



UNIVERSIDAD TECNICA PARTICULAR DE LOJA

Carrera de Computación

Estudiante: Giovanni Nahin Ludeña
Pardo

Asignatura: Fundamentos
Computacionales

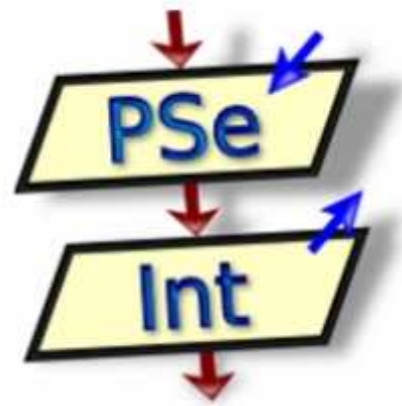
Paralelo: "A"

Docente: Ing. René Elizalde

PSEINT

PSeInt es la abreviatura de los estados de computación de PSeudo Intérprete, una herramienta educativa creada en Argentina, utilizada principalmente por estudiantes para aprender los fundamentos de la programación y el desarrollo de la lógica. Es un software muy popular de su tipo y es ampliamente utilizado en universidades de Latinoamérica y España.

Utiliza pseudocódigo para la solución de algoritmos.



HISTORIA DE SU CREACIÓN

El software comenzó a desarrollarse en octubre del 2003 utilizando Borlan C++ Builder debido a que era esta herramienta que recomendaba la cátedra por la facilidad que brinda para el desarrollo de interfaces visuales y estar presente en los laboratorios de la facultad. comenzó con una prueba de concepto, ya que era el primer proyecto que iba a realizar con C++. Al ser un software para uso exclusivamente didáctico, no se presentan grandes problemas de rendimiento (no se tienen que interpretar algoritmos de miles de líneas sumamente complejos) por lo que pudo tomar cierta libertad en la implementación.

Casi dos años más tarde, gracias a la lista de correo del grupo de Usuarios de Linux del litoral (Lugli) llegó la invitación a participar en la recopilación de Software libre de la colección que se distribuiría gratuitamente entre los docentes que lo solicitaron por el ministerio de Educación de la Nación y llevado a cabo por los grupos gleducar, solar y educ.ar. Sugirió entonces a través de una lista de correo creada para la recopilación del PSEINT en dicho CD.

USO DE "PSEINT"

PSeInt está pensado para asistir a los estudiantes que inician en la construcción de programas o algoritmos computacionales. El pseudocódigo se suele utilizar como primer contacto para introducir conceptos básicos como el uso de estructuras de control, expresiones, variables, etc., sin tener que lidiar con las particularidades de la sintaxis de un lenguaje real. Este software pretende facilitar al

principiante la tarea de escribir algoritmos en este pseudo - lenguaje presentando un conjunto de ayuda y asistencia. Brinda algunas herramientas adicionales que ayuden a encontrar errores y comprender la lógica de los algoritmos.

CARACTERÍSTICAS Y FUNCIONES DE PSeInt

- ✚ Características y Funcionalidades de PSeInt:
- ✚ Presenta herramientas de edición básicas para escribir algoritmos en pseudocódigo en español
- ✚ Permite la edición simultánea de múltiples algoritmos
- ✚ Presenta ayudas para la escritura
- ✚ Auto completado
- ✚ Ayudas Emergentes
- ✚ Plantillas de Comandos
- ✚ Coloreado de Sintaxis
- ✚ Intervalo inteligente
- ✚ Puede ejecutar los algoritmos escritos

ESTRUCTURAS DE CONTROL

La Estructura Repetitiva Mientras (While)

El while es una estructura que se ejecuta mientras la pregunta de control obtiene una respuesta verdadera, cuando la respuesta a la pregunta de control es falsa esta abandona el ciclo. Este tipo de estructura es recomendable cuando dentro del programa se desconoce el momento en que se va abandonar el ciclo. Por ejemplo, si necesitamos realizar un programa que solicite números y los sume hasta que el usuario ingrese un número negativo, como no se sabe en que momento el usuario ingresará un valor negativo, la estructura recomendable es el While (Mientras). La característica principal del While es que este primero pregunta y después hace.

```
1  Proceso elwhile
2      escribir "Ingrese un número"
3      leer numero
4      c<-1
5      Mientras c<=numero Hacer
6          Imprimir "Vamos en el giro ", c
7          c<-c+1
8      Fin Mientras
9  FinProceso
```

Centinela que inicia en 1

Pregunta que controla el bucle

Incrementamos el centinela para permitir el fin del bucle

La Estructura Repetitiva Para (For)

Es una estructura repetitiva que se emplea cuando se conoce cuantos giros debe realizar el ciclo, por ejemplo, si se realiza un algoritmo que le solicite al usuario cuantos números va a sumar, el algoritmo conocería la cantidad de giros a partir de la cantidad de números ingresados por el usuario.

```
1  Proceso elfor
2      escribir "Ingrese un número"
3      leer numero
4      Para c<-1 Hasta numero Con Paso 1 Hacer
5          Imprimir "Vamos en el giro ", c
6      Fin Para
7  FinProceso
```

Incremento que se asigna al centinela

Inicializa el sistema

Límite del ciclo

PALABRAS RESERVADAS

Son palabras propias en la sintaxis de un lenguaje de programación y no puede ser usada como el nombre de una variable.

