National University of Computer and Emerging Sciences



**TITLE: PROJECT PROPOSAL OF COMPUTER ORGANIZATION AND ASSEMBLY LANGUAGE**

**TEAM NAME: BRAINLESS CODERS**

**TEAM MEMBERS:**

1. **MUHAMMAD HUZAIFA (18F-0331)**
2. **FAIQ HUMAYUN (18F-0288)**
3. **ABDULLAH MUGHAL (18F-0314)**

INTRODUCTION:

We are presenting a game in our project which is named as a Word Hunt. It is type of game whose result totally depends on player’s attention. In this game the player is displayed an already drawn board from which player tries to find the hidden word. Each time the player enters a word, the computer tells him whether the word id false or right.

Features:

1. You can play game multiple times if you want.
2. You will be given a output after each input.
3. Your should find 4 words.
4. You will be given 5 points for each right guess.

**SOURCE CODE:**

include irvine32.inc

.data

menu db "Main menu ",0Ah,0h

op1 db "Press 1 to play ",0Ah,0h

str0 db " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_",0Ah,0h

str1 db " |v a m n o u d h e w|",0Ah,0h

str2 db " |i v k l t z s d o w|",0Ah,0h

str3 db " |e i w s d j i k a z|",0Ah,0h

str4 db " |w e r x f h p k e w|",0Ah,0h

str5 db " |a q b c y g w l x o|",0Ah,0h

str6 db " |b s t a c k i e w l|",0Ah,0h

str7 db " |z k o j i l h w p x|",0Ah,0h

str8 db " |m i x f d u q o c p|",0Ah,0h

str9 db " |g l t v c d w p x i|",0Ah,0h

str10 db " |f m a a m m o k w m|",0Ah,0h

str11 db " |- - - - - - - - - -|",0Ah,0h

input db 6 dup(?)

key1 db "view",0h

key2 db "stack",0h

key3 db "maam",0h

key4 db "mix",0h

error db "Wrong input.",0Ah,0h

count db 0h

count1 db 0h

msg db "You have successfully found one of the four word!.",0Ah,0h

result db "You have got points : ",0h

.code

main proc

Call main\_menu

Call table

Call Takeinput

Call compare

Call Takeinput

Call compare

Call Takeinput

Call compare

Call Takeinput

Call compare

mov edx,OFFSET result

call WriteString

mov eax,0h

mov al,5h

mov bl,count1

mul bl

Call WriteDec

exit

main endp

main\_menu PROC

MOV DH, 3

MOV DL, 45

CALL GoToXY

MOV EDX, OFFSET menu

CALL WriteString

MOV DH, 5

MOV DL, 45

CALL GoToXY

MOV EDX, OFFSET op1

CALL WriteString

call readchar

call clrscr

RET

main\_menu ENDP

table proc

mov edx,OFFSET str0

Call WriteString

mov edx,OFFSET str1

Call WriteString

mov edx,OFFSET str2

Call WriteString

mov edx,OFFSET str3

Call WriteString

mov edx,OFFSET str4

Call WriteString

mov edx,OFFSET str5

Call WriteString

mov edx,OFFSET str6

Call WriteString

mov edx,OFFSET str7

Call WriteString

mov edx,OFFSET str8

Call WriteString

mov edx,OFFSET str9

Call WriteString

mov edx,OFFSET str10

Call WriteString

mov edx,OFFSET str11

Call WriteString

ret

table endp

Takeinput proc

mov edx,0h

mov edx,OFFSET input

mov ecx,6h

Call readString

ret

Takeinput endp

compare proc

mov ebx,0h

mov eax,0h

mov esi,OffSet input

mov edi,Offset key1

mov al,4

mov count,al

;

l1:

mov bl,[edi]

cmp [esi],bl

je l2

mov ecx,1h

jmp l3

l2:

inc esi

inc edi

mov ecx,2h

dec count

cmp count,0h

je l4

loop l1

l3:

;

mov ebx,0h

mov esi,OffSet input

mov edi,Offset key2

mov al,5

mov count,al

l6:

mov bl,[edi]

cmp [esi],bl

je l7

mov ecx,1h

jmp l8

l7:

inc esi

inc edi

mov ecx,2h

dec count

cmp count,0h

je l4

loop l6

l8:

;

mov ebx,0h

mov esi,OffSet input

mov edi,Offset key3

mov al,4

mov count,al

l9:

mov bl,[edi]

cmp [esi],bl

je l10

mov ecx,1h

jmp l11

l10:

inc esi

inc edi

mov ecx,2h

dec count

cmp count,0h

je l4

loop l9

l11:

;

mov ebx,0h

mov esi,OffSet input

mov edi,Offset key4

mov al,3

mov count,al

l12:

mov bl,[edi]

cmp [esi],bl

je l13

mov ecx,1h

jmp l14

l13:

inc esi

inc edi

mov ecx,2h

dec count

cmp count,0h

je l4

loop l12

l14:

;

mov edx,OFFSET error

Call WriteString

jmp l15

l4:

mov edx,offset msg

Call WriteString

inc count1

l15:

ret

compare endp

end main



