

FANZIX
2.0

BACKPACKERS
GUIDE TO THE UNIVERSE PART 1

For those spechums who have never save the Universe.

ENGLISH SUPPLEMENT



WELCOME



Warning: English is not the writers' native language. Read own responsibility!



Welcome to my reader!

The issues of FanZiX have only been published in Hungarian. This special issue is an exception. The reason for the English edition is justified by the extraordinary nature of the subject. The subject, which at first glance does not seem to be overpowering...

Backpackers Guide to the Universe Part 1 was written by Bob Hamilton. Bob Hamilton is an individuality, a man, who dreamed a mega-colossal game, not too only a game, but a trilogy. The trilogy was not finished, only the first part, indeed, the first part was ready partly only actually, the graphics were cutted, there was no sound, the game was buggy... Backpackers was the last hope of Fantasy, but the sales were not as expected, the publisher went bankrupt and the game got to be forgotten. Nowadays the rating of the game is rather weak (WOS: 6,43/10, position is only the 4th between the five Fantasy games), there is no walkthrough, no error correction, no pokes, no interest. But why this game was so unsuccessful? The players' expectations were too high seeing the Crash preview, where the game shown better face. Or the game was unplayable due to a bug, which has not been corrected since then. Or it was just too complicated and couldn't motivate the players... Who knows. If you read the next few pages, maybe you will see the game differently, maybe you will re-evaluate your expectations, maybe you will understand the relationships and the logics of the game, maybe you will appreciate it better.

m/zx

CONTENTS

WELCOME	2
RATINGS	3
TIPS & TRICKS	5
BUGS	11
PROGRAMMER'S CORNER	14
HACK ATTACK	18

There are some articles only in Hungarian what are not in this issue. Beg your pardon but we can't translate our mailing-story and the full guide. If you are interested in, see there. Thank you for your comprehension.

I don't write a reference list about pictures or anything else. Ask if the source of something interests you.

Publisher, editor, author and collector of the foreign contents:
m/zx and Pgyuri, mzx or pgyuri at sinclair dot hu

RATINGS

M/ZX

This is the first part of a trilogy what would be the last. Two programs, a game and an encyclopaedia. The challenge is complex, not a really shoot-em-up type but more than a simply and average puzzle.

The world is colourful, varied, it is shown Bob Hamilton had a graphic artist helper. Ziggy got out his capsule at last and we saw how he could walk around on his thin legs like a true superhero. The labyrinth is very interesting, it as special feeling, the rooms are almost unique independently with the limited graphics elements. After longer playtime all 256 rooms are almost recognised. But there is no sound, no noise. First time it was annoying (I have no huge expectations), I should like to hear any sound effect at firing or walking. After some played days this fault went away...

Last game of Fantasy has perfect control. The sensibility is top quality, maybe there is a little problem when Ziggy walks and reaches the end of a platform. Sometimes he stuck a moment like he would have fear of the deep. And one more: if I press BREAK after picking up an object to hyper jump back to my ship, sometimes Ziggy left an object in the box. This can be solve by leaving the box somewhere before pressing BREAK.

The game doesn't expect very high skill but the logic and the making of good strategy is important. Summarize it is a hard, long play. In this type of games are very important the entertainment for the players who can quit very quickly if it is bored. Unfortunately the program has some bugs what limits the players. The repairing should be important, without these the addictivity of the game was low, it gave hard moments for the players and made impossible to finish the game. Positive point is having save-load option so the game can be continued later. Negative point is the missing Part 2 and 3 and we never enjoyed Ziggy's adventure.



RATINGS

Pgyuri

Must be grown up to this game. Like a child the rooms were so same, there were too many keys and everything was understandable due to missing English knowledge. Independent of these all looked nice, colourful and exciting. In adult age the labyrinth is clear-cut and learnable. The constant place of objects made the game to a chess challenge where we had to find out the correct order of steps. First time all thing looked rambling, too complex, there were too much connected data. Going forward we knew the game better and the tasks looked cleaner, we saw the light at the end of tunnel.

