



aprenderaprogramar.com

Verificación mental y verificación (prueba) por seguimiento escrito de algoritmos. (CU00230A)

Sección: Cursos

Categoría: Curso Bases de la programación Nivel II

Fecha revisión: 2024

Autor: Mario R. Rancel

Resumen: Entrega nº 29 del Curso Bases de la programación Nivel II

24

VERIFICACIÓN MENTAL DE ALGORITMOS

La verificación mental consiste en “recorrer” el algoritmo mentalmente. Durante ese recorrido hacemos el seguimiento de:

- El flujo del programa.
- Los valores de las variables.
- Procesos relevantes que hayan tenido lugar como extracción o almacenamiento de datos en dispositivos externos.

Para el caso planteado de doble anidamiento, podemos realizar una verificación mental sin demasiado problema puesto que tenemos tres variables y dos bucles. Sin embargo, para algoritmos complejos nuestra capacidad mental no va a ser suficiente por lo que esta técnica será útil sólo para evaluar procesos sencillos.

VERIFICACIÓN POR SEGUIMIENTO ESCRITO

La verificación por seguimiento escrito vendría a ser plasmar el seguimiento mental que hacemos sobre papel. Tiene la ventaja de que requiere un mínimo empleo de nuestra memoria, ya que de todo va quedando registro escrito, así como de evitarnos el tener que empezar desde el principio si nos perdemos, ya que siempre podemos volver a empezar desde el último punto escrito del que estamos seguros.

Para el caso planteado el seguimiento escrito vendría a reflejar lo siguiente:

Entrada a bucle controlado por i , que tomará valores de 1 a 5.

i vale 1.

Entra a bucle controlado por j , que tomará valores de 1 a 5.

j vale 1. A vale $i * j = 1 * 1 = 1$. Se muestra A . j progresa a 2.

A vale $i * j = 1 * 2$. Se muestra 2. j progresa a 3.

A vale $i * j = 1 * 3 = 3$. Se muestra 3. j progresa a 4.

A vale $i * j = 1 * 4 = 4$. Se muestra 4. j progresa a 5.

A vale $i * j = 1 * 5 = 5$. Se muestra 5. j progresa a 6 y supera el valor preestablecido, por lo que se sale del bucle.

i progresa a 2. Entrada a bucle controlado por j , que tomará valores de 1 a 5. j vale 1. A vale $i * j = 2 * 1 = 2$. Se muestra 2. j progresa a 2.

A vale $i * j = 2 * 2 = 4$. Se muestra 4. j progresa a 3.

A vale $i * j = 2 * 3 = 6$. Se muestra 6. j progresa a 4.

A vale $i * j = 2 * 4 = 8$. Se muestra 8. j progresa a 5.

A vale $i * j = 2 * 5 = 10$. Se muestra 10. j progresa a 6 y supera el valor preestablecido, por lo que se sale del bucle. i progresa a 3. Entrada a bucle controlado por j , que tomará valores de 1 a 5. j vale 1.

A vale $i * j = 3 * 1 = 3$. Se muestra 3. j progresa a 2.

A vale $i * j = 3 * 2 = 6$. Se muestra 6. j progresa a 3.

A vale $i * j = 3 * 3 = 9$. Se muestra 9. j progresa a 4.

A vale $i * j = 3 * 4 = 12$. Se muestra 12. j progresa a 5.

A vale $i * j = 3 * 5 = 15$. Se muestra 15. j progresa a 6 y supera el valor preestablecido, por lo que se sale del bucle. i progresa a 4. Entrada a bucle controlado por j , que tomará valores de 1 a 5. j vale 1.

A vale $i * j = 4 * 1 = 4$. Se muestra 4. j progresa a 2.

A vale $i * j = 4 * 2 = 8$. Se muestra 8. j progresa a 3.

A vale $i * j = 4 * 3 = 12$. Se muestra 12. j progresa a 4.

A vale $i * j = 4 * 4 = 16$. Se muestra 16. j progresa a 5. A vale $i * j = 4 * 5 = 20$. Se muestra 20. j progresa a 6 y supera el valor preestablecido, por lo que se sale del bucle. i progresa a 5. Entrada a bucle controlado por j , que tomará valores de 1 a 5. j vale 1.

A vale $i * j = 5 * 1 = 5$. Se muestra 5. j progresa a 2.

A vale $i * j = 5 * 2 = 10$. Se muestra 10. j progresa a 3.

A vale $i * j = 5 * 3 = 15$. Se muestra 15. j progresa a 4.

A vale $i * j = 5 * 4 = 20$. Se muestra 20. j progresa a 5.

A vale $i * j = 5 * 5 = 25$. Se muestra 25. j progresa a 6 y supera el valor preestablecido, por lo que se sale del bucle. i progresa a 6 y supera el valor preestablecido, por lo que se sale del bucle.

El lenguaje que se emplea en este seguimiento escrito es libre y puede “soslayar” algunos pasos que se consideren no relevantes. Puede emplearse un lenguaje abreviado que evite la repetición de texto. Por ejemplo:

(-->) Bucle 01 $i \rightarrow 1$ a 5. $i = 1$. (-->) Bucle 02 $j \rightarrow 1$ a 5. $j = 1$. $A = 1$. $j = 2$. $A = 2$. $j = 3$. $A = 3$. $j = 4$. $A = 4$. $j = 5$. $A = 5$. $j = 6$. Fin bucle 02 (<-->) $i = 2$. (-->) Bucle 03. $j \rightarrow 1$ a 5. $j = 1$. $A = 2$. $j = 2$. $A = 4$. $j = 3$. $A = 6$. $j = 4$. $A = 8$. $j = 5$. $A = 10$. $j = 6$. Fin bucle 03 (<-->) $i = 3$. (-->) Bucle 04. $j \rightarrow 1$ a 5. $j = 1$. $A = 3$. $j = 2$. $A = 6$. $j = 3$. $A = 9$. $j = 4$. $A = 12$. $j = 5$. $A = 15$. $j = 6$. Fin bucle 04 (<-->) $i = 4$. (-->) Bucle 05 $j \rightarrow 1$ a 5. $j = 1$. $A = 4$. $j = 2$. $A = 8$. $j = 3$. $A = 12$. $j = 4$. $A = 16$. $j = 5$. $A = 20$. $j = 6$. Fin bucle 05 (<-->) $i = 5$. (-->) Bucle 06 $j \rightarrow 1$ a 5. $j = 1$. $A = 5$. $j = 2$. $A = 10$. $j = 3$. $A = 15$. $j = 4$. $A = 20$. $j = 5$. $A = 25$. $j = 6$. Fin bucle 06 (<-->) Fin bucle 01 (<-->)

Abreviando o no esta técnica resulta repetitiva, de lectura no demasiado agradable y un tanto tediosa. Su uso se limita por tanto a alguna aplicación de fines didácticos o a procesos poco largos en los que mentalmente “nos enrollamos” y decidimos escribir un pequeño seguimiento para verificar que se cumple lo esperado.

Próxima entrega: CU00231A

Acceso al curso completo en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente: http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=36&Itemid=60