

Práctica No. 11

Interfaz entre lenguaje C89 y el lenguaje ensamblador para el procesador 8086

Objetivo:

Diseñar e implementar un programa en lenguaje C que haga uso de rutinas en lenguaje ensamblador que fueron ensambladas en un archivo independiente al programa principal.

Materiales:

Turbo C++, TASM.exe

Teoría:

- Interfaz entre el lenguaje C y el lenguaje ensamblador

Desarrollo:**PARTE 1.**

A partir de los dos archivos en el Anexo A, realizar los siguientes pasos:

1. Utilice los programas TASM.exe para ensamblar **myputc.asm**, esto generará el archivo **myputc.obj**, utilizar la opción /ml.
2. Crear un proyecto en Turbo-C (**Prac12.prj**) que contenga los archivos **Prac12.c** y **myputc.obj**; compile y ejecute el proyecto.
3. Al ejecutar el proyecto este ejecuta el programa **Prac12.c**, el cual presenta la cadena “Hola Mundo” en pantalla. Para esto, el programa hace uso del apuntador a carácter `str` el cual apunta al inicio de la cadena; el programa toma carácter por carácter para desplegarlo uno por uno haciendo uso de la función externa `myputchar`. Esta función es calificada como externa pues no está definida en **Prac12.c**, pues `myputchar` se encuentra ensamblada en el archivo **myputc.obj** que forma parte del proyecto. La función está diseñada para ser invocada desde un programa en lenguaje C de tal forma que `myputchar` respeta el protocolo y reglas de paso de parámetros. Así esta función recibe el carácter como parámetro y lo presenta en pantalla haciendo uso de la interrupción 21H servicio 2 del sistema.

Actividad para validar el desarrollo de esta parte:

1. El programa deberá ejecutarse y mostrar el mensaje: Hola Mundo

PARTE 2.

Implementar un programa que multiplique dos matrices de 3x3. Las matrices *A* y *B* deberán ser declaradas en lenguaje C, posteriormente enviadas a ensamblador a multiplicarse, y finalmente la matriz resultante *C* deberá regresar al lenguaje C para su debida impresión.

Actividad para validar el desarrollo de esta parte:

1. El programa deberá ejecutarse y mostrar el resultado tal como la descripción lo pide.

Anexo A*myputc.asm*

```
dosseg
.model small
.code
public _myputchar
_myputchar PROC
    push bp
    mov bp,sp
    mov dl,[bp+4]
    mov ah,2
    int 21h
    pop bp
    ret
_myputchar ENDP
END
```

Prac11.c

```
#include<stdio.h>
extern void myputchar( char x );
char *str = {"Hola Mundo\n"};
void main (void)
{
    while(*str)
        myputchar(*str++);
    getchar();
}
```