



GRADO EN INGENIERÍA SOFTWARE

Arquitectura de Computadores

2024-2025

Este es el enunciado de la Práctica de Logisim I relacionada con ensamblador Z80 con el emulador Zeus

Evaluación y condiciones: Se recuerda a los alumnos que esta práctica se realiza por parejas que deben comunicadas a los respectivos profesores. La nota de esta práctica cuenta la mitad del 30% de la nota final que está asignado a prácticas/proyectos. Es decir, cuenta un 15% de la nota final.

Es necesario obtener mínimo un 4 mínimo por separado en cada uno de los proyectos y para aprobar la asignatura y 4,5 de media entre las dos prácticas evaluables.

Advertencia: La detección de uso de IA conllevará el suspenso de la práctica y la solicitud de apertura de expediente disciplinario para la pérdida de la convocatoria ordinaria. Si usas construcciones que no se han visto en clase deberás pasar un examen oral para explicar por qué no has sido capaz de realizar programas sencillos sin recurrir a programación avanzada del Z80 que no entra en el temario.

PROYECTO I Zeus

ENUNCIADO:

En esta práctica debes resolver cuatro retos sencillos de programación para ejercitarte para el examen.

Programa 1. Cifrado Romano (3 puntos)

Usaremos una técnica de cifrado muy simple que consiste en desplazar cada carácter un número de caracteres fijo. Así, si el carácter es 'A' y la clave es 3, se cifra como 'D', o si es 'F', se cifra como 'I'. Para simplificar el ejercicio, si la suma es mayor que 'Z' el carácter se cifra con el carácter ASCII cuyo código es la el del original más la clave. Para descifrar, el método es el contrario, se resta la clave de cada carácter del contenido del mensaje. SOLO USAMOS MAYÚSCULAS.

El programa cifrará un texto llamado "TextoOriginal", guardará el resultado en "TextoCifrado" y, a continuación descifrá "TextoCifrado" para guardarlo en "TextoDescifrado".

La clave se calculará al principio usando la media aritmética de los meses de nacimiento de los alumnos que forman el grupo de trabajo. Sumando ambos meses, dividiéndolos por dos y quedandonos la parte entera.

Definir las variables siguientes:

- MesA y MesB para guardar los meses de nacimiento del alumno A y del B.
- RomanKey, donde guardarás el valor de la clave
- TextoOriginal, cuyo valor inicial es "THAIS HABET NIGROS NIVEOS LAECANIA DENTES"
- TextoCifrado y TextoDescifrado, inicializados a blancos.

En el video mostrareis el resultado del programa sobre ambas cadenas en la pestaña ZeusCode

```
008040 C3 3F 80 07 0A 08 00 54 48 41 49 53 20 48 41 42 .?.....THAIS HAB
008050 45 54 20 4E 49 47 52 4F 53 20 4E 49 56 45 4F 53 ET NIGROS NIVEOS
008060 20 4C 41 45 43 41 4E 49 41 20 44 45 4E 54 45 53 LAECANIA DENTES
008070 5C 50 49 51 5B 28 50 49 4A 4D 5C 28 56 51 4F 5A \PIQ[ (PIJM\ (VQOZ
008080 57 5B 28 56 51 5E 4D 57 5B 28 54 49 4D 4B 49 56 W[ (VQ^MW[ (TIMKIV
008090 51 49 28 4C 4D 56 5C 4D 5B 54 48 41 49 53 20 48 QI (LMV\M[THAIS H
0080A0 41 42 45 54 20 4E 49 47 52 4F 53 20 4E 49 56 45 ABET NIGROS NIVE
0080B0 4F 53 20 4C 41 45 43 41 4E 49 41 20 44 45 4E 54 OS LAECANIA DENT
0080C0 45 53 4C 40 39 41 4B 18 40 39 3A 3D 4C 18 46 41 ESL@9AK.@9:=L.FA
```

Programa 2. Cifrado Romano usando el stack (3 puntos)

Repetir el ejercicio 1, pero usando una única rutina CifradoRomano, a la que se pasarán por el stack y por este orden, la referencia de la cadena que se quiere cifrar/descifrar, la referencia de la cadena destino de la operación, la longitud de ambas cadenas, la clave RomanKey, y por último, en un byte la operación que se desea realizar, que será el carácter 'C' para Cifrado y 'D' para descifrado.

La rutina opera sobre la cadena de destino por referencia y devuelve como valor el resultado de la operación que será 1 si se ha completado con éxito o 0 si la operación solicitada no existe.

En el video debéis mostrar capturas del stack con los parámetros pasados antes de la llamada. Modificad el texto original para que sea: "BEATI HISPANI QUIBUS BIBERE VIVERE EST"

También deberás mostrar el resultado final en la memoria.

```
si quiero cifrar, sumo la clave roman_key a cada caracter de texto_original. tipo_operacion defb 'C'
si quiero descifrar, resto la clave roman_key a cada caracter de texto_original. tipo_operacion defb 'D'
```

Programa 3. Sumar matrices (2 puntos)

El z80no puede definir matrices, pero para determinadas operaciones una matriz se puede manejar como un vector. Por ejemplo, una matriz 3x3 puede convertirse en un vector fila de 9 entradas si se escribe la primera fila, a continuación la segunda y luego la tercera, y sumar esa matriz a otra, elemento a elemento, definida también como vector.

Define MATRIZ_A y MATRIZ_B como vectores de 9 elementos y la variable Dimension con valor 9. Construye un bucle que realice la operación MATRIZ_A + MATRIZ_B y deje el resultado en la

variable MATRIZ_RESULTADO. Cada vez que se ejecute el bucle se invocará la rutina SumaElementos que toma el elemento correspondiente de la MATRIZ_A, lo suma al de MATRIZ_B y deja el resultado en MATRIZ_Z.

En el video deberás mostrar los valores finales de estas variables.

Programa 4. Multiplicar una matriz por un escalar (2 puntos)

La multiplicación de una escalar por una matriz implica multiplicar el número por cada elemento individual de la matriz. En Z80 no hay operación nativa de multiplicación, por tanto, recurrimos a usar la definición de esta operación, que es sumar tantas veces una cantidad como indica el multiplicador.

Modifica el programa 3 para que la MATRIZ_A se sume tantas veces como indica la variable Escalar y guarda el resultado en MATRIZ_RESULTADO. Tienes que usar la misma rutina SumaElementos del programa anterior.

En el video deberás mostrar los valores finales de estas variables.

ENTREGABLES Y EVALUACIÓN

La entrega de esta práctica se realizará en Blackboard exclusivamente por un solo miembro integrante del grupo y dentro de la fecha tope señala por el profesor. La entrega fuera de plazo conllevará una penalización en la nota.

Elementos a entregar:

- Un ZIP que contendrá los cuatro ficheros .ASM, llamados PROG1.ASM, PROG2.ASM, PROG3.ASM y PROG4.ASM
- Un pdf que debe contener:
 - o nombre y apellidos de los integrantes del grupo con una descripción de las soluciones implementadas, incluyendo pantallazos de la pestaña Zeus Code con el contenido del resultado de las ejecuciones.
 - o Link a un video corto (no más de 5 minutos subido a onedrive Y CONCEDER PERMISO DE ACCESO PARA EL PROFESOR, youtube o similar. NO SUBIR A BLACKBOARD) en el cual tiene que aparecer:
 - La voz de ambos alumnos integrantes del equipo
 - La pantalla de un ordenador donde se vaya viendo el código y la ejecución de los programas.
 - o Finalizar con conclusiones finales explicando lo que os ha llamado más la atención, lo que os ha resultado más difícil etc.