

# 3D CONSTRUCTION KIT



ZX Spectrum, Didaktik

## Ú V O D

Program 3D CONSTRUCTION KIT Vám umožní tvořit vlastní hry v 3D-grafice pomocí systému FREESCAPE (např. Total Eclipse, Castle Master, apod.). Záleží pouze na Vaší fantazii.

Skládá se ze tří samostatných částí :

- THE ENVIRONMENT EDITOR - editor objektů, hracího prostoru, průchodů a barev.
- THE CONDITION EDITOR - editor programu, hlášení, informací a umožňuje testovat vytvořený hrací prostor bez obrazovky editoru.
- THE FREESCAPE COMPILER - skompiluje vytvořenou hru do samostatného programu.

### EDITOR GRAFIKY A POZADÍ

Editor grafiky a pozadí umožňuje editovat hrací prostor (3D-svět) a pohybovat se v něm. Pohyb je možný ve třech směrech:

Osa - X : vlevo, vpravo  
Y : nahoru, dolů  
Z : dopředu, dozadu

K pohybu lze použít ikon nebo kláves (příloha). Stiskem klávesy SYMBOL SHIFT lze rychlost pohybu regulovat.

### Hrací prostor

Hrací prostor je ohraničen prostorovým hranolem o rozměrech 8192 x 4096 x 8192. Bod o souřadnicích 0,0,0 je nejnižším, nejlevějším rohem hranolu. V každém místě lze provádět natočení v úhlu od 0 do 360° po 5°. Pozice udaná jako  $X = 0^0$ ,  $Y = 0^0$ ,  $Z = 0^0$ , je pohledem přímo před sebe a  $X = 0^0$ ,  $Y = 180^0$ ,  $Z = 0^0$ , je pohled vzad.

Je možno definovat až 254 částí. Část může sloužit k tvorbě např. vnitřku domu nebo jeho okolí. Nelze se pohybovat mimo tyto části. Části nejsou spolu spojeny geometricky, ale mají tzv. ENTRANCE, společné body sloužící k průchodu. Průchod je dán pozicí a místem průchodu.

## Objekty

-----

Objekty jsou stavebními prvky pro tvorbu hracího plánu. Přes tyto objekty nelze procházet. Rozměry objektů jsou udány v jiných rozměrech než jsou souřadnice hracího prostoru. Rozměr 1 odpovídá rozměru 64 v hracím prostoru.

Je nadefinováno několik základních objektů. Složitější útvary je nutno vytvořit pomocí těchto základních.

### Základní objekty :

Krychle	- Cuboids
Pyramida	- Pyramids
Jehlan	- Rectangles
Trojhranný jehlan	- Triangles
Čtyřhranný jehlan	- Quadrilaterals
Pětihřanný jehlan	- Pentagons
Šestihřanný jehlan	- Hexagons
Čáry	- Lines
Čidla	- Sensors (Body, které odhalí Vaši pozici a např. po Vás střelí)

### Atributy objektů

-----

Objekt má tři atributy : - viditelný (visible)  
- neviditelný (invisible)  
- zničený (destroyed)

### Barvy objektů

-----

Každá strana objektu může mít jinou barvu (odstín). Odstín 0 má zvláštní prioritu, není kreslen. Může být použit ke kreslení neviditelných stran objektu k urychlení tvorby hry. Nebo ho použít k vytvoření neviditelné bariéry, která bude mít atribut viditelná a nebude možno přes ní přejít.

Ve snaze ušetřit co nejvíce paměť je možno vytvořit nějaký objekt (např. dřevo) a ten umístit např. v části 255. Objekty z této části pak mohou být použity později v libovolné další části použitím příkazu GLOBAL v editoru. Objekt, který byl vytvořen v části 255 (dřevo) se může objevit v jiné části pouze ve stejném místě.

Objekty máme vytvořeny a co dál!? Potřebujeme s objekty provádět různé operace. K tomu nám slouží jazyk FCL.

