

Ejercicios de Programación - Sebesta

Lenguajes de Programación - ESPOL

29 de enero de 2014

1. Introducción

Las respuestas propuestas en este repositorio son producto del trabajo de los estudiantes de la materia “Lenguajes de Programación” de la ESPOL, correspondientes a las preguntas del libro de Robert Sebesta, Concepts of Programming Languages.

2. Preguntas y Respuestas

2.1. Capítulo 5: Nombres, Enlaces y Alcances

2.1.1. **Pregunta 4: Escriba un script en Python con subprogramas triplemente anidados, donde cada subprograma referencie a variables que han sido definidas en otros niveles de la jerarquía**

Escriba su respuesta con claridad. En las secciones de código utilice listings. Capítulo 5 pregunta 5 En caso de querer llamar en nuestro programa números pseudoaleatorios, que han sido generados por un subprograma se necesita del history-sensitivity para almacenar la última iteración de la variable localmente. Ejemplo de un subprograma generando numeros pseudoaleatorios:

```
public void generarSerieDeAleatorios () {  
  
    Random numAleatorio;  
    numAleatorio = new Random ();  
    for (int i=0; i < serieAleatoria.size(); i++) {  
        serieAleatoria.set(i, numAleatorio.nextInt(1000) );  
    }  
}
```

2.2. Capítulo 8: Statement-Level Control Structures

■ Pregunta 4

Pros y contras de usar palabras únicas de cierre en sentencias compuestas.

Pros:

Legibilidad: Cuando vemos un `endif` o `endwhile` en un programa escrito por alguien más, es claro ver que bloque finaliza.

Usando llaves es menos claro sobre todo cuando el bloque cubre muchas líneas de código y hay bastantes laves.

Llaves son mucho más difícil de leer, en particular para saber donde el bloque termina.

Ej:

```
if ( <expression1> )
    if ( <expression2> )
        <stmt>
else
    <stmt2>
```

A cuál `if` pertenece `else`? Con el uso de `endif`, no hay inconveniente. Por ejemplo:

```
if (<expression1>) then
    if (<expression2>)
        <stmt1>
    endif
else
    <stmt2>
```

```
if (<expression1>) then
    if (<expression2>)
        <stmt1>
else
    <stmt2>
endif
```

Cons: Capacidad de escritura: El uso de palabras reservadas significa que hay más palabras reservadas; por lo tanto menos palabras disponibles para el programador.

Esto también quiere decir que hay más palabras por aprender; caso contrario nos limitaríamos a unas cuantas palabras reservadas.