**Instituto Tecnológico Superior del Oriente del Estado de Hidalgo**

Ingeniería en Sistemas Computacionales



***Lenguajes y Automatas II***

***T1-2 Definir lenguaje de programación***

*Seventh semester*

**taught by:**

*D.C.C. Elías Ruíz Hernández*

**Por los alumnos:**

*Dominguez Ruiz Serguei Drago - 18030536*

*Guevara Alva Reyes- 18030296*

*Reynoso Aldana Isaac - 18030229*

*APAN, HGO 06 FEBRERO DE 2022*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre (Lex) Sintaxis (Synt) Notas del profesor** | | | | |
| **1.** |  | classe *Nombre* argo::~  variabili::~  ~fine  principale\_inizio::~ | classe Mi\_Clase argo::~  variabili::~  //aquí van variables  ~fine  principale\_inizio::~  // aquí van instrucciones  ~fine  *nombre*\_inizio::~  //aqui van instruccione  ~fine  ~fine\_classe | Es un intérprete de un lenguaje orientado a objetos y como tal, debe tener clase inicial y  método de  inicio (el main) Aquí en el ejemplo se  usaron dos  puntos como  un abrir y cerrar llave, pero tu  podrías usar tabulación como  identación  (Python alike), llaves,  Corchetes o una palabra reservada tip  INIT & END, etc. |
|  | **Encabezado**  **o POO** | ~fine  *nombre*\_inizio::~  ~fine  metodo *nombre* ::\_inizio~  ~fine |
|  |  | ~fine\_classe |
|  |  |  |
|  |  |  |
| **1 Funciones** | | **Imprimir** - stampa::  **Recibe -** ricevere::  **Potencia** - potenza::  **Tipo de dato -** dati:: | stampa:“cadena”:  stampa:$variable:  ricevere :$variable:  stampa:potenza:$num,$exponente:: | Cabe  mencionar que los MatSeno y MatCoseno  también  pueden ir aquí. Al menos debe haber una función de  input y otra de output a |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | consola. |
| **2. Sentencias** | ripetere :$i=0, f=$fin:~  ~fine\_ripetere  se:VERO:~ stampa:”Si es verdadero realiza está línea”:?  altra :  stampa:”Si es falso realiza está línea”:?  ~fine\_se | Repite $variable inicio fin ….  FIN\_repite  (por ejemplo: repite  $contador 40 80  TAB Otras instrucciones FIN\_repite  $x=6?  se :x < 5:~  stampa:”x es menor que 5”:?  altra:  stampa:”x es mayor que 5”::?  ~fine\_altra | *Aquí yo voy a marcar en la*  *revisión si sí o sí sugiero añadir o quitar cosas*  *(debe haber un bucle y un*  *condicional).* |
| **3. Operadores aritméticos** | +  -  \*  /  % | $X=5+6?  $Y=6-4?  $Z = 6\*5?  $Z= :6\*5:\*9-:7/4:? | *Aquí yo te*  *puedo sugerir si usar una*  *modalidad funcional o una modalidad*  *imperativa.* |
| **4. Operadores relacionales** | >  <  >=  <=  !=  == | $x=5>6? (devuelve VERO o IMPOSTORE) | *Aquí puede que te sugiera*  *cambiar a una modalidad funcional o*  *imperativa.* |
| **5. Operadores**  **lógicos** | &  || | $b= $var & $var (devuelve VERO o IMPOSTORE)  $b= $var || $var (devuelve VERO o IMPOSTORE) | *Podrían ser*  *más!* |
| **6. Operadores de asignación** | Sobrecargado:  =  Assign  = | $A = $b |  |
| **7. operadores de uso especial** | infinito | $a = 0/0  No marca error, pero devuelve *infinito*  El Enter separa instrucciones. No admite dos instrucciones en una línea. | *Puede que te*  *sugiera aceptar infinito, NaN, Nulos y otros.*  *Tal como*  *separadores* |
| **8. Operadores de comentarios** | # …. | # Esto es un comentario | *Los clásicos de comentarios*  *con # (python)*  */\* \* /, (java, C + +).* |
| **Variables** | Empiezan con $.  Las booleanas van en mayúsculas | $Ciao  IMPOSTORE  VERO | *Todas las*  *variables*  *globales de la clase deben estar en la*  *sección de* |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | *propiedades de la clase*  *contenedora. Podrías tener variables*  *locales en cada método, pero no serían*  *accesible por otros miembros de la clase.* |
| **constantes** | Dobles comillas para  cadenas, y los  números pueden ir con punto o sin  punto. (float e int) | “Ciao mondo”  5.4798  65 |  |
| **Tipos de datos** | **Booleano** - Bool  **int** -totale  **float** - punto .  **string** - catena “”  **Int\_array** - pronto:: | IMPOSTROE  VERO  5  6.8  “Ciao mondo”  pronto $lista : ”hola” , ”mundo” :  totale $x = 64 | *Sería*  *interesante*  *Multiplicación matricial….*  *¿Verdad?*  *Los tipos*  *mínimos son int, string,*  *array\_int. Float es deseable. List podría ser*  *deseable.* |
| **Declaración de función** | metodo *nombre* ::\_inizio~  *codigo de funcion*  ~fine  metodo ***somma***:$a, $b:\_inizio~  $res= :$a + $b:?  restituzione :$res:?  ~fine  stampa:***somma***:  metodo ***moltiplicare***: $a,$b :\_inizio~  $res= :$a \* $b:?  restituzione :$res:?  ~fine  stampa:***moltiplicare***: | metodo *resta*:$var1 , $var2:\_inizio~  $res = $var1 - $var2  ~fine  chiamare:resta:$var1 , $var2::?  metodo ***somma***:$a, $b:\_inizio~  $res= :$5 + $5:?  restituzione :$res:?  ~fine  stampa:***10***:  metodo ***moltiplicare***: $a,$b :\_inizio~  $res= :$2 \* $5:?  restituzione :$res:?  ~fine  stampa:***10***: | *En algunos equipos podría haber funciones declarativas, funcionales,*  *ambas o*  *simplificadas.*  *(solamente una, solamente un*  *argumento, infinitos*  *argumentos, etc.) Se sugiere evitar la lambda, por el enfoque POO. Recordar que todas las*  *declaraciones de función,*  *SIEMPRE son miembros de la clase.* |

Y una vez creada su tabla con su propuesta, va a escribir un código de varias líneas que haga algo (calcule el producto de dos números ingresados de teclado, calcule alguna operación aritmética, declare variables y haga un ciclo de ejemplo.

Un código posible usando la tabla de arriba podría ser:

**classe** MiPrimeraClase **argo**::**~**

**variavili**::~

totale $a = 5?

punto $b = 7.2?

catena $c = “Que Pez”?

~**finie**

**principale\_inizio**::~

**$c** = 10?

$d = chimare:*sumar*:$var1, chimare:por: $var2 , 4 : : : :?

**ricevere**:$f:?

**stampa**:“valores son”:? **stampa**:$d:?

**stampa:**“El cuadrado de lo que ingresaste es”:? **stampa**: chimare:por:$var1, $var2 : : :?

chimare:decir\_hola:$numero\_veces: :?

~**fine**

**metodo** decir\_hola : $veces : \_inizio~

ripetere :$i=0, f=$veces:~

**stampa**:“Holaaa varias veces”:?

**stampa**:$b:?

#Esto es un comentario no hace nada

~fine\_ripetere

**stampa**:“La variable ingresada fue: ”:?

**stampa:**$veces**:?**

~fine

~fine\_classe

**Comentarios del profesor:**