En PSeInt son las siguientes:

- | | |
|-------------|------------------|
|) Proceso |) Paso |
|) Inproceso |) Hacer |
|) Definir |) Mientras |
|) Como |) FinSi |
|) Logico |) Finpara |
|) Entero |) FinMientras |
|) Real |) Subproceso |
|) Caracter |) Repetir |
|) Si |) Escribir |
|) Entonces |) Leer |
|) Sino |) Dimension |
|) Para |) Borra Pantalla |

PSeInt: Comando Escribir

Escribir: Se utiliza para mostrar un texto en el programa.



Tipos de Escribir:

-) Escribir: Es normal, pero al terminar el texto, enviará a la línea de abajo.
-) Escribir Sin Saltar: Es utilizado para mostrar un texto sin enviar a la línea de abajo.

Desarrollo:

-) Si es texto (*no variable asignada*) debe estar encerrado entre comillas ya sean simples o dobles (*Propiedades de los datos*), ejemplo:

'Texto'

-) El texto a mostrar debe estar después del comando 'Escribir', ejemplo:

Escribir 'Texto'

Ejemplo en programa:

```
Escribir 'Hola'
Escribir Sin Saltar '¿Como '
Escribir 'estas?'
```

PSeInt: Comando Para

Para: Trabaja de manera similar al 'Repetir', solo que, aquí asignamos las variables dentro del comando.

Desarrollo:

-) Debe existir previamente un valor de llegada
-) Para comenzar el comando se debe escribir 'Para' seguido de un variable a la cual ahí mismo se le asigna un valor inicial, ejemplo:



Para variableInicio <- 1

-) Luego se debe escribir 'Hasta' y el valor de llegada, este valor debe ser colocado predeterminado o proveniente de una variable anterior, ejemplo:
-) Para variableInicio <- 1 Hasta 10
-) variableLlegada = 10 Para variableInicio <- 1 Hasta variableLlegada
-) Seguidamente hay que escribir 'Con Paso' e indicar el numero de pasos que dará por cada repetición, es decir, cuanto se le sumará al valor, y terminar con un 'Hacer' ejemplo:

Para variableInicio <- 1 Hasta 10 Con Paso 1 Hacer

-) Después de ello, se debe colocar la secuencia de acciones y al final terminar el comando con un 'Fin Para', ejemplo:

Para variableInicio <- 1 Hasta 10 Con Paso 1 Hacer

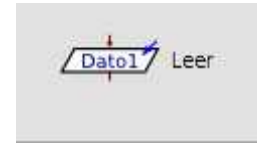
acciones

Fin Para

```
Escribir 'Indique su edad'
Escribir 'Te daré aplausos por cada 4 años que tenga tu edad'
Leer respuestaUsuario
variableLlegada <- trunc(respuestaUsuario / 4)
Para variableRepetir<-1 Hasta variableLlegada Con Paso 1 Hacer
    Escribir '*Aplauso*'
Fin Para
```

PSeInt: Comando Leer

Leer: Permite al usuario asignar el valor a una variable, ya sea Numérico o Carácter.



Desarrollo:

-) Debes indicar el nombre de una variable
-) Esta puede ser usada en 'Escribir' siempre y cuando no utilice comillas de ningún tipo
-) La acción 'Leer' debe estar previa a la variable, ejemplo:

Leer nombreVariable

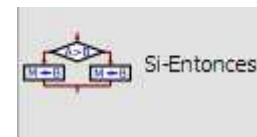
Ejemplo en programa:

```
Escribir '¿Como te llamas?'
Leer nombrePersona
Escribir Sin Saltar 'Hola '
Escribir nombrePersona

Escribir '¿Como te llamas?'
Leer nombrePersona
Escribir Sin Saltar 'Hola '
Escribir 'nombrePersona'
```

PSeInt: Comando Si-Entonces

Si-Entonces: Este comando trabaja como condiciones, es decir, 'Si' condición previa se cumple 'Entonces' se hará una acción, esto trabaja como condición de valor de verdad (verdadero 'v' o falso 'f'), si la condición se cumple (o sea es 'v') entonces se hace la acción en caso contrario es 'f'



Desarrollo:

-) Se debe escribir la palabra ' Si ' seguida de la condición tras este la palabra 'Entonces', ejemplo:

Si condición Entonces

-) Cuando la condición se cumple hará las acciones asignadas, al finalizar todas las acciones que se deberán hacer en este 'Si-Entonces', se debe terminar con un 'FinSi', ejemplo:

Si condición Entonces

Acciones

FinSi

-) En caso de querer dar una acción si la condición no se cumple, antes del 'FinSi' debe existir un 'Sino' seguido de sus respectivas acciones, ejemplo:

Si condición Entonces

Acciones si la condición se cumple

Sino

Acciones si la condición no se cumple

FinSi

Ejemplo en programa:

```
Escribir 'Hola'
nombreAdivinar <- 'Jose'
Escribir '¿te llamas ', nombreAdivinar, '? (si o no) '
Leer respuestaUsuario
Si respuestaUsuario = 'si' Entonces
    Escribir 'Que bien! He adivinado'
Sino
    Escribir 'Ah! Que mal, he fallado'
FinSi
```


PSeInt: Comando Según

Según: Permite trabajar con un sistema de valor numero, donde, según el valor elegido ocurría una secuencia de acciones.



