

Para Universidad Autónoma de Baja California
Facultad de Ciencias Químicas e Ingeniería



Alumno: Eduardo Marcelo Gutiérrez Soto
Profesora: Evangelina Lara Camacho
Practica 11 Paso de parámetros por línea de comandos

Teoría

Responda las siguientes preguntas.

1- Que es el Prefijo del Segmento del Programa (PSP).

Es una estructura de datos de 256 bytes que contiene información sobre el programa que tenemos en ejecución y se encuentra almacenada después de la pila del programa.

2- De que distintas maneras se puede obtener la dirección segmento de PSP?, cuando se usa una interrupción de DOS para obtenerla, en que registro se retorna la dirección del segmento?.

Se tienen dos maneras para poder obtener la dirección del segmento de PSP una es utilizando los registros, DS y ES ya que ellos apuntan a dicho registro inicialmente, de la primer manera utilizando esos registros lo que hacemos es forzar al registro ES para acceder al registro de PSP de la siguiente manera, podremos utilizar un índice ya sea SI, este recibiendo la cantidad de parámetros.

Con rutinas del DOS, tendríamos que utilizar lo que es la interrupción 21h y el servicio 62h para obtener los parámetros y BX es el que nos retorna la dirección de los parámetros y haríamos uso de ES para direccionar a ellos. Le podremos mandar a un índice el número de parámetros que sería desde la 80h.

3- En que dirección de desplazamiento (offset) de PSP se encuentra la cantidad de argumentos recibidos (en cantidad de bytes)?

Se pueden obtener en el desplazamiento de 80h se tiene guardados la cantidad de los parámetros recibidos y son 128 bytes.

4- En que dirección de desplazamiento (offset) de PSP se encuentran los argumentos recibidos?

Desde la 82 ya que en la 81 hay un espacio en blanco que no contiene nada

1. Programe el procedimiento `numerosTriangulares` el cual despliega en pantalla los `n` primeros números triangulares.

Yo hice uso de mi programa de `printNumBase` para hacer la impresion de la pantalla de los numeros triangulares ya que se me hacia mas practico utilizar ese a que usara un `itoa` ya que es mas largo ese codigo.

CÓDIGO:

Asi realizo lo que son mis números triangulares recibo en `AX` el numero y ese numero lo utilizo con `CX` para hacer el numero de iteraciones necesarias para poder calcular los `N` números triangulares.

```
xor cx,cx
xor bx,bx
xor dx,dx
mov cx,ax
mov bx,0
mov bh,2

cmp cx,255
mov bl,1
mov al,1
jg fin      ;si es mayor a 255 es que no es un numero valido
numero:
push ax
add bl,1    ;hacemos el (n+1)
mul bl      ;multiplicamos el valor de AL(BL+1)
div bh      ;AL/BL
mov ah,0
push bx
mov bx,10
call printNumBase
mov ah,02
mov dl,', '
int 21h
pop bx
pop ax
inc ax
loop numero

fin:
pop dx
pop cx
pop bx
ret
endp
```

CAPTURA DEL PARÁMETRO POR CONSOLA:

Mi código para obtener el parámetro el el siguiente.

```
parametros proc
    push ax                ;salvo registros a utilizar
    push di                ;
    push si                ;
    push bx                ;

    mov ah,62h             ;ejecutamos el servicio 62h para la obtencion de parametros
    int 21h               ;
    mov di,82h             ;movemos al registro di el valor 82 donde inician los parametros
    mov ah,0dh             ;movemos un enter a ah
@@while: cmp ES:[DI],ah    ;comparamos ES:[DI] con el enter
    je @@fin              ;si es igual entonces termina el ciclo
    mov al,ES:[DI]         ;movemos la letra hacia el registro al
    mov [si],al            ;enviamos a la cadena en si la letra de al
    inc si                 ;incrementamos si para apuntar a la siguiente direccion
    inc di                 ;incrementamos di para apuntar a la siguiente direccion
    jmp @@while

@@fin: mov byte ptr[si],0  ;movemos un 0 al final ya que asi terminan las cadenas
    pop bx                 ;volvemos a poner los registros como estaban
    pop si                 ;
    pop di                 ;
    pop ax                 ;
    endp
```

EJECUCIÓN:

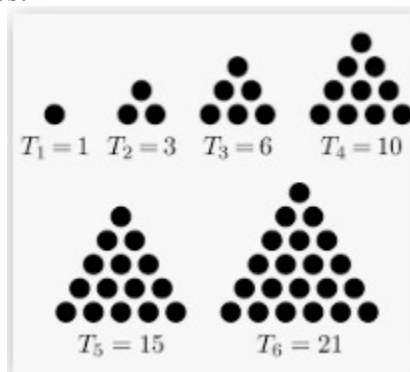
```
C:\NB>P11-A.EXE 6
```

```
Programa que imprime los N numeros triangulares ingresador por el usuario
```

```
1,3,6,10,15,21,
```

```
C:\NB>_
```

Ejemplo de los numeros triangulares:



Basándose en el código que desarrolló en la Práctica 8 sobre escritura de archivos, realice las modificaciones necesarias de forma que el usuario indique por medio de la línea de comando el archivo y el texto que quiere escribir. Ejemplo: > P11c.exe practica.txt Hola mundo

en este ejercicio lo que realice fue modificar mi código para capturar un parámetro por medio de la consola a que pudiera capturar varios parámetros, lo que hacia es que donde tenia un espacio metía un valor null haciendo referencia a final de una cadena y así logre separar lo que fueron mis parámetros, lo que son el nombre del archivo, y el texto a escribir.

