Rogelio Daniel Gonzalez Nieto

LABORATORIO PROGRAMACION DE SITEMAS  AREA DE CIENCIAS DE LA COMPUTACION

REPORTE PRACTICA NO 3

# OBJETIVO DE LA PRACTICA

Realizar el paso 1 para el ensamblador de la sic estándar el cual es:

* Asignar direcciones a todas las proposiciones del programa
* Guardar las direcciones asignadas a todas las etiquetas para usarse en el paso 2

(TABSIM)

* Realizar algún procesamiento de las instrucciones para el ensamblador (BYTE,

RESW, etc.)

Para el cálculo de direcciones siempre se debe de sumar 3 bits a la ultima dirección asignada a excepción de cuando se tienen una cadena, en ese caso se suma 1 bite por cada carácter, un valor hexadecimal se suma 1 bite por cada 2 caracteres y un valor decimal se multiplica por 3 y se convierte a hexadecimal, ese valor se suma a la dirección.

# CODIGO DE LA PRACTICA

private void calcularDirecciones()

{

string dirHexa = "";

int iDirDec = 0;

int iPos = 0;

int iTotal = 0;

foreach (var linea in this.ensamblador.programa)

{

if (this.ensamblador.programa.IndexOf(linea) == 0)

{

linea.sDireccionHEXA = lDirecciones[0].sDireccionHexadecimal;

dirHexa = lDirecciones[0].sDireccionHexadecimal;

dirHexa = dirHexa.Trim('H');

iDirDec = Convert.ToInt32(dirHexa.ToString(), 16);

//iDirDec = lDirecciones[0].iNumeroDecimal;

}

else

{

if (this.lDirecciones[iPos].cRegistro == 'C')

{

iTotal = this.lDirecciones[iPos].sPalabra.Length;

iDirDec = iDirDec + iTotal;

dirHexa = $"{iDirDec:X}";

linea.sDireccionHEXA = dirHexa;

}

else if (this.lDirecciones[iPos].cRegistro == 'X' && this.lDirecciones[iPos].sDireccionHexadecimal != null)

{

iTotal = this.lDirecciones[iPos].sDireccionHexadecimal.Length;

if (iTotal % 2 == 0)

iTotal = iTotal / 2;

else

iTotal = (iTotal / 2) + 1;

iDirDec = iDirDec + iTotal;

dirHexa = $"{iDirDec:X}";

linea.sDireccionHEXA = dirHexa;

}

else if (this.lDirecciones[iPos].iNumeroDecimal != 0)

{

int iNuevo = this.lDirecciones[iPos].iNumeroDecimal \* 3;

iDirDec = iDirDec + iNuevo;

//iDirDec = iDirDec \* iNuevo;

//dirHexa = $"{iDirDec:X}";

dirHexa = $"{iDirDec:X}";

linea.sDireccionHEXA = dirHexa;

}

else

{

iDirDec = iDirDec + 3;

dirHexa = $"{iDirDec:X}";

linea.sDireccionHEXA = dirHexa;

}

}

iPos++;

}

iPos = 1;

foreach (var linea in this.ensamblador.programa)

{

this.dataGridView1.Rows.Add(linea.sDireccionHEXA, linea.sEtiqueta, linea.sCodigoOp, linea.sDireccion);

iPos++;

}

}

# EJECUCION DEL PROGRAMA