Jazyk FCL nám umožňuje popsat vztahy mezi objekty a tím napsat krátký program tzv.CONDITIONS. Jazyk má k dispozici příkazy k manipulaci s objekty a k reakci na plnění úkolů. Dále umožňuje definovat proměnné, které lze využít k odpamatování stavu hry (je možno blokovat přístup k těmto informacím). Těchto proměnných je 128. Prvních 112 je volně přístupných uživateli a zbývající používá systém (od 112), které lze také využít, ale opatrně.

Typy instrukcí jazyka FCL :

K manipulaci s proměnnými.

K manipulaci s objekty.

Ke změně typu pohybu - chození, létání.

K tvorbě podprogramů.

Příkazy ke komunikaci.

Podprogramy jsou krátké programy v jazyku FCL, které jsou ukládány pod číslem, kterým se identifikují. Existují tři typy podprogramů :

- globální
- lokální
- procedury

#### Podprogramy globální

---

Jsou aktivní celou dobu hry. Oznamují např. konec hry, skóre atd.

#### Podprogramy lokální

---

Aktivují se pouze v dané části hry. Např. reakce na čidlo, na kolize atd.

#### Procedury

---

Podprogramy, které mohou být vyvolány jiným podprogramem. Možno využít místo často se opakujících příkazů.

## Komunikace

-----

Funkce sloužící ke komunikaci s hráčem (INSTRUMENT). Tyto informace nejsou viditelné v editoru, ale až ve volbě TEST. Existují tři typy informací :

Sloupce - BARS - zobrazuje hodnotu proměnné ve sloupci svisle nebo vodorovně.

Čísla - NUMERICAL - zobrazuje hodnotu proměnné číslem. Pro 4 až 5 - ti místné číslo je potřeba použít dvou proměnných.

Texty - každý text je uložen v souboru. Má vždy max. délku jednoho řádku. Text má své číslo a pozici zobrazení.

## P O P I S E D I T O R U

Obrazovka je rozdělena na pět částí :

Menu - ukazuje dostupné volby a velikost volné paměti.

Pracovní okno - okno, ve kterém se provádí tvorba hry.

Souřadnice - udávají polohu.

Ikony sloužící k pohybu.

Ostatní ikony, které se mění podle toho co se právě provádí.

Funkce editoru lze volit dvěma způsoby :

- ikonami
- přes menu

Výběr se provádí nastavením kurzoru na příslušnou volbu a stiskem FIRE.

Vlevo dole jsou volby :

VIEW - pohled ze všech stran na hrací prostor. Pro lepší pochopení je na kazetě nahrána demostrační hra, kterou lze hrát použitím volby TEST. Hra se nahraje pomocí volby FILE/LOAD z menu a zadáním názvu KITGAME9. Cílem hry je útěk z tajemného světa na Zem. Možná s využitím rakety ... (cenná rada).

MODE - přepíná způsob pohybu :

WALK - umožňuje chození kolem objektů, případně výstup na ně (pokud nejsou příliš vysoké).

FLY1, FLY2 - umožňují létat po světě.

Dialogová okna - přes dialogová okna jsou zadávány parametry funkcí, které je vyžadují.

## EDITACE FCL

EditáčnÍ okno se objevÍ po volbě CONDITIONS z menu. Blikající místo označuje pozici, na kterou se budou příkazy vypisovat. Editor umožňuje napsat instrukci o délce 8 písmen, než se kurzor přesune na další pozici, kde se zadávají parametry. Okamžitý přesun kurzoru lze vyvolat stiskem SPACE.

Instrukce mohou mít 0 až 3 parametry. Každý parametr může nabývat hodnot 0 až 255. Alfanaumerické znaky lze použít pouze v prvním a čísla v dalších. V případě chyby bude obrazovka blikat červeně a kurzor se nastaví na pozici, kde je chyba.

SHIFT + D - maže prosvětlený řádek.  
SHIFT + E - edituje prosvětlený řádek.  
SHIFT + C - vymaže tabulku hlášení.  
BREAK - opuštění editoru.

