**Ниже Вы можете найти аналог очень распространённой НАСТОЛЬНОЙ игрушки – СЛОЖИ ПАЗЛ. Для того, чтобы «поиграть», необходимо «покопать». Сегодня – это, типа, Ваша «песочница».**

**То, что представлено далее, ЯВЛЯЕТСЯ ЭЛЕМЕНТАМИ ПАЗЛА - СОВОКУПНОСТЬЮ ПЕРЕМЕШАННЫХ ФРАГМЕНТОВ ТЕКСТА на ЯЗА ассемблер. Если его правильно сложить, то получится текст, из которого после трансляции, линковки, получается исполнимый модуль, который в свою очередь будучи загруженным в ОП превращается в программу, которая уже заставляет процессор выполнить действия:**

**1. Принять от пользователя пароль (даётся три попытки, причём количество попыток уменьшается каждый раз, когда пароль набран неверно, И !!!!! даже в случае прерывания программы после ввода неверного пароля – то есть число попыток уменьшается даже тогда, когда программа прерывается нехорошим человеком-редиской-хакером, после неверно набранного пароля. Типа он думает, что если прервать программу, то у него будет не три попытки, а бесконечность…)**

**2. Если пароль введён три раза неверно, исполнимый модуль «ИЗНИЧТОЖАЕТСЯ» из бренного мира компьютера, то есть исполнимый модуль «портится» на диске (не убивается, а портится) и имя его удаляется из реестра файлов файловой системы. ОБМАНУЛ?**

**3. Если пароль набран верно в какой-либо из трёх попыток, то на экран выводится последовательность неких символов, которые Вы увидите на экране и которые нужно сосчитать.**

**То есть нужно «СЛОЖИТЬ ПАЗЛ», чтобы всё получилось. Частички пазла – фрагменты текста ОТДЕЛЕНЫ ДРУГ ОТ ДРУГА несколькими пустыми строками и взяты в прямоугольники. Можно сказать, что дедушка Вас пощадил в том смысле, что он мог ,s перемешать всё-всё, сделав элементом пазла одну строку из исходного текста. Удачи.**

START:

MOV AX,@DATA

MOV DS,AX

MOV AX,ES:[2CH]

MOV ES,AX

MOV SI,-1

SEARCH\_01:

ADD SI,1

MOV AL,ES:[SI]

CMP AL,0

JNE SEARCH\_01

MOV AL,ES:[SI+1]

CMP AL,1

JNE SEARCH\_01

ADD SI,2

MOV BX,OFFSET PROGRAM\_NAME

.CODE

PASSWORD DB 'PASSWORD'

PASSWORD\_LENGTH EQU $-PASSWORD

DELETE\_FILE:

MOV AH,3EH

INT 21H

MOV DX,OFFSET PROGRAM\_NAME

MOV AH,41H

INT 21H

MOV DX,OFFSET GOODBYE

MOV AH,09H

INT 21H

MOV AX,4C00H

INT 21H

END START

PRINT\_10\_SYMBOL\_LINE:

MOV AH,02H

INT 10H

MOV AH,09H

INT 10H

INC DH

CMP DH,25

JNE PRINT\_10\_SYMBOL\_LINE

MOV AX,4C00H

INT 21H

CHECK\_PASSWORD:

MOV DX,OFFSET ATTEMPTS\_LEFT

MOV AH,09H

INT 21H

MOV AL,ATTEMPTS

ADD AL,30H

INT 29H

MOV AL,10

INT 29H

MOV AL,13

INT 29H

MOV DX,OFFSET ENTER\_PASSWORD

MOV AH,09H

INT 21H

MOV DX,OFFSET ENTERED\_PASSWORD

MOV AH,0AH

INT 21H

MOV AL,10

INT 29H

MOV AL,13

INT 29H

.MODEL SMALL

.DATA

PROGRAM\_NAME DB 256 DUP(?)

ENTER\_PASSWORD DB 'Enter password:',10,13,'$'

ENTERED\_PASSWORD DB 11,12 DUP(?)

MOV BX,OFFSET ENTERED\_PASSWORD

MOV AL,[BX+1]

XOR AH,AH

MOV BX,PASSWORD\_LENGTH

CMP AX,BX

JNE WRONG\_PASSWORD

MOV CX,AX

MOV SI,OFFSET PASSWORD

MOV DI,OFFSET ENTERED\_PASSWORD

ADD DI,2

COMPARE\_PASSWORDS:

MOV AL,CS:[SI]

MOV AH,[DI]

CMP AL,AH

JNE WRONG\_PASSWORD

INC SI

INC DI

LOOP COMPARE\_PASSWORDS

COPY\_NAME:

ADD SI,1

MOV AL,ES:[SI]

