Ayesha Mehmood
ID # 6832

Assignment # 05

Assignment NO 5

Question NO 1:-

Q Which register (in 32-bit mode) manages the stack;

Ans :- Extended stack pointer.

Q:- Why is the stack called a LIFO structure;

Ans:- It is called LIFO structure because it contain set of memory blocks in which data is retrieved in order i-e the last value pushed into the stack is the first value popped

(2)

out from the stock.

Q When a 32-bit value is pushed on the Stack. what happen to Esp?

Ans Espi's Decremenited by 4.

Q What would happen if the RET instruction was omitted from a procedure?

Ans:- Execution would continue beyond the End of the procedure into the beginning of onother procedure.

Q How are the word Reaive and returns used in the suggest procedure documentation?

Ans:- A list of input paramiter and their usage labeled by a word. Sdeh as Receives. A description of any values return ed by the procedure labeled by a word, Such

as returns

Q Which procedure in the link library generates a random integer within a selected range;

Ans:- Random Range procedure.

Q Which procedure in the Library display "press [Enter] to continue" and waits for the user to press the Enter key".

Ans:- Wait Msg procedure.

Q write statement that cause a program to pause for 700 milli seconds.

Ans:- code example mov eax, 700
call Delay.

Q Which procedure from the link library writes an unsigned integer to the console window in decimal

formal;

Ans :- Write Dec procedure.

Q Which procedure from the link library place the cursor at a sepecfic console window location?

Ans :- Gotoxy.

Q What are the required input parameter for the DumpMEM procedure?

Ans :- Imagine two possible way of calling the DumpMem procedure.

Require input parameter.

Pushed

```
mov esi, OFFSET array
mov ecx, LENGTHOF array
mov ebx, TYPE array
call DumpMem
```

popad.

OR

```
push   OFFSET   array
push   LENGTHOF array
Push   TYPE     array
call   DumpMEm
```

Q What are the required input parameter for the readstring Produre?

Ans :- EDx contain the offset of an array of bytes, and Ecx contain the maximum number of Character to read.

Q Which procedure in the link library generates a random integer within a selected range?

Ans :- Random Range.

Q Write a sequence of statement that use only PUSH instruction to exchange the value in the EAX and EBX register.

Ans :- A sequence of statement are push ebx, Assume EBX = X and EAX = Y, here the content of EBX (i.e X) is pushed push eax, which is assumed to be Y pop ebx, Y from stack is assigned to EBX, therefore EBX = Y

pop eax; X from stack is assigned to EAX therefore EAX = X.

Q Create a procedure that generate a random string of length L, containing all capital letter, when calling the procedure pass the value of L in EAX and

pass a pointer to an array of byte that will hold the random string. write a test program that calls your product procedure 20 times console window.

program :-

```
; Random strings

INCLUDE Irvine32.inc
TAB = 9                ; ASCII code for Tab
StrLen = 10            ; length of the string

.386
.model flat, sidecall
.stack 4096
ExitProcess PROTO, dwExitCode : DWORD

.data
Str1 BYTE "The 20 random
string are :", 0

.code
main PROC
```

```
mov     edx ,OFFSET  str1 ; "The 20
random  string  are".

call  WriteString   ; writes String
call  CrlF          ; writes on end-of-
                      line sequence to
                      the console
                      window.

mov   ecx ,20       ; create 20 strings.

L1 : mov  edx, offSET  arr1.

mov   eax ,strLen   ; EAX : String length
call  Randomstring  ; EAX : String length
call  Display       ; generate the
                      random String.

mov   al ,TAB
call  writechar
exit

main  ENDP          ; leave a tab space.

Randomstring  PROC  uses  eax esi
mov   ecx ,eax  ;   Ecx = String length
```

```
L1 :    mov     eax , 26
        call    RandomRange
        add     eax ,65  ; EAX gets ASCII of
                           a capital letter
        mov     arr [esi], eax

        inc     esi
        loop    L1

        Randomstring EXDP

        Display  PROC  uses  ean esi ; Display the
                                       genxate
                                       randomstoings

        mov     ecx, eax                ; Ecx = string length

L1 :    mov     eax, arr [esi] ; EAX = AsCIIvalue
        call    write char     ; write the
                                 letter
        inc     esi
        loop    LI
        Display ENDp

        call    dumpregs

        INVOIF  Exit  process , 0

END     main.
```

Q write a program that display a single character at 100 random screen location using a timming delay of 100 milli second Hint: use the GetMax x y procedure of determine the current size of the console window.