Desarrollo:

-) Debe existir previamente la variable con la que se trabajará, ya sea por el comando 'Leer', o por 'Asignar'
-) La variable solo debe ser de valor numérico, no se permite del tipo texto
-) Para comenzar el comando, este inicia con la palabra 'Segun' seguido de la variable numérica previamente existente y seguido de la palabra 'Hacer', ejemplo:

Segun variableNumerica Hacer

-) Tras tener el comienzo del comando, prosigue una secuencia de acciones que trabajan con opciones/condiciones numéricas que las llaman, ejemplo:

1:

acciones

-) Las opciones/condiciones numéricas deben ser valores numéricos (1 , 2 , 3 , etc...), estas terminar con un ':' (dos puntos) y seguido de las acciones hacer si esta opción es elegida

1:

acciones

2:

acciones

3:

acciones

-) Las opciones/condiciones numéricas pueden tener hasta dos valores, unidos con un ',' entre ellos, ejemplo:

1:

acciones

2 , 3:

acciones

-) Tras finalizar las opciones/condiciones numéricas y sus respectivas acciones, se debe finalizar el comando con un 'Fin Segun', ejemplo:

Segun variableNumerica Entonces

opciones y acciones

Fin Segun

```
Escribir 'Seleccione:'
Escribir '(1) Saludo'
Escribir '(2,3) Nada'
Escribir '(4) Prueba'
Escribir '(5) Cerrar'
Escribir Sin Saltar 'Escriba el numero:'
Leer respuestaUsuario
Segun respuestaUsuario Hacer
    1:
        Escribir 'Hola, ¿como estas?'
        Leer saludoUsuario
        Escribir 'Presiona cualquier tecla para cerrar'
    2 , 3:
        Escribir 'No puedo nadar'
        Escribir 'Presiona cualquier tecla para cerrar'
    4:
        Escribir 'Esto es una prueba'
        Escribir 'Presiona cualquier tecla para cerrar'
De Otro Modo:
    Escribir 'Presiona cualquier tecla para cerrar'
Fin Segun
```

PSeInt: Comando Mientras

Mientras: Permite crear un ciclo, que terminará cuando la condición no se cumpla

Desarrollo:

-) Debe existir previamente una variable que será utilizada en la condición
-) Para utilizar el comando, primero se debe escribir 'Mientras' seguido de la condición y luego la palabra 'Hacer', ejemplo:

Mientras condición Hacer

-) Luego del 'Hacer', debe seguir la secuencia de acciones mientras la condición se cumpla, y al final de estas cerrar el comando con 'Fin Mientras', ejemplo:

Mientras condición Hacer

acciones

Fin Mientras



Ninguna acción después del 'Fin Mientras' será ejecutada por el programa, hasta que el Mientras no se termine, es decir:

Mientras condición Hacer

acciones

Fin Mientras

acciones que esperaran que termine el mientras

Ejemplo en programa:

```

variableCondicion <- 0
Mientras variableCondicion != 18 Hacer
    Escribir 'Cuanto es:'
    Escribir '2 + 4 x 5 - 12 / 3'
    Leer variableCondicion
    Si variableCondicion != 18 Entonces
        Escribir 'Te has equivocado, vuelve a intentarlo'
    FinSi
Fin Mientras
Escribir 'Felicidades has acertado!'

```

PSeInt: Comando Repetir

Repetir: Permite hacer una secuencia de acciones hasta completar un condición

Desarrollo:

-) Debe existir previamente una variable con el valor de inicio
-) Debe existir un valor de llegada, ya sea mediante variable o predeterminado
-) Para empezar el comando, se necesita escribir 'Repetir' luego colocar la secuencia de acciones y para finalizar colocar 'Hasta Que' y la condición para finalizar la repetición, ejemplo:



Repetir

acciones

Hasta Que condición

Ejemplo en programa:

```

Escribir 'Coloca tu edad y por cada 4 años que tengas te daré un aplauso'
Leer respuestaUsuario
variableRepetir < 0
variableLlegar <- trunc(respuestaUsuario/4)
Si variableLlegar >= 1 Entonces
    Repetir
        Escribir '*Aplauso*'
        variableRepetir < variableRepetir + 1
    Hasta Que variableRepetir >= variableLlegar
FinSi

```