### Postup při tvorbě a editaci objektu

-----

Prvním krokem je vyčištění pracovního okna a nahrání editoru z pásku příkazem LOAD"". Dále pak zvolíme CREATE. ObjevÍ se nabídka základních objektů. Zvolte např. krychli (CUBE) a v pracovním okně se objevÍ první vytvořený objekt. Teď použijte volbu SHADE a ukáže se :

```
EXIT
001 ENTRANCE
002 CUBE
```

Vyberte CUBE a v dolním okně vlevo se objevÍ popis každé stěny objektu (její barva). Změna se provádí volbou stěny a vepsáním čísla barvy 0 až 15, které odpovídá odstínu vpravo. Stiskem ENTER se změna provede. Stejným postupem obarvíte i ostatní stěny. Ukončení se provede volbou O.K a stiskem FIRE. Dále vyberte EDIT a zvolte 002 CUBE. V okně EDIT je pět ikon. STRETCH zvětšuje a SHRINK zmenšuje objekt. Po úpravách velikosti vraťte velikost na standardní hodnoty (8x8x8). Zvolte O.K, pak COPY a znovu vyberte CUBE. Teď jste právě vytvořili další kopii s názvem 003 CUBE, kterou můžete také editovat.

### Přikazy FILE MENU :

Název : SAVE  
Funkce : Nahrání dat na magnetofon. Po navolení TAPE zadejte číslo (0 - 9). Na magnetofon se nahraje blok s názvem DATAITx - x = číslo verze.

Název : LOAD  
Funkce : Nahrání dat z magnetofonu do počítače. Stiskem ENTER se nahraje první verze, která je v tom okamžiku na pásku.

Název : NEW (pouze u 128k)  
Funkce : Inicializace pracovního okna obrazovky. Musíte zvolit O.K pro aktivaci funkcí nebo ABORT pro zrušení inicializace.

### Přikazy GENERAL MENU :

Název : RESET  
Funkce : Nastaví hru do počáteční pozice, podle nastavení v SETUP.

Název : SETUP  
Funkce : Nastavení parametrů hry.

Název : INSTRUMENT  
Funkce : Editace hlášení, které informují hráče o stavu hry.  
TYPE - typ informace : blank - neaktivní  
  number - číselná  
  H bar - horizontální sloupec  
  V bar - vertikální sloupec  
Sloupec je zobrazen v rozměru bod \* 8, kde bod je roven hodnotě proměnné.  
XPOS, YPOS - pozice informace.  
LENGHT - maximální délka.  
VARIABLE - proměnná pro aktivaci informace.  
COLOUR - atribut barvy (INK+PAPER\*8+BRIGHT\*64+FLASH\*128).

Název : SETWINDOW  
Funkce : Nastavení pozice a rozměru okna pro hru. Jednotkou je 8 bodů. Po zadání zvolte O.K.  
Pozor : Je podmíněno XPOS, YPOS, XSIZE, YSIZE.

Název : TEST  
Funkce : Můžete si zahrát a projít to, co jste si stvořili.  
Ovládání je stejné jako při editaci.

### Příkazy AREA MENU :

- Název : ADD AREA  
Funkce : Vytvoření nové části.
- Název : EDIT AREA  
Funkce : Informuje o hodnotách v dané části - název části, počet objektů apod.
- Název : GOTO AREA  
Funkce : Přejít do jiné části.
- Název : DELETE AREA  
Funkce : Zrušení části.
- Název : COLOUR AREA  
Funkce : Nastavení barvy dané části.
- Název : ADD ENTRANCE  
Funkce : Vytvoří průchod mezi částmi v místě, kde je kurzor a je zapamatován i směr pohledu.
- Název : EDIT ENTRANCE  
Funkce : Editace existujících průchodů.
- Název : GOTO ENTRANCE  
Funkce : Přejít do vybraného průchodu v aktivní části.