MOV [BX],AL

INC BX

CMP AL,0

JNE COPY\_NAME

WRONG\_PASSWORD:

MOV AL,ATTEMPTS

DEC AL

CMP AL,0

JE DESTROY

MOV ATTEMPTS,AL

CALL WRITE\_ATTEMPTS

JMP CHECK\_PASSWORD

CORRECT\_PASSWORD DB 'Correct password!',10,13,'$'

GOODBYE DB 'Self-destruction!',10,13,'$'

ATTEMPTS DB 3

ATTEMPTS\_LEFT DB 'Attempts left: $'

DESTROY:

MOV DX,OFFSET PROGRAM\_NAME

MOV AX,4300H

INT 21H

AND CL,11111110B

MOV AX,4301H

INT 21H

MOV AX,3D02H

INT 21H

MOV BX,AX

MOV CX,0

MOV DX,0

MOV AX,4202H

INT 21H

MOV SI,AX

MOV ATTEMPTS,0

MOV CX,0

MOV DX,0

MOV AX,4200H

INT 21H

WRITE\_ATTEMPTS PROC

MOV DX,OFFSET PROGRAM\_NAME

MOV AX,4300H

INT 21H

MOV AX,4301H

AND CX,11111110B

INT 21H

MOV AX,3D02H

INT 21H

MOV BX,AX

MOV CX,0

MOV DX,0

MOV AX,4202H

INT 21H

MOV CX,0

SUB AX,17

MOV DX,AX

MOV AX,4200H

INT 21H

MOV DX,OFFSET ATTEMPTS

MOV CX,1

MOV AH,40H

INT 21H

MOV AH,3EH

INT 21H

RET

WRITE\_ATTEMPTS ENDP

MOV ATTEMPTS,3

CALL WRITE\_ATTEMPTS

MOV AH,0FH

INT 10H

MOV AL,10

MOV BL,7

MOV CX,80

MOV DX,0

CLEAR\_FILE:

MOV DX,OFFSET ATTEMPTS

MOV CX,1

MOV AH,40H

INT 21H

DEC SI

CMP SI,0

JE DELETE\_FILE

MOV CX,0

MOV DX,1

MOV AX,4201H

INT 21H

JMP CLEAR\_FILE

В том, что представлено выше, в случае правильного ввода пароля на экран выводится сто? символов «рожица» или не «рожица»?. Попробуйте ЗАМЕНИТЬ вывод рожиц НА ВЫВОД 4 000 символов $. Вам в помощь представлен текст (ещё одна свинья), из которого можно «выщемить» ВЫВОД НА ЧЕРНЫЙ-ЧЕРНЫЙ ЭКРАН 4000 символов $ (ОБЫЧНО НА ЭКРАНЕ МОЖНО РАЗМЕСТИТЬ 2000 символов в текстовом режиме!!!!). По крайней мере «ОТЛИЧНИКИ» прошлых лет утверждали, что выводится 4 000 символов $.

FONT DB 16 DUP(?)

CORRECT PROC

MOV BX,OFFSET FONT

MOV BYTE PTR [BX], 01000100B

MOV BYTE PTR [BX+1], 01000100B

MOV BYTE PTR [BX+2], 11101110B

MOV BYTE PTR [BX+3], 10101010B

MOV BYTE PTR [BX+4], 10101010B

MOV BYTE PTR [BX+5], 10001000B

MOV BYTE PTR [BX+6], 11101110B

MOV BYTE PTR [BX+7], 00100010B

MOV BYTE PTR [BX+8], 10101010B

MOV BYTE PTR [BX+9], 10101010B

MOV BYTE PTR [BX+10],11101110B

MOV BYTE PTR [BX+11],01000100B

MOV BYTE PTR [BX+12],01000100B

MOV BYTE PTR [BX+13],00000000B

MOV BYTE PTR [BX+14],00000000B

MOV BYTE PTR [BX+15],00000000B

MOV AX,DS

MOV ES,AX

MOV BP,OFFSET FONT

MOV CX,1

MOV DX,'$'

MOV BX,1600H

MOV AX,1100H

INT 10H

MOV CX,1999

MOV AL,'$'

PRINT:

INT 29H

LOOP PRINT

MOV AX,40H

MOV ES,AX

MOV BH,ES:[62H]

MOV AH,0AH

MOV AL,'$'

MOV CX,1

INT 10H

P:

JMP P

MOV AX,4C00H

INT 21H

MOV AH,3EH

INT 21H

RET

CORRECT ENDP

ПРОСЬБА. КОПАЙТЕ САМОСТОЯТЕЛЬНО!!!!

Нет ЧРТ

***ИНТЕРЕСНО, И ЧТО НАПИСАНО В ПРЯМОУГОЛЬНИКЕ!!!!!!????????***