Summarize the play was good entertaining. Finding and correcting the back-pack bug without peeking anything about the game was hard job. We could solve it so finishing the challenge was a great adventure with my friend. We can describe some criticisms:

- no help about the doors and theirs keys, just trying
- no help about the usable TNTs, just trying
- no help about the objects, just thinking about the pictures
- the status panel has extremely weak solution, very annoying
- too much walking due to the labyrinth has no circle-way

The published map in the magazine gave almost all information, so the time-waster tryings was kicked out. Pity, it was necessary, some graphic solution could help to the players and they could feel success.

However, the game is praised but don't forget we walked through. The control is perfect, the silent is absolutely acceptable, was good feel walking around, collecting objects and creatures. The level of the challenge was extremely high. The final congratulation is missing very much in turn he programmer knew the game would not be continued ever.



TIPS & TRICKS

#1 - THE HAND TRICK

Ziggy has only 4 items (key, disc, tnt) in same time. Sometimes looks it is not enough. For example opening a door what needs 4 keys putting into the keyhole after each other. Then bring out a key what is locked behind it. But how to make it without lose any key:












1. Put 4 keys into the holder 1 to open the door.
2. Leave one key in holder 2.
3. Pick up the locked key (holder 3).
4. Put 4 keys again into holder 2 to open the door again.
5. Leave one key in holder 1.
6. Pick up the key from holder 2.
7. Go out and all 5 keys are out.

Never forget the key outside of the door while working inside or will stuck and loose 1 hour by pressing BREAK. It works almost same with special 4 crystals teleporting, see in Trick #3 at Step 14.

#2 - TNT GUIDE

Can be found seven explosives and seven blasting spots in the labyrinth. In a blasting position, any dynamite causes a destruction, the damage caused is maybe too weak, maybe too big, or in optimal case maybe correct. In most positions have two good choices, except where

							
	✓	<	<	<	<	<	<
	✓	✓	<	<	<	<	<
	>	✓	✓	<	<	<	<
	>	>	✓	✓	<	<	<
	>	>	>	✓	✓	<	<
	>	>	>	>	✓	✓	<
	>	>	>	>	>	✓	✓

TIPS & TRICKS






white dynamite is to be used. So there is a choice, but if you're getting better in the game then will see that this is a very relative freedom. In the table, the columns are the blasting positions, the rows are explosives. For example: blue and red dynamites can use in blue position.



The pictures show the too small, correct and too big destruction.

#3 - THE WAY OF THE TALISMAN

There are 2 parts of the talisman in the labyrinth. They must be collected to finish the game in 100 %. There are 14 keys in 4 colors and with 5 signs, the signs are:

-  C (key with zig-zag lines)
-  F (key with cog sign)
-  K (key with + sign)
-  R (key with lozenge sign)
-  X (key with X sign)

And there is also a special key, the purple EXIT key: 

Steps to acquire the talisman's pieces, print out the original map and see, how to do it:

1. Looking for purple EXIT key using discs and put it into L4 (door will open but do nothing)
2. Pick up yellow crystal (L9) >> pick up red TNT (C3) >> press BREAK
3. Pick up yellow K key (C11) >> pick up cyan TNT (G8) >> pick up yellow C key (A1)
4. Pick up blue F key (C12) [the door at E13 can open by yellow K key]
5. Pick up purple TNT (N13) >> pick up green R key (J6) >> pick up green F key (N7) >> put green R key to O7 (outside the door)

