrtni shranjeni del vrstice
E P	(Previous)	prejšnja stran
E N	(Next)	naslednja stran
E C	(Complete)	zapusti ukaz EDIT
<BREAK>		prekini urejanje
TO ime		urejanje spremenljivk
TO ime parametri		začni definiranje ukaza
END		konačaj definiranje ukaza v načinu TO

1. Želva grafika

Na zaslon je postavljen koordinatni sistem (x,y) , $-128 < x < 127$, $-88 < y < 87$

BACK n	BK	nazaj n korakov	+
F BACKGROUND	BG	barva podlage	≠
CLEAN		počisti zaslon,пусти želvo	≠
CLEARSCREEN	CS	počisti zaslon, želva v izhodišče	≠
DOT [x y]		pika na (x,y) ,пусти želvo	0
FENCE		omeji želvo na zaslon	≠
FORWARD n	FD	naprej n korakov	+
F HEADING		smer želve	+
HIDETURTLE	HT	skrij; ne riše želve, sled pa	+
HOME		želva v izhodišče	+
LEFT n	LT	zasuk za n stopinj v levo	+
F PENCOLOUR	PC	barva sledi	≠
PENDOWN	PD	spusti pero; želva pušča sled	+
PENERASE	PE	želva briše (dele) sledi,	+
		čez katere gre	
PENREVERSE	PX	želva riše in briše (dele) sledi,	-
		čez katere gre	
PENUP	PU	dvigni pero; želva ne pušča sledi	+
F POSITION	POS	položaj (mesto) želve	≠
RIGHT n	RT	obrni n stopinj v desno	+
F SCRUNCH		razmerje enot na oseh	-
SETBG n		postavi barvo podlage na n	≠
SETBORDER n	SETBR	postavi barvo okvirja na n	-
SETHEADING n	SETH	določi smer želve (absolutno)	+
SETPC n		določi barvo črnila na n	≠
SETPOS [x y]		premik v točko (x,y)	≠
SETSCRUNCH [x y]	SETSCR	določi razmerje enot na oseh	≠
SETX n		premik želve do koordinate x=n	+
SETY n		premik želve do koordinate y=n	+
F SHOWNP		ali je želva vidna	≠
SHOWTURTLE	ST	riši želvo (je vidna)	+
F TOWARDS [x y]		smer želve proti točki (x,y)	+
WINDOW		želva lahko gre čez rob zaslona	-
		in se takrat ne vidi	

WRAP		želva lahko gre čez rob zaslona in se prikaže na nasprotni strani - zaslon je "zvit" v svitek (torus)	+
F XCOR		želvina x koordinata	+
F YCOR		želvina y koordinata	+

2. Operacije in funkcije

ARCCOS n		arkus kosinus	O
ARCCOTANGENT n	ARCCOT	arkus kotangens	O
ARCSIN n		arkus sinus	O
ARCTANGENT n	ARCTAN	arkus tangens	×
COSINE n	COS	kosinus	+
COTANGENT n	COT	kotangens	O
DIV a b		kvocient	×
INT n		celi del	×
PRODUCT a b		zmnožek a*b	-
(PRODUCT a1 a2 ... an)		zmnožek a1*a2*...*an	-
RANDOM n		naključno število	+
REMAINDER a b		ostanek	+
ROUND n		najbližje celo število	+
SINE n	SIN	sinus	+
SQRT n		kvadratni koren .	+
SUM a b		vsota a+b	-
(SUM a1 a2 ... an)		vsota a1+a2+...+an	-
TANGENT n	TAN	tangens	O
AND a b		konjunkcija	×
(AND a1 a2 ... an)		konjunkcija	×
NOT a		negacija	+
OR a b		disjunkcija	×
(OR a1 a2 ... an)		disjunkcija	×
+		seštevanje	+
-		odštevanje	+
*		množenje	+
/		deljenje	+
<		manjši	+
>		večji	+
=		enak	+

