



aprenderaprogramar.com

Condicionantes y métodos en la resolución de un problema de programación II. (CU00110A)

Sección: Cursos

Categoría: Curso Bases de la programación Nivel I

Fecha revisión: 2024

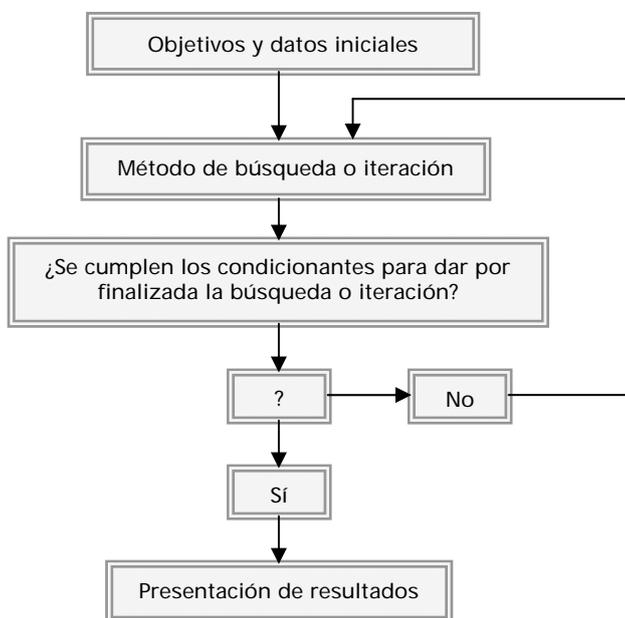
Autor: Mario R. Rancel

Resumen: Entrega nº9 del Curso Bases de la programación Nivel I

24

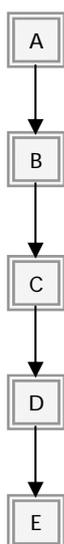
Problema con iteración para búsqueda de soluciones

Problemas con tipología diversa en cuanto a métodos de resolución, complejidad, etc., pero que tienen en común adoptar esquemas reiterativos de forma que es la repetición *n* veces de un método de búsqueda o iteración la que da lugar a unos resultados. El esquema sería del tipo:

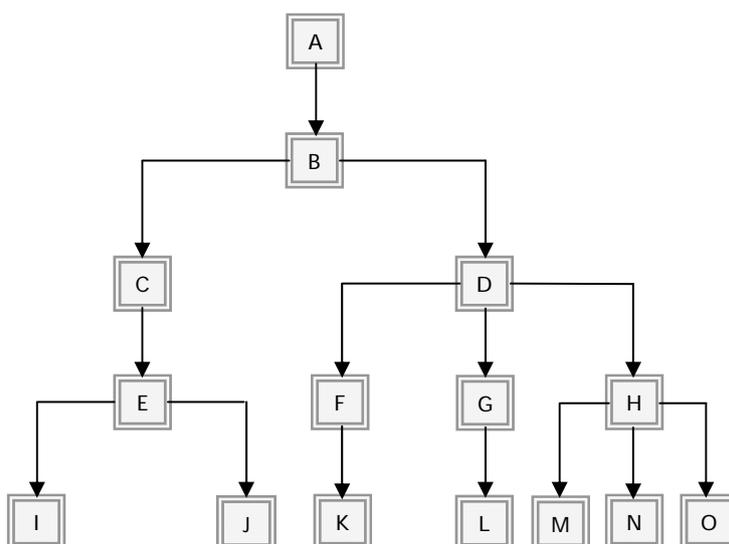


Respecto a este gráfico cabe llamar la atención sobre la existencia de un "proceso circular" o bucle, introduciendo un nuevo tipo de proceso respecto a lo que veníamos viendo.