TIPS & TRICKS

6. Pick up yellow F key (M14) >> put yellow C key to N14 (outside the door)
 7. Put yellow K key to N13
 8. Pick up purple F key (K4) >> put purple EXIT key to L4 (outside the door)
 9. Pick up green X key (I5) >> put blue F key to J5 (outside the door)
 10. Pick up purple crystal (K3) >> put green X key there
 11. Pick up yellow crystal (F3) >> put purple F key there
 12. Pick up blue crystal (K11) >> put yellow F key there
 13. Pick up green crystal (H13) >> put green F key there
 14. Put yellow crystal to I3 >> pick up green K key (K3) >> put crystals in row to I3 >> pick up talisman (I4) >> door (H3) open with green X key
 15. Pick up yellow K key (N13) >> put green X key there
 16. Pick up white TNT (L15) >> put purple crystal there >> pick up yellow X key (C11)
 17. Pick up purple crystal (L15) >> put yellow X key there
 18. Pick up blue TNT (M6) >> put yellow K key there
 19. Go to teleport (F3), but set anywhere the starting crystal to blue then green then purple >> teleport to blue area
 20. Pick up yellow R key (D10) >> pick up green TNT (F8) >> put purple F key there >> teleport to green area
 21. Door (K14) open with yellow R key
 22. Just blow up TNT at green C key (J12)
 23. Pick up yellow K key (M6)
 24. Pick up yellow C key (N14)
 25. Pick up yellow X key (L15) >> put purple crystal there
 26. Pick up green K key (E15) (use 4 yellow keys in row to open the door (F15)) >> put yellow R key to F15 (outside the door)
 27. Pick up green R key (O7) >> put yellow K key there
 28. Pick up green C key (J12) >> put yellow C key there
 29. Pick up green X key (N13) >> put yellow X key there
 30. Use 4 green keys in row to open door (O11) >> pick up talisman (N11)
- The position of the items after 30 steps:

?

TIPS & TRICKS

purple EXIT key:	L4	blue F key:	J5
purple F key:	F8	blue C key:	O15 (useless)
yellow F key:	K11	green F key:	H13
yellow C key:	J12	green C key:	in hand
yellow R key:	E15	green R key:	in hand
yellow X key:	N13	green X key:	in hand
yellow K key:	O7	green K key:	in hand
yellow crystal:	I3	yellow TNT:	K1 (useless)
purple crystal:	L15		
blue crystal:	F3		
green crystal:	E8		

This solution is the "simple" way to access the 2 parts of talisman. It is not enough to continue playing for collecting all creatures. Must combine the ways with collect some objects and creatures or the 12 hours won't be enough.

#4 - THE POSITION OF THE PURPLE EXIT KEY



The purple EXIT key position will be calculated after leaving the START room when the door has closed. It will put only any place of indicators (where the discs work) and if it is taken, the object moves from there to the START room. Worthy to see back after leaving START room it can help to find the EXIT key easier.

#5 - COLLECTING CREATURES

To collect any creature looks easy after the way of talisman due to all keys are accessible but don't forget, every creature wants to its object or will die very soon. Finding the best way for a creature is almost easy but there are some problems:

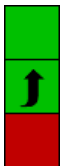
- Just 5 times can go back to START with living creature(s) in the backpack (okay, GAME OVER is not working there)
- The way can be long to keep creatures alive, pressing BREAK is good but +1 hour punishment

And finally some creatures don't like other, kill it as soon as possible!

TIPS & TRICKS

This table shows the relationship between creatures, the meaning of the markings:

	EMU	QUALLY	FLUFFLELUMP	ICE MONSTER	SNOTTOID	URK	GOBHOBLIN	FLOOK	SPINY NORMAN	GOOGLY BIRD	UNIDRAGODILE	PRICKLEPUSS
EMU			↑		↑	☒	↑	↑	↑			
QUALLY			☒	↑	↑	↑		☒	↑	☒	↑	↑
FLUFFLELUMP	☒	☒			↑	☒	☒				☒	↑
ICE MONSTER		☒				☒	☒			☒	↑	
SNOTTOID	☒	☒	☒			☒	↑		☒	☒		☒
URK	↑	☒	☒	☒	☒		☒		↑		↑	☒
GOBHOBLIN	☒		☒	↑	☒	↑		☒	↑	☒		↑
FLOOK	☒	↑					↑		☒	☒	↑	
SPINY NORMAN	☒	☒			↑	☒	☒	↑			☒	☒
GOOGLY BIRD		☒		↑	☒		☒	☒				☒
UNIDRAGODILE		☒	☒	☒		☒		☒	☒			↑
PRICKLEPUSS		☒	☒		☒	↑	☒		↑	☒	☒	



The creatures like each other in any order.

The order is important, for example: URK+EMU correct,
EMU+URK deadly.

Deadly pairs.

BUGS

#1 - THE MAIN BUG

The goal of the game is to collect 12 creatures in the labyrinth and carry them out to the START position. After pick up a creature, it has only short life-time. All creature has an object what is important to keep it alive. These items extend the energy of creatures.

