Forma

Descripción generada automáticamente con confianza baja

**Nombre de la materia**

**Lenguajes y Autómatas I**

**Nombre del alumno**

**Guzmán Zúñiga Sergio Mateus**

**Actividad**

**Caso de estudio**

**Unidad IV**

**Análisis Léxico**

**Nombre del Profesor**

**Villalvazo Mateos José Raúl**

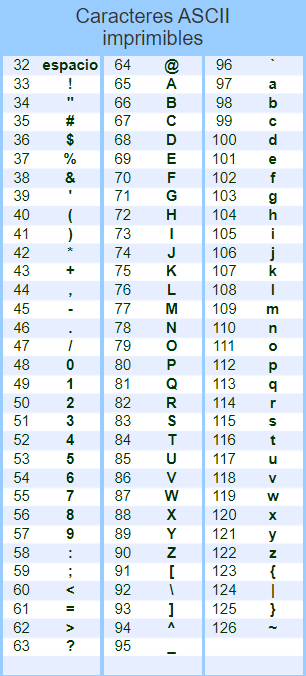
**Fecha**

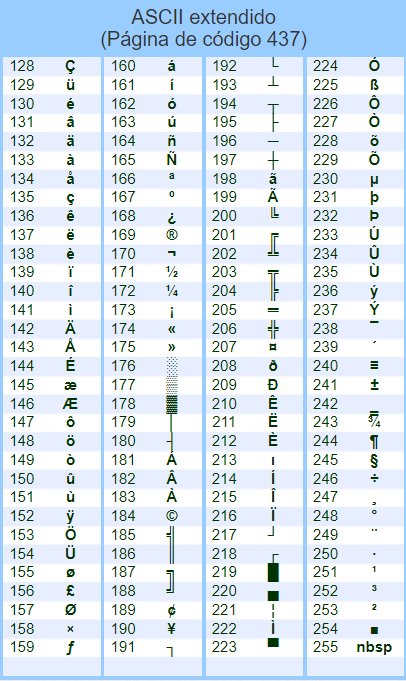
**9 de diciembre del 2023**

**Caso de estudio**

**Introduccion**

En este caso de estudio veremos las bases fundamentales para nuestro proyecto de lenguaje ensamblador o automata, desde el origen de los simbolos **“el codigo ascii”**, seleccionando los elementos que usare en mi proyecto, al igual representare los tipos de operadores que utilizare, el tipo de dato, las sentencias y palabras reservadas que ya se utilizan en algun otro lenguaje, adaptandolo al ambiente en el que sea compatible con mi automata.

Tabla

Descripción generada automáticamente**CODIGO ASCII**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente Tabla

Descripción generada automáticamente Tabla

Descripción generada automáticamente 

Tabla

Descripción generada automáticamente Tabla

Descripción generada automáticamente Tabla

Descripción generada automáticamente

**Alfabeto**

**Declaración de los componentes de mi lenguaje…**

|  |  |
| --- | --- |
| **Entradas** | |
| SCANNER | **Scanner()** |

|  |  |
| --- | --- |
| **Tipo de datos** | |
| INT | **int** |
| STRING | **string** |
| DOUBLE | **double** |

|  |  |
| --- | --- |
| **Salida** | |
| PRINT | **print** |

|  |  |
| --- | --- |
| **Operadores Aritméticos** | |
| Suma | **+** |
| Resta | **-** |
| Multiplicación | **\*** |
| División | **/** |
| Modulo | **%** |

|  |  |
| --- | --- |
| **Operadores Racionales** | |
| Igual a | **==** |
| No igual a | **!=** |
| Mayor que | **>** |
| Menor que | **<** |
| Mayor o igual que | **>=** |
| Menor o igual que | **<=** |

|  |  |
| --- | --- |
| **Operadores lógicos** | |
| AND | **||** |
| OR | **|** |
| NOT | **!** |

|  |  |
| --- | --- |
| **Operador de asignación** | |
| ASIGNACION | **=** |

|  |  |
| --- | --- |
| **Operadores de incrementación** | |
| INCREMENTO | **++** |
| DECREMENTO | **--** |

|  |  |
| --- | --- |
| **Condicionales** | |
| if {  } | **if():**  **ifend** |
| if {  } else {  } | **if():**  **ifend**  **else:**  **elseend** |
| for {  } | **for():**  **forend** |
| while() {  } | **while():**  **whileend** |

**GRAMÁTICAS:**

**SIGNOS TERMINALES = {**a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, \*, \_, \, /, &, !, =, <, >, |, %, ., : , ,, ; (, ), [, ], ?, ^, inicio, iniciovar, scanner, int, string, double, finvar, print, if, ifend,, for, forend, while, whileend, else,elseend,||, |, !, <=, >=, ++, --, ==, !=,),(,\s, \n, \t, ., **}**

**SIGNOS NO TERMINALES = {**<nums>, <Bloque\_codigo>, <Bloque\_variables>, <codigo>, <variable\_aceptada>, <tipo>, <var\_nom>, <cabeza\_var>, <cuerpo\_var>, <var\_declaracion>, <cuerpo\_comillas>, <contenido\_comillas>, <abecedario>, <tokens>,<while\_token>, <if\_token>, <for\_token>, <print\_token>, <else\_token>, <scanner\_token>, <new\_val\_token>, <cuerpo\_while>, <condicion>, <cabeza\_condicion>, <cuerpo\_condicion>, <val\_comparable>, <cuerpo\_if>, <cuerpo\_for>, <cuerpo\_print>, <cuerpo\_scanner>, <else\_token>, <for\_i>, <for\_f>, <new\_val\_token>, <cuerpo\_new\_val>, <operacion>, <clave\_while>, <clave\_for>, <clave\_if>, <clave\_scanner>, <clave\_print>, <clave\_new\_val>, <clave\_else>, <whileend\_token>, <forend\_token>, <ifend\_token>, <aritmeticos>, <space?>, <linea\_espacio>, <fin\_token;>, <fin\_token:>, <igualdad>, <tipo\_int>, <tipo\_double>, <tipo\_string>, <Token\_inicio>, <Token\_fin>, <Token\_iniciovar>,<Token\_finvar>, <digito>, <letras>, <not>, <\_.\_>, <contador>, <comilla\_abre>, <comilla\_cierra>, <caracteres\_xtra>