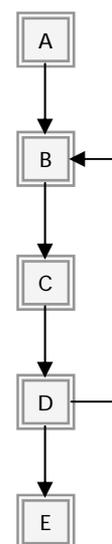
Gráficamente podemos distinguir:



1. Esquema



2. Esquema lineal con bifurcaciones

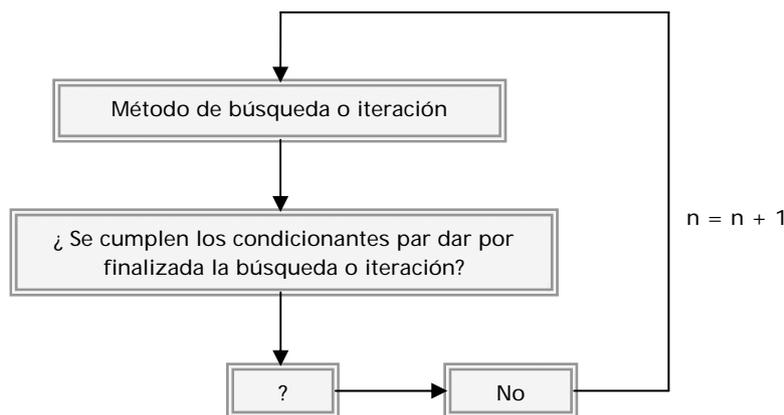


3. Esquema con proceso circular

Los procesos circulares resultan de gran utilidad no sólo para construir los "esquemas" de un programa, sino también como parte de los procesos internos del mismo. (p. ej. un programa con esquema lineal

puro podrá tener procesos circulares. En este caso, en vez de entre dos pasos B – C, estarán "internos" en uno o varios de los pasos A, B, C, D ó E). Por su utilidad e importancia continuaremos hablando de estos procesos durante todo el curso. Ahora plantearemos únicamente algunas cuestiones preliminares de interés.

Volvamos al esquema tipo y supongamos que el programa ha entrado ya en el bucle iterativo. El ordenador está repitiendo el proceso:

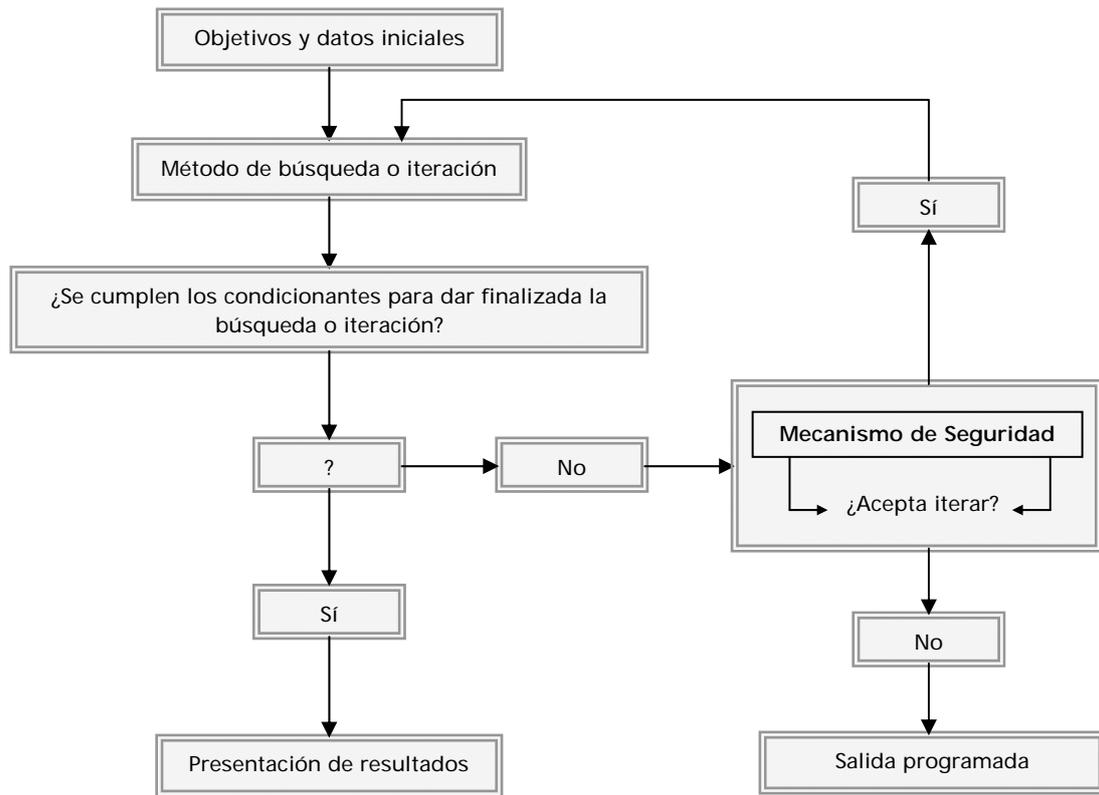


El proceso se va a repetir n veces. Consideremos los posibles valores de n .