The best way would be to collect object(s) before pick up a creature but unfortunately the program overwrites the energy at pick up point and loses the previous collected energy. That's why many creatures are absolutely uncollectable and the challenge is impossible.

#2 - THE BACKPACK RETRIEVED BUG

If any creature is in the backpack after go back to START position (using purple EXIT key or pressing BREAK), all collected objects energy loose, the program overwrites theirs value to 0. It is a real problem, because if any object losts, the creature will die later and the game is impossible to finish.

Repair this bug is complex and not necessary, just keep in mind to collect only those objects what are useful for the chosen creatures.

#3 - THE 'GAME OVER' BUG

The manual is declared the game is over if Ziggy uses more than 12 hours for the challenge or retrieves his backpack with living creatures 5 times. It is not true, the GAME OVER is only depends on time. The program checks the counter to 5 but that never achieve this concrete value.

Repair this bug is not necessary.

#4 - THE 256TH VISITED ROOM

The visited rooms are counted between 2 running from START position. All 256 rooms visiting in one time will be printed "000".

Repair this bug is not necessary, the visited rooms counter is a red her-ring.

BUGS

#5 - THE 'WELL DONE' BUG

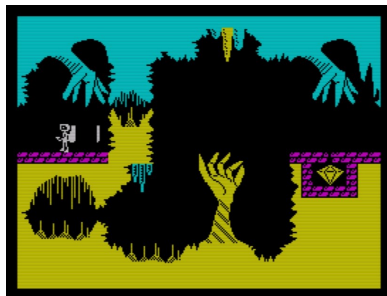
When the game is finished in 100 %, the labyrinth stays opened and there is no any congratulation. The manual refers to saving state is needed for the planned but never made Part 2 but it was not fair for players.

#6 - THE SPECTRUM 128K BUG

The code sheet is not working on 128K Spectrum.

#7 - GRAPHICS BUG

Ziggy loses the single piece of his backpack after teleport.



#8 - COLOUR ANOMALIES

The N10 room is cyan. This it not correct for several things. First, because the door in the cyan ground room is opened by a green key (the ground shows the key colour everywhere). On the other hand, the colour error on the map is quite noticeable.



The left image shows the original colouring (in the centre, the cyan part is the N10 room), the right-hand image is in principle the right colouring. There are two differences between the two images, because the corner of the neighbouring (010) room is rather yellow.

Observing the map, there are several colour anomalies: in rooms D4, I12, L11, K6, L3, N3, N10, O3, O10 and P10. But there is no problem in either place.

BUGS

Another interpretation is that there is no problem with the cyan colour of the room, then the door should be opened with a cyan key (for example the useless cyan C key).

#9 - THE RED TNT POSITION

The red TNT is accessible only using the teleport with yellow crystal.

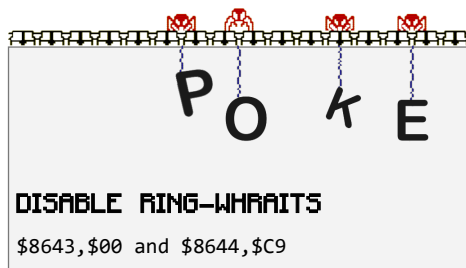
This is a closed area with one way in (using yellow crystal) and one way out (using yellow C key). The door can be opened only inside.

See the problem:

Standing in the yellow teleport area.

- <--- before must be collected the yellow C key (A1) using cyan TNT
- <--- before must be collected cyan TNT (G8) using yellow K key
- <--- before must be collected yellow K key (G11) using red TNT
- <--- before must be collected red TNT (C3) but it is in the yellow teleport area.

So going for the red TNT into the yellow teleport is a stuck point, just escape only by pressing BREAK what is adding 1 hour punishment. Very rare time the red TNT jumped out from its place to the start point, just see the Trick #4.



PROGRAMMER'S CORNER

#1 - THE MAIN BUG

This procedure works if backpack was retrieved with a living creature.

