

## Argumentos enviados por línea de comando

El **Prefijo de segmento de programa** (Program Segment Prefix, PSP) es una estructura de datos de **256 bytes** que contiene información sobre el programa en ejecución. Se encuentra almacenado después de la Pila del programa.

La dirección de **segmento** de PSP se puede obtener de dos maneras, como se explica posteriormente en la sección *Obtener los argumentos de línea de comando*. En el desplazamiento (**offset**) **80h** y **81h-FFh** de ese segmento se encuentran los **argumentos de línea de comando** del programa en ejecución.

- **Desplazamiento 80h:** Contiene la **cantidad** de argumentos. El conteo inicia a partir del **primer espacio** posterior al nombre del programa y **no incluye el símbolo CR** (la tecla Enter presionada por el usuario).
- **Desplazamientos 81h a FFh:** Contiene los **argumentos**. Empieza por el **primer espacio** y termina con **CR**. Son **máximo 127 bytes** de argumentos.

Ejemplo:

Si se ejecuta en la línea de comando el programa `compu.exe` y se le envían los argumentos `-s 56`:

```
compu.exe -s 56
```

En el desplazamiento **80h** de PSP está almacenado un **6**, porque es la cantidad de caracteres ASCII posteriores al nombre del programa (incluye el espacio como se mencionó anteriormente).

A partir de la dirección **81h** están almacenados los caracteres ASCII `"-s 56"` más un **CR** (ASCII 13) al final.

## Obtener los argumentos de línea de comando

Existen dos maneras de obtener la dirección de segmento de PSP:

1. Los registros **DS** y **ES** inicialmente apuntan al PSP.
2. La **int 21h** servicio **62h** retorna en **BX** la dirección de segmento.

*Ejemplo 1:*

; Usando a ES para direccionar a PSP.

```
mov si,80h          ; en la dirección ES:80h está la cantidad de argumentos.
mov al,es:[si]
call printDec       ; despliega en pantalla la cantidad de argumentos.
```

*Ejemplo 2:*

; Usando la int 21h servicio 62h para obtener la dirección de segmento de PSP.

```
mov ah,62h
int 21h
mov es,bx           ; Usando a ES para direccionar a PSP
mov si,80h          ; en la dirección ES:80h está la cantidad de argumentos.
mov al,es:[si]
call printDec       ; despliega en pantalla la cantidad de argumentos.
```

Los ejemplos anteriores muestran como desplegar la **cantidad** de argumentos recibidos. Para obtener los caracteres ASCII de los argumentos, se puede realizar:

```
xor ah,ah
mov cx,ax          ; en el código anterior se copió a AL la cantidad de argumentos.
mov si,81h         ; en la dirección ES:81h inician los argumentos.
@@nxt:
mov al,es:[si]
call putchar       ; despliega en pantalla los argumentos carácter por carácter.
inc si
loop @@nxt
```

## Ejercicio

Programa el procedimiento **esPangrama** el cual recibe una cadena de caracteres como argumento en la línea de comando y despliega en pantalla si la cadena es un pangrama en el idioma inglés. Una cadena es un pangrama si contiene todas las letras del alfabeto.

Ejemplo:

Considerando que su archivo se llama **Ej10**, en la línea de comando de DOSBox se invoca de la siguiente manera:

```
> Ej10.exe The quick brown fox jumps over the lazy dog
```

Donde *The quick brown fox jumps over the lazy dog*, es la cadena a verificar.

El programa debe desplegar:

Si es un pangrama.