### Příkazy CONDITIONS MENU :

- Název : GENERAL  
Funkce : Tvorba (CREATE), rušení (DELETE) a editace (EDIT) programů.
- Název : LOCAL  
Funkce : Jako GENERAL, ale týká se pouze vždy jedné části hry.
- Název : PROC  
Funkce : Jako GENERAL, ale týká se procedur (podprogramů).
- Název : MESSAGE  
Funkce : Tvorba, editace a rušení zpráv.  
Upozornění : Použití pouze ve volbě TEST.



Další ikony :

Název : GLOBAL

Funkce : Ukáže seznam definovaných objektů, kde označení + a -  
znamená : + objekt je v dané části použit,  
- objekt není použit.  
Stiskem FIRE lze znaménka změnit.

Název : COPY

Funkce : Kopíruje vybraný objekt. Objekt se vybírá ze seznamu a kopíruje se na místo, kam ukazuje kurzor.

Název : CREATE

Funkce : Tvorba objektu v aktivní části.

Název : EDIT

Funkce : Editace vybraného objektu. Výběr objektu se provádí ze seznamu objektů. Teď je možné provádět změnu rozměru a polohy.  
UNDO - Ruší provedené změny na editovaném objektu,  
SELECT - výběr dalšího objektu k editaci,  
OK - potvrzení změn.

Název : SHADE

Funkce : Vybarvení daného objektu. Po výběru jsou vlevo zobrazeny strany objektu a v pravé dostupné barvy. Volbou OK se vybarvení provede.

Název : DELETE

Funkce : Vymazání vybraného objektu z paměti.

Název : ATTRIBUTES

Funkce : Zobrazení a možnost změn vlastností objektů. Po výběru objektu se zobrazí typ, název, rozměr, pozice, stav.

Upozornění : SENZORY s hodnotou 0 nestřílí, jinak číslo udává počet střel za sekundu.

## F R E E S C A P E C O M M A N D L A N G U A G E

System FREESCAPE používá jednoduchý jazyk pro popis příběhů v třírozměrném světě.

GENERAL CONDITIONS - příkazy jsou platné pro celý program.

LOCAL CONDITIONS - příkazy se týkají pouze částí programu (AREAS).

PROCEDURE CONDITIONS - podprogramy volané instrukcí CALL.

Označení :

-----

P1, P2, ..., Pn - parametr

V1, V2, ..., Vn - proměnná

### ADCV (Add To Variable With Carry)

---

AOCV P1 V2

Součet hodnot P1, V2 a příznaku přetečení (0 nebo 1). Příznak přetečení se nastaví pokud výsledek překročí 255. Viz ADDV, SUBV, SBCV.

### ADDV (Add To Variable)

---

ADDV P1 V2

Součet hodnot P1 a V2. Příznak přetečení se nastaví pokud výsledek překročí 255.

### AND

---

```
IF <xx>
AND
IF <xx>
THEN příkaz1
[ELSE příkaz2]
ENDIF
```

Provádí logický součin podmínek, pokud jsou všechny pravdivé (true) provede se příkaz1 (může být i více příkazů), jinak (false) příkaz2. Viz IFEQ, IFLT, IFGT, THEN, ELSE, ENDIF, OR.

### ANDV (And Variable)

---

ANDV P1 V2

Logický součin P1 a V2. Výsledek je uložen v V2. Viz ORV, XORV.

### CMPV (Compare Variable With Abs. Value)

---

CMPV P1 V2

Porovnání hodnot P1 a V2 (odečtením V2 od P1 beze změny hodnoty V2). Nastavuje příznaky ZERO (P1=V2) a CARRY(V2>P1). CMPV se používá převážně s IFEQ, IFLT, IFGT.

### CALL (Call Procedure)

---

CALL P1

Skok do podprogramu P1. Po návratu pokračuje program následnou instrukcí po CALL P1.

## CROSS

-----

CROSS P1

Změna ON - P1 = 1, OFF - P1 = 0.