Ans:- Title Random characters (source cpp)
// This program display a single character at 100 random screen locations

include Irvine 32.inc

- data
raw WORD ? : // rows variable to hold num of rows.

cols WORD ? ; // colls variable to hold num of columns.

- Code
main PROC

```
call    clrscr          ;// Set cursor top left
mov     ecx,100         ;// loop for 100
                           times.
L1:     cal  GetMaxXY   ;// Size console
                           window
        mov  rows,ax    ;// return rows
        mov  cols,dx    ;// return colums

        movzx eax,rows  ;// moving rows eax
        call RandomRange;// generate integer
        mov  dh,al      ;// Setting range
                           boundaries

        movzx eax,cols  ;// moving colums
                           to eax
        call Random range;// generate integer
        inov dl,al      ;// setting range
                           boundaries.

        call Gotoxy     ;// cursor alocation

        call writechar  ;// write random
                           character
        mov  eax,100    ;// time 100
                           milliseconds.

        call Gotoxy     ;// cursor
                           a location
```

```
call    writechar       ; // write random
                          character
mov     eax, 100        ; // time 100 milli
                          seconds.
call    delay           ; // pauses program,
                          100 millisecond.

Loop L1                 ; lopping

exit
main ENDP

END main
```

Question No 23

Ans:-

```
; color Matrix
; write a program that displays
  a single characters in all
  possible.
; combination of fore ground and
  back ground color (16x16 = 256).
  The
; color are numbered from 0
  to 15. So you can use
```

```
a       nested      loop    to
; generate    all     possible
  combinations.

; Last . updata :

INCLUDE  Irvine  32. inc

CR  = ODh      ; carriage   return
LF  = OAh      ; line       feed .

. data
prompt  BYTE "please type a
character:" , 0

dispchr  DNORD;

. Code
main  PROC

; set  text  color  to  while
  text  on  black  background.

; Even  though  these  are  the
  default  colors.
```

```
; Each color constant is defined
  in Irvine32.inc

    mov eax, white + (black * 16)

    call SetTextColor
    call Clrscr          ; Clear the
                           Screen

; Get the user to type
  some character for our
  display:

    mov edx, OFFSET prompt1 ;
    call WriteString        please a type --
    call ReadChar


; one now has the desired
  character in AL,
; but we can only used it
  letter, so
; we save it in variable
  dispchar:

    mov dispchar, eax
    call Crlf       , new line
```

```
; Generating all the possible
  color combination i
; In a nested loop require
  a bit more work
; than simply a single loop.
; Exc will be preserved on
  the stack across the inner
  loop:

    mov ecx ; outer loop counter.

LL:- push ecx ; save outer loop
                counter.

; As the text attributes
  above illustrate
; to set colors we need
  to compute
; EAX in a way that
  combine both attributes
; set EAX using the stack
  copy of ECX
; value so the background
  will vary
; slowest (with the outer loop
  counter)

    mov ecx, 16 ; inner loop
      counter
```

```
L2 : pop eax          ; get the outer loop counter in EAX

   : push eax         ; saves it back again
     sub eax, 1       ; shift range from 1-16 to 0-15

     rol eax, 4       ; sets A1 for the background color
     rol eax, ecx     ; add the inner loop counter of ecx

     sub eax, 1       ; set A1 for the background color
     call setTextColor

; EAX is now avaiable for restoring the display character.

     mov eax, dispch
     call writeChar

     loop L2

; Return the cursor to begining of next row of matrix
```

```
call  crlF      ; new line
pop   ecx       ; restore outer loop
                  counter
loop  L1
```

; show the result on screen until user hit enter, then exit.

```
mov   eax, white + (black * 16)
call  set textcolor ; Otherwise we
                      leave black
                      on black

call  crlF          ; new line
call  WaitMsg       ; "press [enter]".....

exit
main  ENDP
END   main.
```

Question NO 8

Ans code example:

```
mov   eax, 700.
call  Delay
```

Question NO 14

Final value in EAX after these instruction

Ans:- Exection . (5)

Question NO 16

Ans:- By looking at this I think that EAX would still equal 30 at the end of line 6 Because eax was just pushed on the stack not a change in value.

Question NO 15

Ans This one I would belive it would equal 10 because of Lifo (lost in first out). EAX would equal 10 because it was the last one in.

Question NO 17

Ans:- C. EAx will be equal 30 on line 6.

Question NO 18

Ans:- A. EDx will be equal 40 on line 6.

Question NO 19

Ans:-

```
push    ebx
push    eax
pop     ebx
pop     eax
```