Federico Agra

L.U.:94186

Proyecto Lenguaje Simplesem

Proyecto – Tema II

Lenguajes de Programación - 2018

Consignas de Estudio

Ejercicio a

Sentencia *for* de Java

for(int var = <valor\_inicial>;<condición\_corte>;<factor\_incremento>)

{

//cuerpo del for

}

En Java, se itera desde un *valor inicial* mientras se cumpla la *condición de corte*, y en cada iteración se incrementa el *valor inicial* según el *factor de incremento.* Si la *condición de corte* no se satisface en el momento en el que se ejecuta la sentencia *for*, nunca se itera dentro de ella.

Sentencia *for* de Pascal

for <valor\_inicial> to <valor\_final>

begin

//cuerpo del for

end

En Pascal, se repite la sentencia (o bloque) dentro del *begin/end* del *for* tantas veces como enteros comprendidos entre el *valor inicial* y *el valor final* (se incrementa en cada iteración el *valor inicial* en una unidad hasta que sea igual al *valor final*).

Ejercicio b

Traduccion en Simplesem de una sentencia *for* de Java

SetD Actual+Δi, 0

ini JumpT end, !(condición) %Salto a end si condición = false

%%%Cuerpo del for;

SetD Actual+Δi, D[Actual+Δi]+F %i=i+F

Jump ini %Salto al inicio del loop

end %%%fuera del for;

Ejercicio c

En la sentencia for de Java no se pueden determinar de antemano la cantidad de iteraciones ya que la condición de corte es una expresión booleana (del tipo while), la cual se evalúa en cada iteración y puede satisfacerse indefinidamente.

Otro punto que impide esto es el hecho de que Java permite modificar el valor de la variable de control dentro del cuerpo del for. Esto hace que puedan aumentar o disminuir la cantidad inicial de iteraciones.

Para poder determinar la cantidad de iteraciones de antemano se debe eliminar la condición de corte, determinando las iteraciones por los valores inicio y fin, y almacenando este cálculo en un almacenamiento auxiliar para que no pueda ser modificado desde el cuerpo de la sentencia.

for(int i = <valor\_inicial>; i..<valor\_final>)

{

//cuerpo del for

}

Se iterará tantas veces como enteros comprendidos entre i y n (como el for de Pascal). El valor i se incrementa una unidad en cada iteración, aunque se podría permitir determinar el factor de incremento para una mayor flexibilidad.

Ejercicio d

Sentencia Continue de While

while(<condicion>)

{

//cuerpo del while antes de continue

continue;

//cuerpo del while despues de continue

}

//código fuera del while

Traduccion en Simplesem de una sentencia *for* de Java

ini JumpT end, !(condición) %Salto a end si condición = false

%%% cuerpo del while antes de continue;

Jump ini %continue

%%% cuerpo del while antes de continue

Jump ini %fin while. Salto al inicio del while

end %%%fuera del while;

Ejercicio e

INSTs

|  |  |
| --- | --- |
| Clase Padre  VT\_Padre  V1  V2  V3 | Clase Hijo2  VT\_Hijo2  V1  V2  V3  V4 |
| Clase Hijo3  VT\_Hijo3  V1  V2  V3  V4  V5 | Clase Hijo4  VT\_Hijo4  V1  V2  V3  V4  V5  V6 |

VTs

|  |  |
| --- | --- |
| Clase Padre  init()(Padre)  superSeter(p1,p2,p3)(Padre)  m3()(Padre)  m4()(Padre)  m5()(Padre) | Clase Hijo2  init()(Hijo2)  superSeter(p1,p2,p3)(Hijo2)  m3()(Padre)  m4()(Hijo2)  m5()(Padre)  m6()(Hijo2)  m7()(Hijo2) |
| Clase Hijo3  init()(Hijo3)  superSeter(p1,p2,p3)(Hijo3)  m3()(Padre)  m4()(Hijo3)  m5()(Padre)  m6()(Hijo2)  m7()(Hijo2) | Clase Hijo4  init()(Hijo4)  superSeter(p1,p2,p3)(Hijo4)  m3()(Padre)  m4()(Hijo4)  m5()(Padre)  m6()(Hijo4)  m7()(Hijo2) |

RAs

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Clase Padre | | | | | | | | | |
| init:  -PR  -ED  -THIS | superSeter:  -PR  -ED  -THIS  -p1  -p2  -p3 | | | m3:  -PR  -ED  -THIS | | m4:  -PR  -ED  -THIS  -x  -w | | | m5:  -PR  -ED  -THIS  -x  -r  -i |
| Clase Hijo2 | | | | | | | | | |
| init:  -PR  -ED  -THIS | superSeter:  -PR  -ED  -THIS  -p1  -p2  -p3  -p4 | | | m4:  -PR  -ED  -THIS  -t1  -t2  -x  -w | | m6:  -PR  -ED  -THIS | | | m7:  -PR  -ED  -THIS  -p |
| Clase Hijo3 | | | | | | | | | |
| init:  -PR  -ED  -THIS | | | superSeter:  -PR  -ED  -THIS  -p1  -p2  -p3  -p4  -p5 | | | | m4:  -PR  -ED  -THIS  -v  -t2  -t1  -x  -w | | |
| Clase Hijo4 | | | | | | | | | |
| init:  -PR  -ED  -THIS | | superSeter:  -PR  -ED  -THIS  -p1  -p2  -p3  -p4  -p5  -p6 | | | m4:  -PR  -ED  -THIS  -x  -t  -i  -t1 | | | m6:  -PR  -ED  -THIS  -t  -r | |
| Clase Lenguajes2018Tema2 | | | | | | | | | |
| init:  -PR  -ED  -v1 | | | | | computar:  -PR  -ED  -p  -x  -r | | | | |