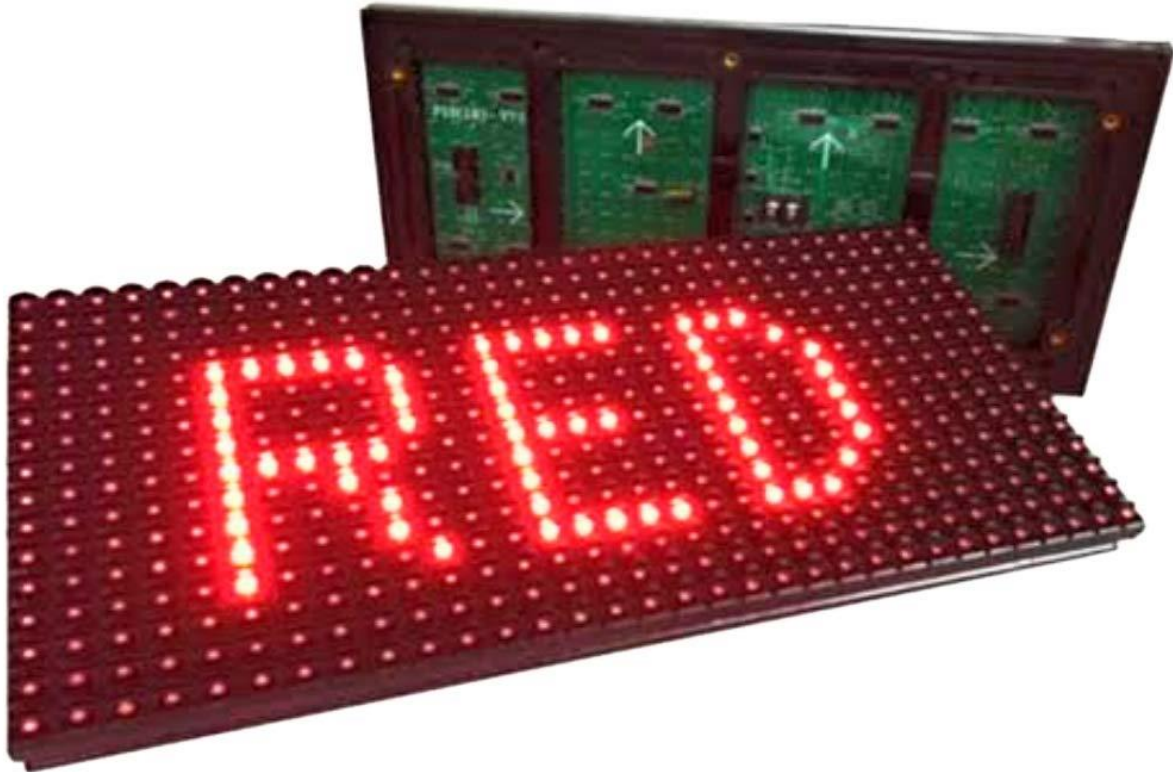




Proyecto 1 Septiembre – Diciembre 2022

LED Displays



1 Introducción

Los dispositivos de salida visuales basados en puntos en matriz (mejor conocidos por su nombre en inglés *Dot-Matrix Display*) se han popularizado enormemente en la última década gracias a los diodos emisores de luz (LED). De particular importancia es su capacidad para mostrar salidas de texto. Usted a sido seleccionado para diseñar el controlador de uno de estos dispositivos.

2 Requerimientos del programa

Diseñe y escriba un programa en Assembly capaz de recibir un *string* dado por el usuario e indicar cómo debe ser mostrado en el *display*. Debe presentársele al usuario un sencillo *prompt* para que introduzca su texto, luego de lo cual el programa imprimirá la salida de forma tal que un espacio indica un LED apagado, y un asterisco indica un LED encendido

Así, por ejemplo, la entrada “HOLA MUNDO” podría producir la salida

```

*   *   ***   *           *           *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*****   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
*   *   ***   *****   *   *   *   *   *   *   *   *   *   *

```

2.1 Organización del programa

Debe diseñar una forma de almacenamiento para el *font* que tenga la representación de cada una de las 26 letras ASCII (no es necesario distinguir entre mayúsculas y minúsculas). La estructura de datos debe ser leída únicamente a través de funciones. Las funciones deben usar el *stack* correctamente. Debe tener una función que ensamble las representaciones de las letras y otra que las imprima.

Deben permitirse *strings* de entrada de hasta 100 caracteres; sin embargo, la salida no debe imprimir líneas de más de 120 caracteres. Si cada letra ocupa 6 caracteres (incluyendo el separador), se debe dividir el *string* en segmentos de 20 caracteres (sin importar que esto corte palabras en dos). Así, por ejemplo, el *string* “EN UN PRINCIPIO TODO ERA OSCURIDAD Y DIOS DIJO HAGASE LA LUZ Y LA LUZ SE HIZO”, se debe imprimir como”

```

EN UN PRINCIPIO TODO
ERA OSCURIDAD Y DIO
S DIJO HAGASE LA LUZ
Y LA LUZ SE HIZO

```

A este efecto, debe haber una función que efectúe esta división.

2.2 Diseño del *font*

Debe diseñar su propio *font* para este proyecto. Todas las letras deben tener el mismo tamaño tanto en ancho como en alto. Las letras deben tener un ancho mínimo de 5 caracteres. Se espera que no haya dos grupos con *fonts* iguales. Observe, por ejemplo, que la letra A se puede representar de cualquiera de las siguientes formas en una matriz de 5×5:

```

***   *   *   *   *
*   *   *   *   *   *   *   *
***** *****   *   *   *   *   *
*   *   *   *   *   *   *   *   *
*   *   *   *   *   *   *   *   *

```

3 Informe

Debe entregar un informe donde se detallen las decisiones tomadas en el proyecto incluyendo:

- Introducción, dando la motivación y estructura del informe
- Diseño del Font
- Diseño de la Estructura de Datos
- Dificultades Encontradas en la implementación u otras decisiones posteriores
- Conclusiones, dando el estado actual del programa y las lecciones aprendidas

4 Evaluación

El proyecto tiene una ponderación de 10 puntos. Se asignarán

- 5 puntos por código
 - 1 punto por su estructura de datos
 - 1 punto por la(s) función(es) de acceso a la estructura de datos
 - 1 punto por su función para dividir el string de entrada
 - 1 punto por su función para ensamblar el string de salida
 - 1 punto por la impresión del string de salida en sí
- 2,5 puntos por ejecución
 - 0,1 puntos por poder imprimir cada letra correctamente con las restricciones pedidas
- 2,5 puntos por su informe
 - 0,5 puntos por su introducción
 - 0,5 puntos por el diseño del Font
 - 0,5 puntos por el diseño de la estructura de datos
 - 0,5 puntos por las dificultades encontradas
 - 0,5 puntos por sus conclusiones

Todo su código debe estar debidamente documentado y siguiendo las convenciones de MIPS.

El programa debe correr sin errores.