THE WRONG CODE:

```
$9281 $21,$34,$92 LD HL,$9234 ; the life-time table of creatures
$9284 $06,$00     LD B,0      ; set 'B' register to 0, 'C' register is
                  ; the creature number.
$9286 $09         ADD HL,BC   ; calculate the creature data position
$9287 $56         LD D,(HL)   ; default energy of the creature store in
                  ; 'D' register
$9288 $21,$4C,$92 LD HL,$924C ; backpack data
$928B $09         ADD HL,BC   ; calculate the creature data position in
                  ; the backpack
$9928 $72         LD (HL),D   ; overwrite the stored energy with the
                  ; creature default energy the objects'
                  ; energy what were taken before are
                  ; stored at same position in the backpack
                  ; with its' creature.
```

Would be nice to ADD the creature energy to the stored energy but it is a complex problem due to some objects have \$FF energy.

It must be checked before ADD, but here is not enough place. A simple OR is the best way, it keeps the energy and calculate with the creature's energy and handles the \$FF value too. See below.

But it is longer only 1 byte. The hacker's luck enters here. The creature and backpack table high byte is same \$92 (stored in 'H' register) so just enough to change the 'L' register in the code:

THE CORRECT CODE:

```
$9287 $7E         LD A,(HL)
$9288 $2E,$4C     LD L, $4C   ; phew, spare 1 byte
$928A $09         ADD HL,BC
$928B $B6         OR (HL)   ; spared 1 byte needs here
$928C $77         LD (HL),A
```

PROGRAMMER'S CORNER

#2 - THE BACKPACK RETRIEVED BUG

This procedure drops out ALL things from the backpack.

THE WRONG CODE:

```
$9E94 $21,$4C,$92 LD HL,$924C ; backpack data
$9E97 $06,$0C     LD B,$0C   ; 12 creatures can be in backpack
$9E99 $3E,$00     LD A,0    ; why not XOR A?
$9E9B $77         LD (HL),A  ; overwrite the energy of those object what
                        ; were collected, but the creature was not
                        ; in the backpack
$9E9C $23         INC HL    ; step to next backpack data
$9E9D $10,$FC     DJNZ $9E9B ; loop 12 times
```

This point should be checked the creature's status but it is missed. Not necessary but the player must be careful to collect only those object what is needed.

#3 - THE 'GAME OVER' BUG

This program part checks one of Game Over condition.

```
$9E9F $21,$DF,$9B LD HL,$9BDF ; counter of the lucky backpack retrievals
                        ; (there was living creature)
$9EA2 $34         INC (HL)  ; increment the counter
$9EA3 $7E         LD A,(HL) ; check the counter value
$9EA4 $FE,$05     CP $05    ; is it 5?
$9EA6 $20,$02     JR NZ,$9EAA ; no, continue playing
$9EA8 $CB,$F6     SET 6,(HL) ; GAME OVER sign
```

After writing this part the programmer forgot it and used the counter higher bits storing some game status. That's why it has never 5 value, a simple command was missed after INC (HL):

```
AND 7 ; just use the count part
      ; (drop status bits)
```

#5 - THE 'WELL DONE' BUG

Unfortunately the game has no end when 100% done. This small program part can give a little satisfaction:

PROGRAMMER'S CORNER

ORIGINAL CODE:

```
$9C3A $CD,$FF,$B3 CALL $B3FF ; first must redirect a calling point
```

MODIFIED CODE:

```
$9C3A $CD,$FF,$B3 CALL $B183 ; the original calling write the main text  
; to the screen it has changed to call our  
; WELL DONE checking routine
```

CHECKING ROUTINE:

```
$B183 $CD,$FF,$B3 CALL $B3FF ; write the main text to the screen  
$B186 $3A,$DE,$9B LD A,($9BDE) ; load actual percent value to 'A'  
; register  
$B189 $FE,$64 CP $64 ; is it 100 %  
$B18B $C0 RET NZ ; no, do nothing  
$B18C $21,$97,$B1 LD HL,$B197 ; set the WELL DONE text address in memory  
$B18F $CD,$FF,$B3 CALL $B3FF ; write "WELL DONE" text to the screen  
$B192 $AF XOR A ; set 'A' register to 0  
$B193 $32,$7C,$79 LD ($797C),A ; lock the main door  
$B196 $C9 RET ; return to the calling point  
  
$B197 $01,$05,$00,$30,$40 ; text data, positions and colours  
$B19B $57,$45,$4C,$4C,$3C, ; "WELL<DONE" text  
$44,$4F,$4E,$45,$FF ; and end text sign
```

#6 - THE SPECTRUM 128K BUG

The code sheet is not working on 128K Spectrum.

