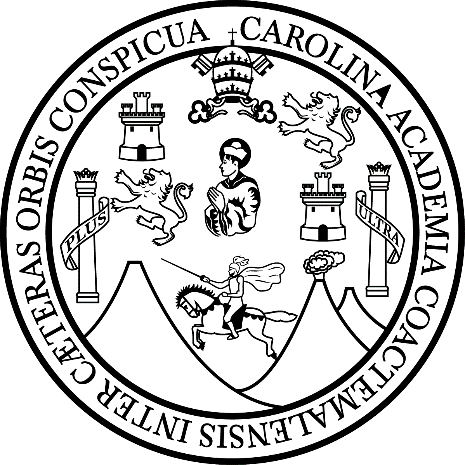
**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Facultad de Ingeniería

Escuela de Ciencias y Sistemas



PROYECTO 2

Manual Tecnico

**Rodrigo Eduardo Carcuz Ortega**

201700633

Guatemala de la Asunción 30 de abril de 2021

Contenido

[Objetivo General: 2](#_Toc69083725)

[Objetivos Específicos: 2](#_Toc69083726)

[RESUMEN : **¡Error! Marcador no definido.**](#_Toc69083727)

[Cargar archivo **¡Error! Marcador no definido.**](#_Toc69083728)

[Ordenar **¡Error! Marcador no definido.**](#_Toc69083729)

[Generar reporte **¡Error! Marcador no definido.**](#_Toc69083730)

# Objetivo General:

● Aplicar los conocimientos adquiridos en el curso sobre el lenguaje ensamblador.

# Objetivos Específicos:

● Aplicar el conocimiento de operaciones básicas a nivel ensamblador.

● Conocer el funcionamiento de las interrupciones.

● Comprender el uso de la memoria en los programas informáticos.

● Consolidar el uso del modo video en lenguaje a bajo nivel.

● Emplear y manipular la paleta de colores del modo video.

● Realizar cálculos para implementar los delays en bajo nivel.

# Gramatica:

S10:S10 Scl {$$=$1+"$"+$2;}

|Scl{$$=$1;}

;

Scl:S1{$$=$1;};

S1: Imprimir{$$=$1;}

|Func{$$=$1;}

|Metodo{$$=$1;}

|LFunc{$$=$1;}

|Var{$$=$1;}

|Avar{$$=$1;}

|DecVec{$$=$1;}

|ModifVec{$$=$1;}

|Listas{$$=$1;}

|Modilist{$$=$1;}

|Addlist{$$=$1;}

|ejecuciones{$$=$1;}

;

ejecuciones : exec Identificador '(' ')' ';' {$$=$1+"$"+$2+"$"+$5;}

|  exec Identificador '(' Lista\_E ')' ';' {$$=$1+"$"+$2+"$"+$4+"$"+$6;};

Var: Tipo Lista\_Id Var1{$$=$1+"$"+$2+"$"+$3;};

Tipo: int{$$=$1;}

|double{$$=$1;}

|boolean{$$=$1;}

|char{$$=$1;}

|String{$$=$1;};

Lista\_Id: Lista\_Id ',' Identificador{$$=$1+"$"+$2+"$"+$3;}

|Identificador{$$=$1;};

Var1:'=' e ';'{$$=$3;}

|';'{$$=$1;};

Avar: Identificador '(' ')' ';'{}

|Identificador '=' e ';'{};

Func: Tipo Identificador '(' Lista\_Parametro ')' '{' Sent123 {$$=$1+"$"+$2+"$"+$3+"$"+$4+"$"+$5+"$"+$6+"$"+$7;}

|Tipo Identificador '(' ')' '{' Sent123 {$$=$1+"$"+$2+"$"+$3+"$"+$4+"$"+$5+"$"+$6;};

Sent123: Sent1 return e ';' '}'{$$=$1+"$"+$5;}

|return e ';' '}'{$$=$4;}

| Sent1 '}'{$$=$1+"$"+$2;}

|'}'{$$=$1;};

Lista\_Parametro: Var2 {$$=$1;};

Var2: Var2 ',' Tipo Identificador{$$=$1+"$"+ $2+"$"+ $3+"$"+ $4;}

|Tipo Identificador{$$=$1+"$"+ $2;};

Metodo: void Identificador '(' Lista\_Parametro ')' '{' Sent113 {$$=$1+"$"+$2+"$"+$3+"$"+$4+"$"+$5+"$"+$6+"$"+$7;}

|void main '(' ')' '{' Sent113  {$$=$1+"$"+$2+"$"+$3+"$"+$4+"$"+$5+"$"+$6;}

|void Identificador '(' ')' '{' Sent113 {$$=$1+"$"+$2+"$"+$3+"$"+$4+"$"+$5+"$"+$6;};

Sent113: Sent1 return ';' '}'{$$=$1+"$"+$4;}

| return ';' '}'{$$=$3;}

|Sent1 '}' {$$=$1+"$"+$2;}

|'}'{$$=$1;};

LFunc: Identificador '(' Lista\_E ')' ';'{$$=$1+"$"+$2+"$"+$3+"$"+$4;};

Lista\_E: Lista\_E ',' e{$$=$1+"$"+$2+"$"+$3;}

|e{$$=$1;};

Imprimir: print '(' e ')' ';'{};

IF: if '(' e ')' '{' Sent11  {$$=$6;}

|if '(' e ')' '{' Sent11  else '{' Sent11 {$$=$6+"$"+$9;}

|if '(' e ')' '{' Sent11  ELS{$$=$6+"$"+$7;}

|if '(' e ')' '{' Sent11  ELS else '{' Sent11 {$$=$6+"$"+$7+"$"+$10;};

