

## Laboratorio de Sistemas Basados en Microprocesadores

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

En esta práctica se utilizará un programa principal escrito en C (pract3.c) que hará llamadas a diferentes funciones, que deben ser desarrolladas en ensamblador en un fichero aparte (pract3.asm). El alumno deberá realizar el correspondiente análisis de los requisitos expuestos y realizar su implementación.

#### Módulo 1: pract3a.asm

En el módulo pract3a.asm se deben desarrollar en ensamblador un conjunto de funciones relacionadas con cálculos matemáticos, que serán llamadas desde el programa principal en C de acuerdo a los siguientes prototipos:

1.-int calculaMediana(int a, int b, int c, int d) (3 ptos)

Devuelve el valor de la mediana de a, b, c y d (números enteros de 16 bits con signo).

2.-void enteroACadenaHexa (int num, char\* outStr) (4 ptos)

Convierte el número “num” en una cadena de caracteres outStr que representa el valor de num en hexadecimal mediante caracteres ASCII (acabada en cero). Por ejemplo, si num = 1020 (3FCh), la cadena outStr contendrá los siguientes 4 bytes: 33h (código ASCII del 3), 46h (código ASCII de F), 43h (código ASCII de C), 00h (terminación).

3.- void calculaLetraDNI(char\* inStr, char\* letra) (3 ptos)

Calcula la letra correspondiente al DNI representado como cadena de bytes inStr (8 caracteres ASCII sin letra, por ejemplo: 12345678), y devuelve en el byte letra el código ASCII correspondiente (para el DNI 12345678 sería el byte 5A, correspondiente al carácter “Z”).

Para verificar el NIF, el algoritmo de cálculo del dígito de control se divide el número entre 23 y el resto se sustituye por una letra que se determina por inspección mediante la siguiente tabla:

|       |   |   |   |   |   |   |   |   |   |   |    |    |
|-------|---|---|---|---|---|---|---|---|---|---|----|----|
| RESTO | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| LETRA | T | R | W | A | G | M | Y | F | P | D | X  | B  |

|       |    |    |    |    |    |    |    |    |    |    |    |
|-------|----|----|----|----|----|----|----|----|----|----|----|
| RESTO | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| LETRA | N  | J  | Z  | S  | Q  | V  | H  | L  | C  | K  | E  |

Por ejemplo, si el número del DNI es 12345678, dividido entre 23 da de resto 14, luego la letra sería la Z: 12345678Z.

---

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: `_calculaLetraDNI`)
- Todas las funciones desarrolladas en ensamblador deben ser declaradas como `PUBLIC` para poder ser llamadas desde el programa principal
- Cada módulo desarrollado en ensamblador contendrá exclusivamente un segmento de código, y debe comenzar con las siguientes líneas:  

```
<nombre módulo> SEGMENT BYTE PUBLIC 'CODE'
ASSUME CS: <nombre módulo>
```
- El programa a desarrollar en C (`pract3.c`) debe estar compilado en modelo `LARGE`. Las funciones desarrolladas en ensamblador serán `FAR`.
- El programa `pract3.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.
- Las cadenas de caracteres en C acaban con un byte de valor cero.

#### **ENTREGA DE LA PRÁCTICA: Fecha y contenido**

Se deberá subir a Moodle un fichero zip que contenga los ficheros fuentes de los programas y el fichero `makefile`. La entrega es por parejas.

Los ficheros a entregar deberán contener en la cabecera los nombres de los autores y el identificador de la pareja. 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 Miércoles: 14 de Abril de 2020 a las 23:55h

Grupos del Jueves: 15 de Abril de 2020 a las 23:55h

Grupos del Viernes: 16 de Abril de 2020 a las 23:55h

### Anexo: Compilación de un proyecto escrito en C y ensamblador

En esta práctica disponemos de 2 ficheros, el programa principal escrito en C (pract3.c) y el módulo que contiene las funciones escritos en ensamblador (pract3a.asm).

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.

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 generar el ejecutable final “pract3.exe” podemos utilizar el siguiente makefile (respetando los tabuladores):

```
all: pract3.exe

pract3.exe: pract3.obj pract3a.obj

    tcc -v -ml -Lc:\compila\tc\lib pract3.obj pract3a.obj

pract3.obj: pract3.c

    tcc -c -v -ml -Ic:\compila\tc\include pract3.c

pract3a.obj: pract3a.asm

    tasm /zi /ml pract3a,,pract3a
```