The game is decode the entry algorithm with the first 908 bytes of ROM. Unfortunately the ROM is different in only 2 bytes on 128K machine both mode (128K or 48K) than the original 48K. It changes the program code but lucky in the accident it counts only different value than what can be found in code sheet.

How to solve ?

The calculating of the entry code depends on the FRAMES system variables. The main idea was simple. Just POKE 23672,0 and POKE 23673,0 in the BASIC loader, then always same value will calculated. Unfortunately the BASIC interpreter working time depends on many environments and many times it gives different running time. It looks dead-end and the loading system doesn't show any other way for correct working on 128K machine.

PROGRAMMER'S CORNER

Finally it was easy to repair. Just insert 7 bytes into the loader and working fine, always give same entry code on both machines (A0).

THE ORIGINAL LOADER:

```
1 PAPER 0: INK 7: BORDER 0: CLS
2 FOR X=0 TO 25: READ c: POKE 53731+x,c: NEXT x
4 CLEAR 65535: RANDOMIZE USR 53731
5 DATA 17,119,28,221,33,0,64,55,62,255,205,86,5,17,134,163,221,33,122,
  92,55,62,255,205,86,5
```

THE MODIFIED LOADER (the inserted part is red):

```
1 PAPER 0: INK 7: BORDER 0: CLS
2 FOR X=0 TO 32: READ c: POKE 53724+x,c: NEXT x
4 CLEAR 65535: RANDOMIZE USR 53724
5 DATA 17,119,28,221,33,0,64,55,62,255,205,86,5,175,50,120,92,50,
  121,92,134,163,221,33,122,92,55,62,255,205,86,5
```

OPERATIONS OF THE LOADER:

```
LD DE,$1C77
LD IX,$4000
SCF
LD A,$FF
CALL $0566      ; LOAD screen + some code
XOR A
LD ($5C78),A    ; same with POKE 23671,0 (set FRAMES to zero)
LD ($5C79),A    ; same with POKE 23672,0 (set FRAMES to zero)
LD DE,$A386
LD IX,$5C7A
SCF
LD A,$FF
CALL $0566      ; LOAD the main program
                  ; the entry code will be calculated with always
                  ; same FRAMES system variable
```

#8 - COLOUR ANOMALIES

There are two solutions to the colour problem of the N10 room:

Change the soil color to green: \$6575,\$F5

Change green R key to cyan C key: \$7F39,\$15 and \$7F3D,\$15

HACK ATTACK

The game has no anti-piracy copying system (except the anti-copied coloured code sheet), but the loading scheme makes impossible using of any POKE. If do not have got Multiface or similar hardware which can access the memory during running time, have to modify the program a little bit and all POKE will be usable.

Work with the original game:

Program: Packers	
BASIC program	358 byte
headerless block	7287 byte
headerless block	41862 byte

Fortunately the BASIC program can be loaded and stopped with MERGE "" command, so the program can be listed:

```
LS 1 PAPER 0: INK 7: BORDER 0: C
2 FOR X=0 TO 25: READ C: POKE
53731+X,C: NEXT X
4 CLEAR 65535: RANDOMIZE USR
53731
5 DATA 17,119,28,221,33,0,64,
55,82,255,205,86,5,17,134,153,22
1,33,122,92,55,62,255,205,86,5
```

The 1st line sets the colours to black.

The 2nd line pokes the loading program from 53731 address by a FOR cycle.

The 4th line sets the RAMTOP by CLEAR command and starts the loading calling by RANDOMIZE USR 53731.

The 5th line keeps the DATA of the original loader.

Here is not used any BASIC number trick like in Speedlock or any other, all value are cor-

rect. Just type 3 STOP and after running the loader can be disassembled using MONS3 or something else.

```
LD DE,7287 ; Part 1 overwrites the system variables after loading
LD IX,16384 ; the intro screen (16384 to 23670)
SCF
LD A,255
CALL 1366
LD DE,41862 ; and the next part loads bytes to the end of memory
LD IX,23674 ; (23674 to 65535). That's why doesn't work any helping
SCF ; POKE in the BASIC program.
LD A,255
CALL 1366
```

HACK ATTACK

How will start the game however there is no more command after loading?

