Gramática manual

Simbolo inicial

Include B

Class C{

Instrucciones\_segundonivel

}

Inicio 🡪 incluirlibreria instrucciones\_primernivel | instrucciones\_primernivel

Incluirlibreria 🡪 INCLUIR\_BIBLIOTECAS ID

Instrucciones\_primernivel 🡪 defclass masclases

defclass 🡪 CLASE ID bloque\_codigo

bloque\_codigo 🡪 LLAVEIZQ inicioaux LLAVEDER

inicioaux 🡪 | instrucciones\_segundonivel

masclases 🡪 instrucciones\_primernivel |

instruccion\_segundonivel        : defclass

                                | defvar

                                | sentencia\_switch

                                | ecuaciones

                                | arreglos

                                | touppercase

                                | despliegue\_arreglo

                                | comentario

                                | acceso\_variables

                                | ciclo\_for

                                | returnstmt

                                | pregunton\_comodijoerasmo

                                | definir\_valor\_variable

                                | deffunction

                                | concatenacion\_cadenas

                                | obtenervalores

                                | colas

| ordenar\_fifo

1. Definición de Variables:

int x = 0;

DefVar 🡪 TIPO\_DATO ID ASIGNACION NUM

1. Definición de Funciones:

Function X(int x, int y){

X=5;

}

deffunction 🡪 CREACION\_FUNCIONES ID PARENTIZQ parametros masparametros PARENTDER LLAVEIZQ inicioaux LLAVEDER

parametros 🡪 TIPO\_DATO ID |

masparametros 🡪 COMA parámetros |

1. Ciclos

ciclos 🡪 BUCLES bloques

bloques 🡪 bloque\_for | bloque\_while | bloque \_do

1. Ciclos while:

                            while(x<3){

instrucciones\_segundonivel

                            }

ciclos 🡪 BUCLES bloques

bloques 🡪 bloque\_for | bloque\_while | bloque \_do

bloque\_while 🡪 PARENTIZQ condicion PARENTDER bloque\_codigo

condicion 🡪 valor OPERADORES\_RELACIONALES valor mascondiciones

valor 🡪 NUM | ID |TEXTO

mascondiciones 🡪 OPERADOR\_LOGICO condición |

1. Ciclo For:

for(**int** h=5;h<=c;h+=5){

**instrucciones\_Segundonivel**

}

ciclos 🡪 BUCLES bloques

bloques 🡪 bloque\_for | bloque\_while | bloque \_do

bloque\_for  🡪 PARENTIZQ defvar condicion PUNTO\_COMA ID ASIGNACION\_C\_OPERACION NUM PARENTDER bloque\_codigo

1. Ciclo do:

do:x<=5 || 5>2{

instrucciones\_segundonivel

}

ciclos 🡪 BUCLES bloques

bloques 🡪 bloque\_for | bloque\_while | bloque \_do

bloque\_do 🡪 DOSPUNTOS condicion bloque\_codigo

1. Condición IF – ELSE – ELSEIF

                            if x>3{

Instrucciones\_Segundonivel

                            }

                            elif x<3{

                               Instrucciones\_Segundonivel

                            }

                            else{

                           Instrucciones\_Segundonivel

                            }

pregunton\_comodijoerasmo 🡪 IF\_CONDICIONAL condicion bloque\_codigo otro\_pregunton

otro\_pregunton 🡪 | ELSE\_CONDICIONAL bloque\_codigo

| ELSE\_IF\_CONDICIONAL condicion bloque\_codigo otro\_pregunton

1. Ecuaciones:

x = +(h,h)

x = \*(h,h)

x = -(h,h)

x = /(h,h)

ecuaciones  : ID ASIGNACION OPERADORES\_MATEMATICOS PARENTIZQ valor\_ecuaciones COMA valor\_ecuaciones PARENTDER

            ;

valor\_ecuaciones        : NUM

                        | ID

                        ;

1. Declaración de arreglos

New números [int,32]

arreglos 🡪 ASIGNACION\_MEMORIA ID CORCHETEIZQ caracteristicas CORCHETEDER

caracteristicas 🡪 TIPO\_DATO definición |

definicion 🡪 COMA NUM

1. Convertir a Mayusculas

ToUppercase(nombre)

touppercase 🡪 CARACTERES\_MAYUS PARENTIZQ ID PARENTDER

1. Despliegue de Arreglo

Números.Print

despliegue\_arreglo 🡪 ID ACCESO\_MODIFICADORES DESPLEGAR\_CARACTERES

1. Acceso a variables

X: global

acceso\_variables 🡪 ID DOSPUNTOS MODIFICADOR\_ACCESO

1. Retornar valores

15\_return;

returnstmt 🡪 valor GUION\_BAJO RETORNO PUNTO\_COMA

Valor 🡪 NUM | ID

1. Comentarios

#Hola Mundo

comentario 🡪 COMENTARIOS

Token COMENTARIOS 🡪 \#.+

1. Sentencia Switch

        switch:D{

            case a:b<x={

instrucciones\_segundonivel

            }

            case b={

instrucciones\_segundonivel

            }

        } endswitch

sentencia\_switch 🡪 SWITCH DOSPUNTOS ID LLAVEIZQ escenarios LLAVEDER TERMINACION\_SWITCH

escenarios 🡪 BLOQUE\_CASE definicioncase masescenarios

definicioncase 🡪 ID asignacion\_case ASIGNACION bloque\_codigo

asignacion\_case 🡪 DOSPUNTOS condición |

masescenarios 🡪 escenarios |

1. Concatenacion de cadenas

string\_concat("Blob", s)

concatenacion\_cadenas 🡪 CONCATENAR\_CADENAS PARENTIZQ valores COMA valores PARENTDER

valores 🡪 TEXTO | ID

1. Obtener valores

X = getabstime

obtenervalores 🡪 ID ASIGNACION OBTENCION\_VALORES

1. Declaración de colas

New números [(int,32)]

colas 🡪 ASIGNACION\_MEMORIA ID CORCHETEIZQ PARENTIZQ caracteristicas PARENTDER CORCHETEDER

caracteristicas 🡪 TIPO\_DATO definición |

definicion 🡪 COMA NUM

1. Ordenar por FIFO para colas

Fifo {numeros}

ordenar\_fifo 🡪 METODOS\_GESTION LLAVEIZQ ID LLAVEDER

1. Def\_valor\_variable

X=5;

definir\_valor\_variable 🡪 ID asignavalor PUNTO\_COMA

asignavalor 🡪 ASIGNACION valor |