

Práctica 3b: Diseño de programas utilizando C y ensamblador del 80x86

En esta práctica se realizará un programa principal escrito en C (pract3b.c) que hará llamadas a diferentes funciones, que deben ser desarrolladas en ensamblador en un fichero (módulo) diferente. El alumno deberá realizar el correspondiente análisis de los requisitos expuestos y realizar su implementación. Cada una de las funciones desarrolladas tiene el mismo peso en la nota final de la práctica.

El programa pract3b.c lo deberá preparar el alumno y deberá incluir un menú de usuario similar al de la práctica anterior (p3a).

Módulo: pract3bs.asm

En el módulo pract3bs.asm se deben desarrollar en ensamblador un conjunto de funciones relacionadas con cadenas, que serán llamadas desde el programa principal en C de acuerdo a los siguientes prototipos:

1.- int encuentraSubcadena (char* str, char* substr)

Devuelve el índice a partir del cual se encuentra contenida la subcadena "substr" en la cadena "str". Si la subcadena no se encuentra dentro de la cadena, devuelve el valor -1. Por ejemplo, si str="abcdefghijklmn" y substr="defghijk", la función debe devolver el valor 3.

2.- unsigned int calculaSegundoDC (char* numCuenta)

Devuelve como valor de retorno el segundo dígito de control de una cuenta bancaria correspondiente al número de cuenta de 10 dígitos dado como cadena de caracteres ASCII (por ejemplo: "0438853602"). Para realizar el cálculo, debemos multiplicar cada dígito de la cuenta por 1, 2, 4, 8, 5, 10, 9, 7, 3, 6 respectivamente, empezando por la izquierda. Sumando las 10 cantidades obtenidas, y efectuando el módulo 11 de la suma, el resultado a devolver por la función será igual a 11 menos el módulo 11 de la suma. Con una excepción, si el resultado final obtenido es 10, debe retornar el valor 1. Por ejemplo, para numCuenta="0438853602", la función debe retornar el valor 9 (como número entero de 16 bits sin signo).

Notas:

- El compilador precede todas las referencias externas con el carácter "_", por lo que es necesario que todas las funciones desarrolladas en ensamblador comiencen con el mismo (ejemplo: `_calculaSegundoDC`)
- Todas las funciones desarrolladas en ensamblador deben ser declaradas como PUBLIC para poder ser llamadas desde el programa principal

- No se permite la declaración de variables globales en ensamblador. De hecho, cada módulo desarrollado en ensamblador contendrá **exclusivamente** un solo segmento de código, que debe comenzar con las siguientes líneas:

```
<nombre módulo> SEGMENT BYTE PUBLIC 'CODE'
ASSUME CS: <nombre módulo>
```
- El programa a desarrollar en C (pract3b.c) debe estar compilado en modelo LARGE. Las funciones desarrolladas en ensamblador serán FAR.
- El programa pract3b.c pedirá al usuario diferentes valores necesarios para probar cada una de las funciones desarrolladas en ensamblador en los diferentes módulos, mostrando el resultado obtenido para cada operación en el formato adecuado de manera que facilite su comprensión.
- Todas las cadenas de caracteres en C acaban con un carácter NULL (último byte a cero).

ENTREGA DE LA PRÁCTICA: Fecha y contenido

Se deberá subir a Moodle un fichero zip que contenga los ficheros fuentes de los programas (**pract3b.c** , **pract3bs.asm**) y el fichero **makefile**.

Los ficheros a entregar deberán contener en la cabecera el nombre del autor. Así mismo, el código de los ficheros entregados deberá estar correctamente tabulado y comentado. La falta de comentarios o la baja calidad de éstos, será calificada negativamente.

El límite de fecha de subida de los ficheros, para cada grupo es el siguiente:

Grupos del Jueves: 28 de Abril a las 23:55h

Grupos del Viernes: 29 de Abril a las 23:55h

Anexo: Compilación de un proyecto desarrollado en C y ensamblador
--

El programa pract3b.c lo deberá preparar el alumno y deberá incluir un menú de usuario similar al de la práctica anterior (p3a).

Para realizar la compilación del programa en C se utilizará el compilador del TurboC (tcc). Para conocer todas las opciones de compilación que ofrece podemos ejecutar tcc sin parámetros dentro del DosBox. Los módulos en ensamblador se compilarán como en las prácticas anteriores, utilizando el tasm.

El programa en C debe ser compilado con la opción *-ml* (memory model large), mientras que los módulos en ensamblador deben ser ensamblados con la opción */ml* (case sensitivity on all symbols).

Para crear el ejecutable final “pract3b.exe” debéis generar un makefile similar al de la práctica anterior.