

La recursión es un proceso que permite expresar soluciones simples y naturales a ciertos problemas, es importante considerar que no todos los problemas son naturalmente recursivos.

un objeto recursivo es aquel que aparece en la definición de sí mismo, así como el que se llama a sí mismo. Los árboles, por ejemplo, representan estructuras de datos, no lineales y dinámicas, más eficientes que existen actualmente en computación. la característica de los árboles que son estructuras inherentemente recursiva. es decir cualquier actividad de programación que realice con árboles se utiliza recursividad.

la recursión puede presentar de dos maneras diferentes:

- Directa. el programa o subprograma se llama directamente a sí mismo.
- Indirecta: el subprograma llama otro subprograma, y éste, en algún momento, llama nuevamente al primero.

en toda definición recursiva de un problema siempre se deben establecer dos pasos muy importantes; el paso básico y el paso recursivo. El primero, uno o varios, dependiendo del problema, se utiliza como condición de parada o fin de la recursividad. A éste llegamos cuando encontramos la solución del problema o cuando decimos que ya no vamos a seguir, porque no están dadas las condiciones para hacerlo. El paso segundo, por otra parte, propicia la recursividad. Se pueden presentar uno o varios, nuevamente dependiendo del problema a resolver.

Cuando se analiza la solución recursiva de un problema es importante determinar con precisión cuáles serán los pasos básico y recursivo. en cada vuelta del ciclo es importante que nos acerquemos cada vez más a la solución del problema, o sea, al paso básico. sí esto no ocurre entonces podemos estar ante un ciclo extraño. Es decir, el problema estaría mal definido y, en tal caso, la máquina se quedaría ejecutando por tiempo indefinido el programa en que estuviera, y sólo terminaría al agotarse la memoria.

Osvaldo Cairó/Silvia Guardati. (206). Estructura de datos tercera edición. México: McGraw-Hill