## COLOUR

-----

COLOUR P1 P2

Změna barev na obrazovce.

P1 - typ atributu : 0 - ink

1 - paper

2 - bright

3 - flash

P2 - hodnota atributu. Bity : 0 - 2 ink

3 - 5 paper

6 - bright

7 - flash

## DELAY

-----

DELAY P1

Zastaví děj na nastavený čas. Čas je dán  $P1 * (1/50)$  sekund.

## DESTROY

-----

DESTROY P1 [P2] (objekt[,část])

Označí objekt jako zničený.

Např.:

```
IFSHOT 4 2 ; objekt 4 byl setřelen v části 2 ?
```

```
THEN
```

```
DESTROY 4 2 ; jestliže ano musí být označen jako zničený
```

```
ENDIF
```

## ELSE

-----

```
IF <xx>
```

```
THEN příkaz1
```

```
ELSE příkaz2
```

Je pouze částí testovací instrukce IF a vykoná se pouze pokud výraz <xx> je nepravdivý.

END

----

END

Ukončuje podprogramy. Pokud není použita, program pokračuje až narazí na ENDIF.

ENDGAME

-----

ENDGAME

Provede se inicializace programu. Používá se pro ukončení hry.

Např. : Proměnná 10 obsahuje čas.

CMPV 0 10

IFEQ

THEN

ENDGAME ; Ukončení hry po vynulování času

ENDIF

ENDIF

----

ENDIF

Musí ukončit každý IF.

GOTO

----

GOTO P1 [P2] (entrance[,area])

Přechod mezi jednotlivými částmi hracího prostoru.

P1 - číslo přechodu,

P2 - název části.

Např.:

IFSHOT 9

THEN

GOTO 1 2

ENDIF

;Jestliže hráč střelil do dveří 9 bude pokračovat průchodem 1 v části 2.

## IFACTIVE

-----

```
IFACTIVE P1 [P2] (objekt[,část])
IFACTIVE o [a]
THEN
příkazy
ENDIF
```

Testuje aktivitu objektu.

Např.:

```
IFACTIVE 4
THEN
INVIS 4
ENDIF
```

;Jestliže objekt 4 je aktivní nechť je neviditelný.

## IFCRUSH

-----

```
IFCRUSH
THEN
příkazy
ENDIF
```

Testuje kolizi hráče a objektu.

## IFEQ (If equal)

-----

```
IFEQ
THEN
příkazy
[ELSE
příkazy]
ENDIF
```

Instrukce po IF se provede tehdy, když předchozí instrukce vrací příznak ZERO. Viz CMPV.

## IFFALL

-----

```
IFFALL
THEN
příkazy
[ELSE
příkazy]
ENDIF
```

Instrukce po IF se provede tehdy, když hráč spadne z nebezpečné výšky. Lze nastavit FALL ABILITY v SETUPu.

## IFGT (If Greater Than)

-----

```
IFGT
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Instrukce po IF se provede tehdy, když předchozí instrukce nastavila CARRY a resetovala ZERO.

## IFHIT

-----

```
IFHIT P1
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Test na kolizi objektu P1.

Např.:

```
IFHIT 4
THEN
INVIS 4
VIS 5
ENDIF
```

Pokud objekt 4 je v kolizi stane se neviditelný a objekt 5 viditelný. Možno použít při otevření dveří 4 a jejich výměna za dveře otevřené 5.

## IFLT (If Less Than)

-----

```
IFLT
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Instrukce po IF se provede tehdy, když předchozí instrukce resetovala CARRY a ZERO.

## IFSENSED

-----

```
IFSENSED
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Instrukce po IF se provede tehdy, když hráč vstoupil do prostoru hlídaného senzorem.

## IFSHOT

-----

```
IFSHOT o (objekt)
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Test zda objekt o je sestřelen.

## IFTIMER

-----

```
IFTIMER
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Instrukce po IF se provede tehdy, když uplynul čas nastavený v SETUPu.

