

**INSTITUTO TECNOLÓGICO DE TOLUCA**

Ingeniería en Sistemas Computacionales

Lenguajes y Autómatas II

Ensamblador

Integrantes:

Hernández Carmona Cristopher

Muñoz Sánchez René

#### Metepec, Estado de México, noviembre de 2017

**S.E.P S.E.S.T.N.M. T.N.M**

Contenido

[Código Fuente 3](#_Toc499749877)

[Código Intermedio 3](#_Toc499749878)

[Tabla de símbolos 4](#_Toc499749879)

[Cuádruples 5](#_Toc499749880)

[Salida 6](#_Toc499749881)

# Código Fuente

#INICIO

int a = 0;

int b = (3 \* 2);

int c = 2 \* a;

if ((a-0\*b\*1+c-0)>=(2\*3+5-3+a-0)){

msj a;

msj "Hola, mundo";

}

#

# Código Intermedio

a = 0

T1 = 3 \* 2

b = T1

T2 = 2 \* a

c = T2

T3 = 0 \* b

T4 = T3 \* 1

T5 = a - T4

T6 = T5 + c

T7 = T6 - 0

T8 = 2 \* 3

T9 = T8 + 5

T10 = T9 - 3

T11 = T10 + a

T12 = T11 - 0

IF T7 >= T12 GOTO 10

GOTO 20

10:

PRINT a

PRINT "Hola , mundo "

GOTO 30

20:

# Tabla de símbolos

|  |  |  |
| --- | --- | --- |
| TOKEN | LEXEMA | DESCRIPCION |
| 0 | #INICIO | INIT |
| 2 | int | ENTERO |
| V | a | VARIABLE |
| 21 | = | OPERANDO |
| I | 0 | NUMERO ENTERO |
| 29 | ; | PUNTO\_Y\_COMA |
| 2 | int | ENTERO |
| V | b | VARIABLE |
| 21 | = | OPERANDO |
| X | ( 3 \* 2 ) | EXPRECION |
| 29 | ; | PUNTO\_Y\_COMA |
| 2 | int | ENTERO |
| V | c | VARIABLE |
| 21 | = | OPERANDO |
| X | 2 \* a | EXPRECION |
| 29 | ; | PUNTO\_Y\_COMA |
| 7 | if | SI |
| 23 | ( | AB\_PAR |
| 23 | ( | AB\_PAR |
| V | a | VARIABLE |
| 21 | - | OPERANDO |
| I | 0 | NUMERO ENTERO |
| 21 | \* | OPERANDO |
| V | b | VARIABLE |
| 21 | \* | OPERANDO |
| I | 1 | NUMERO ENTERO |
| 21 | + | OPERANDO |
| V | c | VARIABLE |
| 21 | - | OPERANDO |
| I | 0 | NUMERO ENTERO |
| 24 | ) | CERR\_PAR |
| 22 | >= | COMPARADOR |
| 23 | ( | AB\_PAR |
| I | 2 | NUMERO ENTERO |
| 21 | \* | OPERANDO |
| I | 3 | NUMERO ENTERO |
| 21 | + | OPERANDO |
| I | 5 | NUMERO ENTERO |
| 21 | - | OPERANDO |
| I | 3 | NUMERO ENTERO |
| 21 | + | OPERANDO |
| V | a | VARIABLE |
| 21 | - | OPERANDO |
| I | 0 | NUMERO ENTERO |
| 24 | ) | CERR\_PAR |
| 24 | ) | CERR\_PAR |
| 25 | { | AB\_LLAVE |
| 1 | msj | MENSAJE |
| V | a | VARIABLE |
| 29 | ; | PUNTO\_Y\_COMA |
| 1 | msj | MENSAJE |
| T | "Hola , mundo " | VALOR TEXTO |
| 29 | ; | PUNTO\_Y\_COMA |
| 26 | } | CERR\_LLAVE |
| 14 | # | FIN |