ELS: ELS else if '(' e ')' '{' Sent11  {$$=$1+"$"+$8;}

|else if '(' e ')' '{' Sent11 {$$=$7;};

Sent11: Sent1 '}' {$$=$1;}

|Sent1 Senten ';' '}' {$$=$1;}

|Senten ';' '}' {}

|'}'{};

Senten: break {}

|return {}

|return e {}

|continue {};

    e: e '&&' e

    {$$=$1+"$"+$2+"$"+$3;}

    | e '||' e

        {$$=$1+"$"+$2+"$"+$3;}

    |'!' e

        {$$=$1+"$"+$2;}

    | '-' e %prec UMINUS

        {$$=$1+"$"+$2;}

    | '\*''-' e %prec UMINUS

        {}

    | '(' e ')'

        {}

    | e '==' e

        {}

    | e '!=' e

    {}

    | e '<=' e

    {}

    | e '>=' e

        {}

    | e '<' e

    {}

    | e '>' e

        {}

    | e '^' e

    {}

    | e '/' e

        {}

    | e '\*' e

    {}

    | e '%' e

        {}

    | e '+' e

        {}

    | e '-' e

        {}

    |CADENA

    {}

    |Identificador '[' e ']'

        {}

    |CARACTER

        {}

    | ENTERO

        {}

    | DECIMAL

        {}

    |true

        {}

    |false

        {}

    |LFunc

        {}

    |Tipo

        {}

    |tolower '(' e ')'

        {}

    |toupper '(' e ')'

        {}

    |lenght '(' e ')'

        {}

    |truncate '(' e ')'

        {}

    |round '(' e ')'

        {}

    |typeof '(' e ')'

        {}

    |tostring '(' e ')'

        {}

    |tochararray '(' e ')'

        {}

    |Identificador

        {};

Swit: switch '(' e ')' '{' Cas Def '}' {$$=$6+"$"+$7;};

Cas: Cas case e ':' Sent1 {$$=$1+"$"+$5;}

|Cas case e ':' {$$=$1;}

|Cas case e ':' Sent1 break ';'{$$=$1+"$"+$5;}

|Cas case e ':' break ';'{$$=$1;}

|case e ':' {}

|case e ':' Sent1 {$$=$4;}

|case e ':' break ';'{}

|case e ':' Sent1 break ';'{$$=$4;};

Def: default ':' Sent1 break ';'{$$=$3;}

|default ':' break ';'{}

|default ':' Sent1 {$$=$3;}

|default ':' {};

Whil: while '(' e ')' '{' Sent111 {$$=$6;};

Sent111: Sent1 Senten ';' '}'{$$=$1;}

|Sent1 '}'{$$=$1;}

|Senten ';' '}'{}

|'}'{};

Do: do '{' Sent11  while '(' e ')' ';'{$$=$3;};

Fo: for '(' Fo1 ';' e ';'  Aum ')' '{' Sent112  {$$=$10;};

Sent112: Sent1 Senten ';' '}'{$$=$1;}

|Senten ';' '}'{}

|Sent1 '}'{$$=$1;}

|'}'{};

Fo1: Tipo Identificador '=' e{}

| Identificador '=' e{};

Aum:AUMETO{}

|DECREMENTO{};

    DecVec: Tipo '[' ']' Identificador '=' new Tipo '[' e ']' ';' {$$=$1+"$"+$4+"$"+$9;}

     | Tipo '[' ']' Identificador '=' '{' Lista\_E '}' ';' {$$=$1+"$"+$4+"$"+$7;};

    ModifVec: Identificador '[' ENTERO ']' '=' e  ';'{$$=$1+"$"+$3+"$"+$6;} ;

    Listas: list '<' Tipo '>' Identificador '=' new list  '<' Tipo '>' ';' {$$=$1+"$"+$3+"$"+$5;}

         | list '<' Tipo '>' Identificador '=' tochararray '(' e ')' ';' {$$=$1+"$"+$3+"$"+$5;};

    Modilist : Identificador '[' '[' ENTERO ']' ']' = e ';' {$$=$1+"$"+$4+"$"+$8;} ;

    Addlist : Identificador '.' add '(' e ')' ';' {$$=$1+"$"+$3;} ;

    Sent1: Sent1 IF{$$=$1+"$"+$2;}

    |Sent1 Swit{$$=$1+"$"+$2;}

    |Sent1 Whil{$$=$1+"$"+$2;}

    |Sent1 Do{$$=$1+"$"+$2;}

    |Sent1 Var{$$=$1+"$"+$2;}

    |Sent1 Imprimir{$$=$1+"$"+$2;}

    |Sent1 LFunc{$$=$1+"$"+$2;}

    |Sent1 Avar{$$=$1+"$"+$2;}

    |Sent1 Fo{$$=$1+"$"+$2;}

    |Sent1 DecVec{$$=$1+"$"+$2;}

    |Sent1 ModifVec{$$=$1+"$"+$2;}

    |Sent1 Listas {$$=$1+"$"+$2;}

    |Sent1 Modilist{$$=$1+"$"+$2;}

    |Sent1 ejecuciones{$$=$1+"$"+$2;}

    |Sent1 Addlist{$$=$1+"$"+$2;}

    |IF{$$=$1;}

    |Swit{$$=$1;}

    |Whil{$$=$1;}

    |Do{$$=$1;}

    |Var{$$=$1;}

    |Imprimir{$$=$1;}

    |LFunc{$$=$1;}

    |Avar{$$=$1;}

    |DecVec{$$=$1;}

    |ModifVec{$$=$1;}

    |Listas{$$=$1;}

    |Modilist{$$=$1;}

    |Addlist{$$=$1;}

    |ejecuciones{$$=$1;}

    |Fo{$$=$1;};