## IFVIS

-----

```
IFVIS P1 [P2] (objekt[,část])
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Test na viditelnost (INVISIBLE) objektu v hrané části.

## INVIS

-----

INVIS P1 [P2]

Objekt P1 v části P2 se stane neviditelným.  
Např.:

```
IFSHOT 8
THEN
INVIS 9
ENDIF
```

;Jestliže objekt 8 je sestřelen, tak objekt 9 se stane neviditelným.

## MODE

-----

MODE P1

Instrukce pro změnu pohybu.

```
P1 - 0 - plavání
      1 - chození
      2 - běh
      3 - let 1
      4 - let 2
```

## OR

--

```
IF <xx>
OR IF <xx>
THEN
přikazy
[[ELSE
přikazy]
ENDIF
```

Logický součet používaný ve spojení s instrukcí IF.

## ORV

---

ORV P1 V2

Logický součet P1 a V2. Výsledek je uložen v V2.

## PRINT

-----

PRINT mm xx yy

Vypiše zprávu mm na pozici danou souřadnicemi xx a yy.



## REDRAW

-----

### REDRAW

Upraví zobrazení objektů podle nových vlastností, pokud došlo ke změně.

## SETV

-----

### SETV P1 V2

Provede přiřazení  $V2:=P1$ .

## SOUND

-----

### SOUND P1

Zazní tón P1 (0-12). Viz dodatek.

## SBCV

-----

### SBCV P1 V2

Odečte hodnotu P1 a příznak CARRY od V2.

## SYNCSND

-----

### SYNCSND P1

Synchronizuje zvuk s okrajem.

## TEXTCOL

-----

### TEXTCOL oo

Nastavení barvy oo pro text zprávy.

## THEN

-----

```
IF <xx>
THEN
přikazy
[ELSE
přikazy]
ENDIF
```

Používá se pouze ve spojení s instrukcí IF a vykoná se pouze tehdy, když je výraz <xx> pravdivý.

## TIMER

-----

TIMER P1

Nastavení času P1\*(1/50) sekund.

## TOGVIS

-----

TOGVIS P1 [P2] (objekt[,část])

Změna stavu objektu z viditelného na neviditelný a naopak.

## VIS

---

VIS P1 [P2] (objekt[,část])

Nastavení stavu objektu na viditelný.

## XORV

-----

XORV P1 V2

Logický výběrový součet. Výsledek je v V2.

## P Ř Í K L A D Y

Jak projít z jedné části do druhé části hracího prostoru ?

-----

V příkladu použijeme objekty : 3 - dveře, 4 - průchod přes dveře. Průchod je neviditelný - INVISIBLE.

```
IFACTIVE 3
THEN
INVIS 3
VIS 4
ENDIF
```

Experimentujeme stiskem SPACE a nastavíme kurzor na dveře a stiskneme A. Dveře 3 se musí změnit na dveře 4 (průchod dveřmi). Novou část vytvoříme pomocí ADD AREA a vrátíme se do části 1 pomocí GOTO AREA a napíšeme další program :

```
IFHIT 4
THEN
GOTO 1 2
ENDIF
```

Teď projdeme přes průchod a musíme se ocitnout v části 2.

## Změna viditelného stavu objektu na neviditelný

---

Pokud chceme, aby se objekt 3 změnil z neviditelného na viditelný po výstřelu, napíšeme lokální podprogram :

```
IFSHOT 3
THEN
INVIS 3
ENDIF
```

Střelíme do objektu 3 a efekt musí nastat.

### Vytvoření zvukového efektu

---

Zvuk se ozve, když posuneme objektem 5.

```
IFACTIVE 5
THEN
INVIS 5
SYNCSND 1
ENDIF
```

### Jak používat proměnné ?

---

Vytvoříme proměnnou, která bude zvětšovat svoji hodnotu, když bude daný objekt 3 sestřelen.