The second part fill the whole memory what meanings the stack will be overwritten. Here is stored that address where to the ROM loading routine (1366) must return. So after loading the game will start simply. But where is this return address exactly ?

Have to know where is the stack-pointer after loading the intro screen. Perhaps the CLEAR command sets it to 65535 but the BASIC system changes its value. Write this small program:

```
32768 237,115,254,127 LD (32766),SP ; storing the stack pointer (SP reg.)
32772 201             RET           ; at 32766-32767 address
```

In BASIC:

```
10 CLEAR 65535
20 POKE 32768,237: POKE 32769,115: POKE 32770,254: POKE 32771,127:
   POKE 32772,201
30 RANDOMIZE USR 32768
40 PRINT (256*PEEK 32767)+PEEK 32766
```

and look, the stack-pointer shows the 65512 address.

The game will read the return address from 65510 address. The concrete value is not important now just must knowing the actual value of stack-pointer.

The program has simple Spectrum loading scheme, not turbo or any special, can be loaded with original style. The main trick is modifying the position of the loader in the memory, moving to the first part of screen (it will disappear) and after full loading the memory will be changed anywhere.

```
53731 17,119,28      LD DE,7287
53734 221,33,0,64    LD IX,16384
53738 55             SCF
53739 62,255         LD A,255
53741 205,86,5       CALL 1366           ; loading Part 1.
53744 49,0,72        LD SP,18432       ; setting the stack pointer new place
                                     ; else the game will start like the
                                     ; original

53747 33,1,210       LD HL,53761
53750 17,0,64        LD DE,16384
53753 1,0,8          LD BC,2048       ; moving the loader program of the
53756 237,176        LDIR           ; Part 2 and the poker to the screen
53758 195,0,64       JP 16384         ; starting loading
```

HACK ATTACK

```
53761 17,1134,163 LD DE,41862 ; it will run at its new place
; (16384)
53764 221,33,122,92 LD IX,23674
53768 55 SCF
53769 62,255 LD A,255
53731 205,86,5 CALL 1366 ; loading Part 2.
53774 49,230,255 LD SP,65510 ; setting the stack-pointer to the
; original place what is less with 2
; than the checked value (after CALL
; the stack pointer decrement by 2
; and will return to the address what
; is stored at 65510.
53777 62,0 LD A,0 ; the BASIC line 3 will poke these
53779 50,67,134 LD (34371),A ; commands
53782 62,201 LD A,201
53784 50,68,134 LD (34372),A ; turning off ring-whraits
53787 201 RET ; return, starting the game
```

Write the data of pokes in decimal format in the line 6 like this (for ex. want to turn off the ring-whraits with 34371,0 and 34372,0). Put the value loader cycle in the line 3 where the final value of the FOR cycle is the number of data in the 6th line divide by 2, now $4/2=2$ and modify the line 2 and line 5 at the red points:

```
1 PAPER 0: INK 7: BORDER 0: CLS
2 FOR X=0 TO 45: READ c: POKE 53731+x,c: NEXT x
3 LET M=53777: FOR X=1 TO 2: READ C: POKE M+2,50: POKE M+3,
  C-INT(C/256)*256: POKE M+4,INT (C/256): READ C: POKE M,62:
  POKE M+1,C: LET M=M+5: NEXT X: POKE M,201
4 CLEAR 65535: RANDOMIZE USR 53731
5 DATA 17,119,28,221,33,0,64,55,62,255,205,86,5,49,0,72,33,1,210,17,
  0,64,1,0,8,237,176,195,0,64,17,134,163,221,33,122,92,55,62,255,205,
  86,5,49,230,255
6 DATA 34371,0,34372,201
```

Save the BASIC program to have and put more pokes anytime later: SAVE "Packers+" LINE 1

Run the program: RUN

Skip the BASIC program on the cassette and press play. The intro screen will load then the top part of the screen will disappear (the 'secret' program will go there). The loading will continue and finally the game will start.