3. Seznamski ukazi

ob je beseda ali seznam

F ASCII z		koda ASCII znaka z	+
F BUTFIRST ob	BF	objekt brez prvega elementa	+
F BUTLAST ob	BL	objekt brez zadnjega elementa	+
F CHAR n		znak s kodo ASCII n	+
F COUNT ob		število elementov v objektu	+
F EMPTYP ob		ali je objekt prazen (enakovredno OR ob=["] ob=")	×
F EQUALP ob1 ob2		ali sta objekta enaka	×
F FIRST ob		prvi element objekta	+
F FPUT ob seznam		dodaj objekt na začetek seznama	+
F ITEM n ob		n-ti element objekta	+
F LAST ob		zadnji element objekta	+

F LIST ob1 ob2	ustvari seznam z objektoma kot elementoma	+
F (LIST ob1 ob2 ... obn)	ustvari seznam objektov	+
F LISTP ob	ali je objekt seznam	⊗
F LPUT ob seznam	dodaj objekt na konec seznama	+
F MEMBERP x ob	ali je x element objekta	⊗
F NUMBERP ob	ali je objekt število	⊗
F SENTENCE ob1 ob2 SE	ustvari seznam iz elementov obeh objektov	+
F (SE ob1 ob2 ... obn)	ustvari seznam iz elementov posameznih objektov	+
F WORD w1 w2	ustvari stik dveh besed	+
F (WORD w1 w2 ... wn)	ustvari stik danih besed	+
F WORDP ob	ali je objekt beseda	⊗

4. Definicije ukazov

COPYDEF "ime1 "ime2	prepiši telo ukaza ime2 v ime1	o
DEFINE "ime [[par] [v1] [v2] ... [vn]]	... [vn]]	+
	sestavi ukaz z imenom ime, s parametri par in vrsticami v1 ... vn	
F DEFINEDP w	ali je beseda w ime ukaza	-
F PRIMITIVEP w	ali je beseda w praukaz loga	-
F TEXT "ime	seznam s seznamom parametrov in seznamami vrstic ukaza ime (obratno od DEFINE)	+

5. Prireditveni stavek

MAKE "ime ob	priredi spremenljivki ime vrednost objekta oziroma izraza	+
F NAMEP ob	ali ima objekt vrednost	⊗
F THING "ime	vrednost imena (običajno enakovredno :ime)	+

6. Krmilni stavki

BYE	zapusti LOGO (vanj se vrnemo z RUN)	⊗
IF pog [ukazi1] [ukazi2]	če je pogoj izpolnjen, izvedi seznam ukazi1, sicer izvedi seznam ukazi2	⊗
OUTPOUT ob OP	vrni objekt ob v kličoči ukaz	+
REPEAT n [ukazi]	ponovi n-krat ukaze v seznamu	+
RUN [ukazi]	izvedi ukaze v seznamu (enakovredno REPEAT 1 [ukazi])	+
STOP	vrni se v kličoči ukaz	+
TOPLEVEL	vrni se v osnovni nivo (sprotni način)	+
F TRUE	logični da	o
F FALSE	logični ne	o

7. Branje in izpis

F KEYP	ali je pritisnjena kakšna tipka	⊗
PRINT ob PR	izpiši vredost objekta in se postavi na začetek nove vrstice	+
(PRINT ob1 ob2 ... obn)	izpiši vrednost objektov; med objekti po en presledek	+
F READCHAR RC	počaka na pritisk na tipko in vrne ustrezni znak	⊗

F	READLIST	RL	preberi celo vrstico kot seznam	⊗
	SHOW ob		izpiši objekt	⊗
	SOUND [trajanje višina]		zapiskaj	-
	STARTROBOT		želvine ukaze pošiljaj robotu	-
	STOPROBOT		ustavi pošiljanje ukazov robotu	-
	TYPE ob		izpiši objekt	⊗
	(TYPE ob1 ob2 ... obn)		izpiši objekte brez presledkov	⊗
	WAIT n		počakaj n/50 sekunde	0

8. Zaslonski ukazi

	BRIGHT n		svetlost 0/1	-
	CLEARTEXT	CT	počisti prostor za besedilo	⊗
	COPYSCREEN		prenesi zaslon na tiskalnik	⊗
F	CURSOR		mesto (koordinati) tečke	⊗
	FLASH		vkluči utripanje	-
	INVERSE		zamenja vlogo papirja in črnila	0
	NORMAL		prekliče FLASH in INVERSE	0
	OVER n		zamenja črnilo, čez katerega gre	-
	SETCURSOR [a b]	SETCUR	postavi tečko na koordinato (a,b)	⊗
	SETTC [a b]		določi barvo papirja in črnila	⊗
	TEXTSCREEN	TS	zaslon namenjen za izpisovanje	⊗
F	TEXTCOLOUR	TC	seznam z barvo papirja in črnila	⊗