```
IFSHOT 3
THEN
ADDV 25 21
```

Hodnotu 25 jsme tímto vložili do proměnné 21. Nastavení hodnoty proměnné je např.: SETV 30 21.

Dá se to použít i jako

```
CMPV 0 21
IFLT
THEN
SETV 30 21
ENDIF
```

Jestliže proměnná 21 obsahuje hodnotu větší jak 0, bude jí přiřazena hodnota 30.

Hodnoty dvoubajtové 0 - 65535 musíme uložit do dvou bajtů tzn. použít dvě proměnné. Např. chceme výsledek skóre jako pětimístné číslo. Do proměnné 10 uložíme nižší bajt a do 11 vyšší bajt čísla.

Skóre je 100 :

ADDV 100 10  
ADCV 0 11

Skóre je 300 :

ADDV 44 10  
ADCV 1 11 ;  $1 * 256 + 44 = 300$

Odečtení hodnoty 300 :

SUBV 44 10  
SBCV 1 11

## S E Z N A M   S Y S T É M O V Ý C H   P R O M Ě N N Ý C H

- 112 - bod pohledu, pozice x (nižší bajt)
- 113 - bod pohledu, pozice x (vyšší bajt)
- 114 - bod pohledu, pozice y (nižší bajt)
- 115 - bod pohledu, pozice y (vyšší bajt)
- 116 - bod pohledu, pozice z (nižší bajt)
- 117 - bod pohledu, pozice z (vyšší bajt)
- 118 - bod pohledu, obrat okolo x (0 - 71, po 5°)
- 119 - bod pohledu, obrat okolo y (0 - 71, po 5°)
- 120 - bod pohledu, obrat okolo z (0 - 71, po 5°)
- 121 - kód stisknuté klávesy
- 122 - počítadlo přerušení (nižší bajt)
- 123 - počítadlo přerušení (vyšší bajt)
- 124 - číslo právě hrané části hracího prostoru
- 125 - čítač střel (0 - nelze střelet, 255 - čítač neaktivní)
- 126 - znakový generátor (nižší bajt)
- 127 - znakový generátor (vyšší bajt)

## P O U Ž I T Í   K O M P I L Á T O R U

Po nahrání kompilátoru, vložíme do magnetofonu pásek na který chcete nahrát skompilovanou hru. Kompilátor nahraje první část programu. Teď nahrajeme do kompilátoru data předtím nahrané z editoru. Nahrajeme skompilované data na náš pásek. Teď nahrajeme obrázek (vytvořený např. pomocí ART STUDIA) a dále nahrajeme poslední část z kompilátoru.

A již máme první vlastní 3D hru. Autoři programu pouze žádají, aby bylo uvedeno, že program byl vytvořen pomocí 3D CONSTRUCTION KIT.

### D O D A T E K

#### OVLÁDÁNÍ :

O - vpřed	K - vzad
Z - vlevo	X - vpravo
R - nahoru	
F - dolů	
I - pohled vpřed	U - pohled vzad
W - obrat vpravo	Q - obrat vlevo
L - pohled dolů	P - pohled nahoru
M - posun vpravo	N - posun vlevo
A - aktivace objektu	
B - střelba	
SPACE - typ pohybu	
SYMBOL SHIFT - rychlost	
BREAK - restart hry nebo návrat do editoru	

#### Zvukové efekty :

0 - lup	4 - aktivace	8 - bonus 1
1 - pink	5 - úder	9 - bonus 2
2 - vrčení	6 - pád	10 - bonus 3
3 - střela	7 - porážka	11 - otevření dveří
		12 - zavření dveří

#### Povolené rozsahy rozměrů :

Pozice objektu	X - 0 až 127
	Y - 0 až 63
	Z - 0 až 127
Rozměr objektu	X - 0 až 127
	Y - 0 až 63
	Z - 0 až 127
Bod umístění	X - 0 až 8191 (128x64)
	Y - 0 až 4093 (64x64)
	Z - 0 až 8191 (128x64)

JACK