



UNIVERSIDAD AUTONOMA DE SAN LUIS POTOSÍ

FACULTAD DE INGENIERÍA

ÁREA DE CIENCIAS DE LA COMPUTACIÓN

“LABORATORIO PROGRAMACIÓN DE SISTEMAS”

**PRACTICA 8:
“CARGADOR - MAPA DE MEMORIA”**

**PROFESOR:
ING. AGUSTÍN HERNÁNDEZ GARCÍA**

ALUMNO: Braulio Alejandro García Rivera

CLAVE ÚNICA: 239196

FECHA: 02/11/2020

SEMESTRE: 2020-2021/I

1. INTRODUCCIÓN

En este documento se explica cómo se implementó la carga de un programa de la SIC Estándar utilizando una matriz que simula el mapa de memoria y los registros que componen esta arquitectura.

2. INSTALACIÓN Y CONFIGURACIÓN

En Visual Studio 2017 se instalaron las siguientes extensiones:

- Antlr4Code
- ANTLR Language Support

También se instalaron las bibliotecas de Antlr4 como paquetes NuGet:

- Antlr4 v4.6.6
- Antlr4.CodeGenerator v4.6.6
- Antlr4.Runtime v4.6.6

La configuración se realizó en base a la explicación del profesor y los ejemplos de la plataforma didacTIC.

3. INTERFAZ GRÁFICA

Se selecciona el botón MAPA DE MEMORIA para que nos muestre la siguiente interfaz:

The screenshot shows a Windows application window titled 'Form1'. It contains a memory map table and a register status table.

Dirección	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2010	04	20	27	50	A0	1F	54	A0	23	2C	20	2A	38	20	13	48
2020	4F	4C	41	FF	FF	FF	FF	00	00	00	00	00	04	FF	FF	FF

Tamaño del programa: 00001D

Registro	Valor Inicial	Valor Final
CP	002010	
A	FFFFFF	
X	FFFFFF	
L	FFFFFF	
CC	null	

Después seleccionamos el botón abrir para cargar un archivo .obj, posteriormente muestra el CP y el nombre del archivo que se seleccionó, así como la carga del mapa de memoria correspondiente a este último.

4. IMPRESIÓN DEL MAPA DE MEMORIA

A través del constructor ObjetoMem se asigna una matriz donde se rellena con FF cada posición, verifica que haya código objeto y dependiendo el tamaño del programa se crea, se calculan las direcciones válidas y se va de diez en diez en valor hexadecimal, se lee el registro H para obtener el tamaño del programa, ahora continua a leer registros T para poner la dirección inicial y posicionarse en el registro, se insertan los datos del registro de dos en dos en la matriz, y al final se imprime la matriz en un DataGridView para poder visualizar los valores y el tamaño del programa, así como los valores de los registros en otro DataGridView.

5. PROBLEMAS

Un problema fue al calcular el tamaño de las filas del DataGridView y se implementó un patrón matemático para poder visualizar las direcciones correctas, se hizo una prueba comparándolo con un algoritmo de fuerza bruta.

6. POSIBLES MEJORAS

Una posible mejora seria la estructura del mapa de memoria para poder cargar los valores de una manera más practica evitando varias confusiones, ya que esta implementación fue complicada.

7. CONCLUSIÓN

En esta práctica se cargaron los valores de un programa .obj en el mapa de memoria para poder visualizar estos últimos de manera correcta, fue una practica complicada.