- a) **$n = 0$** : no se llega a entrar en el bucle porque la solución fue inmediata. Esto será totalmente inusual, ya que el uso de métodos iterativos deriva de la imposibilidad de obtener soluciones inmediatas.
- b) **n es un número pequeño¹**: se alcanza una solución rápidamente.
- c) **n es un número grande¹**: se alcanza una solución con cierta demora.
- d) **n tiende a infinito**: no se alcanza una solución con tiempos de espera razonables, o simplemente, no se alcanza una solución.
- e) **n vale infinito**: se ha entrado en un proceso circular puro, en el que es imposible que se lleguen a cumplir los condicionantes para dar por finalizada la búsqueda o iteración. El proceso se repetirá continuamente sin obtener solución. Esta es una situación de "ordenador colgado" o "bloqueado". Situación obviamente indeseable y contra la que un programador debe luchar.

Descrito ya en qué consiste el bucle infinito veamos el procedimiento que usaremos para evitar que se produzca. Se basará en introducir un mecanismo de seguridad o proceso de evaluación para determinar si se debe repetir el bucle o bien desviar el flujo hacia una salida programada. El esquema quedaría de la siguiente forma:

¹ El llamar a un número de iteraciones "pequeño" o "grande" depende de la complejidad del proceso y tiempo que se invierte en cada iteración, de la potencia del procesador, etc. No hay por tanto valores de referencia, aunque consideramos que a partir de 10^4 iteraciones conviene "tener cuidado".



Es pues el mecanismo de seguridad el que decide si repetir o no una nueva iteración. Los criterios para pasar a una salida programada pueden ser uno o varios, debiendo decidir nosotros cuántos usamos y el grado de restricción para cada uno de ellos. Ejemplo por analogía con un motor:

Caso 1. Único Criterio.

Criterio de Parada	Grado de Restricción
Sobrecalentamiento	$T_{max} \geq 70 \text{ }^\circ\text{C}$

Caso 2. Dos Criterios.

Criterio de Parada	Grado de Restricción
Sobrecalentamiento	$T_{max} \geq 50 \text{ }^\circ\text{C}$
Caída de Tensión	$\Delta X > 6\% X_n$

Caso 3. Múltiples criterios: sobrecalentamiento, caída de tensión, vibración, ruido, revoluciones, etc.

En programación será habitual usar algunos de estos criterios para dar paso a una salida programada:

1. Se detectan datos de entrada inductores de bucle infinito.
2. El número de iteraciones supera el valor que se ha fijado como aceptable.
3. El tiempo empleado en iteraciones supera el valor que se ha fijado como aceptable.
4. Se detecta que la solución está tendiendo a más o menos infinito.
5. Solución se sale del rango previsto o deseado.

A su vez, la salida programada podría consistir en diferentes posibilidades:

1. Mensaje de advertencia y fin de programa.
2. Mensaje de advertencia y solicitar nuevos datos iniciales.
3. Mensaje de advertencia y mostrar resultados con salvedades.
4. Modificación automática de vectores de búsqueda y solicitar autorización para continuar.

Mostrar resultados con salvedades consistirá en dar el valor al que se ha llegado, indicando que no se cumplen los criterios de aceptación.

La modificación de valores de búsqueda hace referencia a cambios en los parámetros del método, por ejemplo agrandar un vector direccional que nos está acercando a la solución. Solicitar autorización deriva de la necesidad de limitar al ordenador a nuestro criterio cuando los tiempos de búsqueda se hacen muy grandes.

Trataremos de repasar lo visto con algunos ejemplos y ejercicios.

Próxima entrega: CU00111A

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=28&Itemid=59