CÓDIGO DE PARÁMETROS:

```
;PROCEDIMIENTO PARA CAPTURA DE CARACTERES DESDE LA CONSOLA
parametros proc
    push ax                ;salvo registros a utilizar
    push di                ;
    push si                ;
    push bx                ;

    mov ah,62h             ;ejecutamos el servicio 62h para la obtencion de parametros
    int 21h                ;
    mov di,82h             ;movemos al registro di el valor 82 donde inician los parametros
    mov ah,0dh             ;movemos un enter a ah
    mov bl,20h             ;movemos un espacio al bl
while: cmp ES:[DI],ah       ;comparamos ES:[DI] con el enter
    je @@fin               ;si es igual entonces termina el ciclo
    mov al,ES:[DI]         ;movemos la letra hacia el registro al
    mov [si],al            ;enviamos a la cadena en si la letra de al
    cmp byte ptr [si],bl
    je @@par
    continuar:             ;continuamos con el ciclo
    inc si                 ;incrementamos si para apuntar a la siguiente direccion
    inc di                 ;incrementamos di para apuntar a la siguiente direccion
    jmp @@while

par:
    mov byte ptr [si],0
    jmp continuar         ;nos sirve para separar los parametros

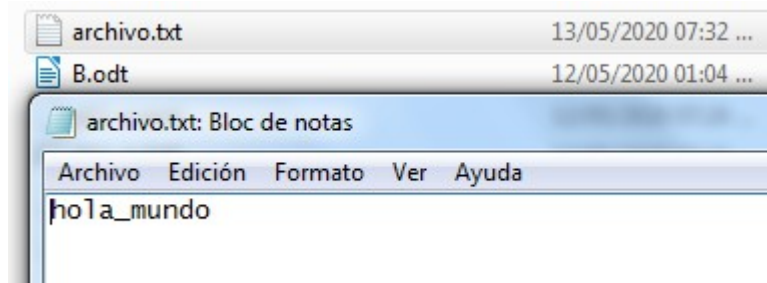
fin: mov byte ptr[si],0    ;movemos un 0 al final ya que asi terminan las cadenas
    pop bx                ;volvemos a poner los registros como estaban
    pop si                ;
    pop di                ;
    pop ax                ;
    endp
; *****
```

Se modifiqué muy poco para lo de la captura de parámetros lo que hacía fue comparar donde tenía un espacio cambiarlo por un carácter null.

EJECUCIÓN:

```
C:\B>p11-c.exe archivo.txt hola_mundo  
Archivo escrito  
C:\B>
```

Mandamos los parámetros correspondientes lo que es el nombre del archivo y el texto que le vamos a ingresar y en nuestro archivo de texto tendremos el siguiente texto.



y de esa forma logramos grabar un texto en un archivo mandando el nombre del archivo desde la línea de comandos y el texto que queremos ingresar.

CÓDIGO DE ARCHIVOS:

```
mov si,offset AECHIVO_2  
call parametros ;MANDAMOS LA CADENA DONDE VAMOS ALMACENAR LOS PARAMETROS  
  
mov si,0 ;INICIALIZAMOS EL INDICE EN 0  
com:cmp byte ptr [AECHIVO_2+si+12],0  
je @nxt  
inc si ;CONTEO DE BYTES A ESCRIBIR  
jmp com ;CALCULAMOS EL NUMERO DE BYTES QUE SE VAN A ESCRIBIR  
@nxt: ;obtenemos la cantidad de bytes a escribir  
;*****FUNCIONES DE ARCHIVOS*****  
MOV AL,1h ;ACCESO SOLO A ESCRITURA  
MOV DX,OFFSET AECHIVO_2 ;ABRIMOS LA CANEDA QUE INGRESAMOS  
CALL ABRIR_ARCHIVO ;ABRIMOS EL ARCHIVO  
MOV BX,AX ;MOVEMOS EL HANDLE  
MOV CX,SI ;CANTIDAD DE CARACTERES A ESCRIBIR  
MOV DX,OFFSET [AECHIVO_2+12];ACCEDEMOS A LA CADENA QUE GRABAREMOS EN EL ARCHIVO  
CALL ESCRIBIR_ARCHIVO ;ESCRIBIMOS EL ARCHIVO  
CALL CERRAR_ARCHIVO ;CERRAMOS EL ARCHIVO  
;*****ARCHIVO ESCRITO Y CERRADO*****  
  
MOV DX,OFFSET MENS ;IMPRIMIMOS UN MENSAJE DE QUE SE GRABO EL TEXTO  
CALL PUTS |
```

Conclusiones y dificultades.

La practica se me hizo muy interesante el como podemos acceder a los valores desde la consola y de como nos las tenemos que ingeniar para poder hacer uso de ellos por que al momento de involucrar mas parámetros se hace un poco mas difícil la cosa al momento de acceder a ellos, en el ejercicio que tuve complicaciones fue en el del pangrama, no se que pase por alto pero no me salio el programa pero se que la idea que tengo para solucionarlo es mas eficaz que estar accediendo a un arreglo de letras y hacer todo desde ahí, lo que hacia era comparar la letra actual con una letra almacenada en al y si se repetía no tomar e las letras repetidas, pero al igual no supe que hice mal que al ejecutarlo me se me bloqueaba el dos box, de todos modos dejare el código en los archivos de entrega.

Bibliografías

desconocido. (desconocido). Interrupciones 21h. 13/05/20, de desconocida Sitio web:
http://ict.udlap.mx/people/oleg/docencia/ASSEMBLER/asm_interrup_21.html