9. Delovni prostor

namesto imena lahko v ukazih navedemo tudi seznam imen

	ERALL		zbriši vse	⊗
	ERASE "ime	ER	zbriši ukaz ime	+
	ERN "ime		zbriši spremenljivko ime	⊗
	ERNS		zbriši vsa imena spremenljivk in njihove vrednosti	⊗
	ERPS		zbriši vse definicije ukazov	⊗
	POALL		izpiši vse	⊗
	PO "ime		izpiši definicijo ukaza ime	⊗
	PONS		izpiši imena in vrednosti vseh spremenljivk	⊗
	POPS		izpiši vse ukaze	⊗
	POTS		izpiši imena vseh ukazov	⊗
	EDNS		vkluči urejanje imen	⊗

10. Datotečni ukazi in tiskanje

Imamo štiri tipe datotek s podaljški:

LOG	- ukazi in spremenljivke
BIN	- posnetek dela pomnilnika (strojna koda, ...)
SCR	- slika - vsebina zaslonskega pomnilnika
TXT	- vsebina urejevalnika

SAVE "ime [ime1 ime2 ...]	shrani ukaze iz seznama na datoteko ime.LOG	-
SAVEALL "ime	shrani vsebino delovnega prostora na ime.LOG	⊗
SAVED "ime	shrani urejevalnikov prostor	⊗
SAVESCRC "ime	shrani sliko na zaslonu na ime.SCR	⊗
SETDRIVE n	izberi mikrotračnik (1-8), 0-trak	⊗

CATALOG	seznam datotek na mikrotračniku	⌘
ERASEFILE "ime tip	zbriši datoteko	+
LOAD "ime tip	vnesi datoteko	+
LOADD "ime	vnesi urejevalnikovo datoteko	⌘
LOADSCR "ime	vnesi zaslonsko sliko	⌘
PRINTON	vkluči izpis na tiskalnik	⌘
PRINTOFF	izključi izpis na tiskalnik	⌘
COPYSCREEN	preinesi zaslon na tiskalnik	-

11. Sistemski ukazi

F NODES	število prostih členov	+
RECYCLE	spusti proste (odrabljene) člene (garbage collection)	⌘
.CONTENTS	izpiši vse, kar poznaš	+
.PRIMITIVES	izpiši praukaze	-
.RESERVE n	zasedi prostor za strojno kodo	-
.RESERVED	izpiše podatke o zasedenem prostoru	-
.BLOAD "ime naslov	vnese strojno kodo z ime.BIN na naslov	⌘
.BSAVE "ime [naslov dolžina]	shrani strojno kodo na ime.BIN	⌘
.SETSERIAL n	postavi hitrost prenosa	-
F .SERIALIN	prebere RS232	-
.SERIALOUT n	pošlje n na vrata RS232	-
.DEPOSIT naslov n	postavi v zlog naslov vrednost n (POKE)	+
F .EXAMINE naslov	vrednost zloga naslov (PEEK)	+
.CALL naslov	poženi strojno kodo na danem naslovu	+

12. Sporočila o napakah

Not enough inputs to ...	ni dovolj vhodov (podatkov) za ukaz ...
I don't know how to ...	ne vem, kaj naj naredim z ... (Logo ne pozna ukaza)
You don't say what to do with ...	nisi povedal, kaj naj naredim z ...
... does not output to ne vrača vrednosti v ...
... is used by LOGO	... uporablja LOGO (hoteli smo definirati praukaz)
... is already defined	... je že definiran (že definirane ukaze lahko urejamo samo z EDIT, mi pa smo hoteli s TO)
... is not true or false	... vrednost ni TRUE ali FALSE (na primer izraz za geslom IF)
... is not a word	... ni beseda (določeni ukazi delujejo samo na besedah kot vhodih)
... defined	... definiran
To many inside parentheses	preveč notranjih okroglih oklepajev (verjetno manjka kakšen zaklepaj)
... open file problem	problem z odprtjem datoteke