# Cuádruples

|  |  |  |  |
| --- | --- | --- | --- |
| Operador1 | Operador2 | Operando | Resultado |
| 0 |  | = | a |
| 3 | 2 | \* | T1 |
| T1 |  | = | b |
| 2 | a | \* | T2 |
| T2 |  | = | c |
| 0 | b | \* | T3 |
| T3 | 1 | \* | T4 |
| a | T4 | - | T5 |
| T5 | c | + | T6 |
| T6 | 0 | - | T7 |
| 2 | 3 | \* | T8 |
| T8 | 5 | + | T9 |
| T9 | 3 | - | T10 |
| T10 | a | + | T11 |
| T11 | 0 | - | T12 |
| T7 | T12 | >= | 10 |
|  |  | GOTO | 20 |
|  |  |  | 10 |
| a |  |  | PRINT |
| "Hola , mundo " |  |  | PRINT |
|  |  | GOTO | 30 |
|  |  |  | 20 |

# Salida

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Código ensamblador** | **Salida** |
| Encabezado | org 100h\n  include 'emu8086.inc'\n  DEFINE\_PRINT\_STRING  DEFINE\_PRINT\_NUM  DEFINE\_SCAN\_NUM  DEFINE\_PRINT\_NUM\_UNS  TITLE  .MODEL SMALL  .STACK 64M  .DATA  .CODE  END | org 100h  include 'emu8086.inc'  DEFINE\_PRINT\_STRING  DEFINE\_PRINT\_NUM  DEFINE\_SCAN\_NUM  DEFINE\_PRINT\_NUM\_UNS  TITLE Codigo Ensamblador  .MODEL SMALL  .STACK 64M  .DATA  a dw ?  b dw ?  T2 dw ?  c dw ?  T5 dw ?  T6 dw ?  T7 dw ?  T11 dw ?  T12 dw ?  msj1 db 10,13,"Hola , mundo " ,'$'  .CODE  inicio PROC FAR  MOV AX,@data  MOV DS,AX  MOV AX,0  MOV a,AX  MOV AX,6  MOV b,AX  MOV AX,a  ADD AX,a  MOV T2,AX  MOV AX,T2  MOV c,AX  MOV AX,a  SUB AX,0  MOV T5,AX  MOV AX,T5  ADD AX,c  MOV T6,AX  MOV AX,T6  SUB AX,0  MOV T7,AX  MOV AX,8  ADD AX,a  MOV T11,AX  MOV AX,T11  SUB AX,0  MOV T12,AX  MOV CX,T7  CMP CX,T12  JAE A10  JMP A20  A20:  A10:  MOV AX,a  CALL PRINT\_NUM  MOV AH,09h  LEA DX,msj1  INT 21h  A30:  JA A20  MOV AH,4CH  INT 21h  inicio ENDP  END  ret |
| Variables | n db 0 |
| Mensajes | num db 10,13, “n”, ‘$’ |
| Constante | n db va |
| .CODE | .CODE  inicio PROC FAR  MOV AX,@data  MOV DS,AX |
| Suma | MOV AX,a  ADD AX,b  MOV r, AX |
| Resta | MOV AX, a  SUB AX, b  MOV r, AX |
| Multiplicación | MOV AX, a  MUL b  MOV r, AX |
| División | MOV AX, a  XOR DX,DX  DIV b  MOV r, AX |
| Igual | MOV CL, a  CMP CL, b  JE r  JMP rn |
| Diferente | MOV CL, a  CMP CL, b  JNE r  JMP r |
| Mayor | MOV CL, a  CMP, b  JA r  JMP rn |
| Menor | MOV CL, a  CMP, b  JB r  JMP rn |
| Mayor o Igual | MOV CL, a  CMP CL,b  JAE r  JMP rn |
| Menor Igual | MOV CL, a  CMP CL, b  JBE r  JMP rn |
| Retorno | JA A |
| Lectura | CALL SCAN\_NUM  MOV r,CX |
| Imprimir mensaje | MOV AH, 09h  LEA DX, a  INT 21h |
| Imprimir variable | MOV AX, a  ADD DL, 30h  MOV AH, 02  INT 21h |
| Imprimir asignación | MOV AX, a  MOV b, AX |
| Final | MOV AH, 4CH  INT 21h  inicio ENDP  END  ret |