Unidad III

Estructuras no lineales

3.1. Recursividad.

Recurrencia, recursión o recursividad es la forma en la cual se especifica un proceso basado en su propia definición. Siendo un poco más precisos, y para evitar el aparente círculo sin fin en esta definición:

Un problema que pueda ser definido en función de su tamaño, sea este N, pueda ser dividido en instancias más pequeñas (< N) del mismo problema y se conozca la solución explícita a las instancias más simples, lo que se conoce como casos base, se puede aplicar inducción sobre las llamadas más pequeñas y suponer que estas quedan resueltas.

Para que se entienda mejor a continuación se exponen algunos ejemplos:

- Factorial: Se desea calcular $n!$ (el factorial de n, que se define como el producto de todos los enteros positivos de 1 a n). Se puede definir el problema de forma recurrente como $n(n-1)!$; como $(n-1)!$ es menor que $n!$ podemos aplicar inducción por lo que disponemos del resultado. El caso base es $0!$ que es 1.
- Algoritmo de ordenación por fusión: Sea v un vector de *n* elementos, podemos separar el vector en dos mitades. Estas dos mitades tienen tamaño *n*/2 por lo que por inducción podemos aplicar la ordenación en estos dos subproblemas. Una vez tenemos ambas mitades ordenadas simplemente debemos fusionarlas. El caso base es ordenar un vector de cero o un elemento, que está trivialmente ordenado y no hay que hacer nada.

En estos ejemplos podemos observar como un problema se divide en varias (una o más) instancias del mismo problema, pero de tamaño menor gracias a lo cual se puede aplicar inducción, llegando a un punto donde se conoce el resultado (el caso base).

Nota: aunque los términos "recursión" y "recursividad" son ampliamente empleados en el campo de la informática, el término correcto en castellano esrecurrencia. Sin embargo este último término es algo más específico.
Véase relación de recurrencia

3.2. Árboles.

- Un árbol es una estructura no lineal en la que cada nodo puede apuntar a uno o varios nodos.
- También se suele dar una definición recursiva: un árbol es una estructura en compuesta por un dato y varios árboles.

Esto son definiciones simples. Pero las características que implican no lo son tanto.

3.3. Grafos.

Desafortunadamente no existe una terminología estandarizada en la teoría de los grafos, por lo tanto es oportuno aclarar que las presentes definiciones pueden variar ligeramente entre diferentes publicaciones de estructura de datos y de teoría de grafos, pero en general se puede decir que un grafo como indica su nombre lo indica es la representación (para nuestro caso) gráfica de los datos de una situación particular, ejemplo:

Los datos contienen, en algunos casos, relaciones entre ellos que no es necesariamente jerárquica. Por ejemplo, supongamos que unas líneas aéreas realizan vuelos entre las ciudades conectadas por líneas como se ve en la figura anterior (más adelante se presentaran grafos con estructuras de datos); la estructura de datos que refleja esta relación recibe el nombre de grafo.

Se suelen usar muchos nombres al referirnos a los elementos de una estructura de datos. Algunos de ellos son "elemento", "ítem", "asociación de ítems", "registro", "nodo" y "objeto". El nombre que se utiliza depende del tipo de estructura, el contexto en que usamos esa estructura y quien la utiliza.

En la mayoría de los textos de estructura de datos se utiliza el termino "registro" al hacer referencia a archivos y "nodo" cuando se usan listas enlazadas, árboles y grafos.

También un grafo es una terna G = (V,A,j), en donde V y A son conjuntos finitos, y j es una aplicación que hace corresponder a cada elemento de A un par de elementos de V. Los elementos de V y de A se llaman, respectivamente, "vértices" y "aristas" de G, y j asocia entonces a cada arista con sus dos vértices.