



Tarea 3

Indicaciones:

- Debe ser resuelta en las parejas de trabajo definidos.
- La fecha máxima de entrega es el día 29 de Enero del 2021, como se describe en el cronograma del curso.
- Debe ser entregada a través del sistema Mediación Virtual. Remita el código fuente únicamente (Archivo .asm) de cada uno de sus programas así como el PDF con los diseños, explicaciones, notas de cálculo, etc. El formato del archivo debe ser **SUNOMBRE_T3.asm**, puede remitir todos estos archivos de manera comprimida en un solo archivo .zip y el documento pdf debe nombrarse **SUNOMBRE_T3.pdf**.

Se tiene una tabla de datos de **1 byte sin signo**, cuya longitud es definida por la constante **LONG**, ubicados a partir de la posición **DATOS**. Además se tiene otra tabla llamada **CUAD** que contiene los números que tienen **raíz cuadrada entera** en el intervalo de los posibles valores de **DATOS** (4,9,16,25,36...). Se desea crear un arreglo llamado **ENTERO** que contendrá la **raíz cuadrada de CANT** números de **DATOS** que estén en **CUAD**. De esta manera, si **CANT=5**, el programa deberá buscar los primeros **5 números en DATOS que tengan raíz cuadrada entera**, es decir, que estén contenidos en **CUAD**. Además debe calcular esa raíz cuadrada y colocar los valores en **ENTERO**. La tabla **DATOS** puede tener valores repetidos.

Se debe diseñar e implementar un programa para realizar estas acciones en la tarjeta **DRAGON 12+**. Este programa debe incluir cuatro subrutinas:

1. **Subrutina LEER_CANT**: Esta subrutina es invocada desde el programa principal y debe recibir el valor de **CANT** por medio del teclado, utilizando la subrutina **GETCHAR**. La subrutina debe validar que el valor ingresado sea un número entre 1 y 99. La subrutina **LEER_CANT** debe desplegar el siguiente mensaje en pantalla:
> **INGRESE EL VALOR DE CANT (ENTRE 1 Y 99):**

Luego de que se presione una tecla numérica la subrutina debe validar que la tecla **presionada es numérica**, en caso contrario debe permanecer leyendo el teclado hasta que una tecla numérica sea presionada. Luego de leer el primer dígito, la subrutina **LEER_CANT**, desplegará ese valor en la pantalla y procederá a leer la **segunda tecla numérica**. Al recibirse la segunda tecla numérica, la subrutina debe calcular el valor ingresado como un número entero y devolver el valor en la variable **CANT**. El valor ingresado por el usuario debe aparecer al final del mensaje en el Terminal. Los valores entre 1 y 9 deberán ser recibidos como 01, 02..., 09. El valor **00 NO es un valor válido y no debe ser aceptado**.



Universidad de Costa Rica
Escuela de Ingeniería Eléctrica
Microprocesadores
IE-0623

EIE

Escuela de
Ingeniería Eléctrica

2. **Subrutina BUSCAR:** Esta subrutina es invocada desde el programa principal y busca los **CANT** valores de **DATOS** que estén en **CUAD**. Cada vez que encuentre un valor válido la subrutina **BUSCAR** debe llamar a la subrutina **RAIZ**. Además la subrutina debe llevar un contador de los valores encontrados en la variable **CONT**. La subrutina **BUSCAR** le pasará el valor a **RAIZ** por medio de la pila y por este mismo medio **RAIZ** devolverá el resultado. La subrutina **BUSCAR** debe poner este resultado en el arreglo **ENTERO**.

3. **Subrutina RAIZ:** Esta subrutina calcula la raíz cuadrada. El valor al cual se le debe obtener la raíz cuadrada debe ser pasado a la subrutina por la pila y la subrutina devolverá el valor calculado por la pila. **Nota: Para calcular el valor de la raíz cuadrada se utilizará el Algoritmo Babilónico.**

4. **Subrutina Print_RESULT:** Esta subrutina es llamada desde el programa principal y debe imprimir los resultados de la siguiente manera:

> CANTIDAD DE VALORES A ENCONTRADOS: <CONT>

Este resultado debe aparecer dos líneas por debajo del último mensaje desplegado. Adicionalmente esta subrutina deberá imprimir, en el terminal, el arreglo **ENTERO** calculado, de la siguiente manera.

> ENTERO: <num_1> , <num_2> , ..., <num_k>

Estos resultados deberán aparecer dos líneas por debajo del último mensaje desplegado.

Nota: Para desplegar todos los mensajes se debe utilizar la subrutina **Printf**.

De esta manera una corrida del programa, para:

DATOS = 4, 9, 18, 4, 27, 63, 12, 32, 36, 15.

Deberá generar en el Terminal lo siguiente:

> **INGRESE EL VALOR DE CANT (ENTRE 1 Y 99): 07**

> **CANTIDAD DE NUMEROS ENCONTRADOS : 4**

> **ENTERO: 2, 3, 2, 6**

- Realice el diseño del programa que resuelve este problema.
- Haga el código en ensamblador del S12 para este programa. Edítelo directamente en el **asmIDE**. Utilice las directivas de ensamblador apropiadas para definir las localizaciones de memoria. Coloque **LONG** en **\$1000**, **CANT** en **\$1001**, **CONT** en **\$1002**, **DATOS** en **\$1020** y **CUAD** en **\$1030** y **ENTERO**



Universidad de Costa Rica
Escuela de Ingeniería Eléctrica
Microprocesadores
IE-0623

EIE

Escuela de
Ingeniería Eléctrica

-
- en \$1010. Utilice los valores de **DATOS** y de **LONG** que le permita probar su código con el direccionamiento de las estructuras de datos indicado.
- c. El programa debe colocarse en la memoria RAM a partir de la posición **\$2000**. El programa debe estar debidamente estructurado y documentado.
 - d. Ensamble y depure el programa, hasta que genere el archivo **.S19**.
 - e. Corra el programa en la tarjeta **DRAGON 12** con un conjunto de datos de su elección y verifique que corre correctamente.