

Complejidad algorítmica.

La complejidad algorítmica es una métrica teórica que se aplica a los algoritmos en este sentido.

El objetivo de la complejidad algorítmica es estudiar de forma genérica (e independiente a la máquina) los recursos (tiempo y cantidad de memoria) requeridos por un algoritmo para resolver un problema.

Principios:

- Invarianza: Dado un algoritmo S, y dos implementaciones I1 e I2 de dicho algoritmo, cuyos tiempos de ejecución son $t_1(n)$ y $t_2(n)$, entonces existen constantes naturales k y n_0 , tales que $\forall n \geq n_0, t_1(n) \leq k \cdot t_2$
- Operación elemental: Una operación elemental es aquella cuyo tiempo de ejecución está acotado superiormente por un valor constante que depende solo de la máquina donde se ejecuta y es por tanto independiente de los parámetros del problema que resuelve un algoritmo.

Tiempo de ejecución y tamaño de la entrada.

Se desea tener una medida de la duración del tiempo de ejecución de un algoritmo en función del tamaño de la entrada.

A través de llamados al sistema operativo se puede conocer el valor del reloj de tiempo real.

Invocando al reloj, antes y después de realizar el algoritmo se tendrá una medida de la duración del tiempo de ejecución. Sin embargo esta medida es muy dependiente del hardware (memoria, reloj, procesador), del sistema operativo (multitarea, multiusuario) y puede variar significativamente dependiendo del computador, del compilador, y de la carga del sistema. Al disponer de sistemas con multiprocesadores o que la ejecución sea distribuida también afecta medir el tiempo con cronómetro.

Por la razón anterior como una medida del tiempo de ejecución, se considera contar las instrucciones del lenguaje de alto nivel que son necesarias realizar.

El tamaño de la entrada debe ser precisado con más detalle. Podría ser el número de bits que miden la información que el algoritmo procesa, pero en forma tradicional se considera el número de elementos o componentes básicas que son sometidos al proceso.

Por ejemplo si tenemos un arreglo de n componentes, y el algoritmo tiene por objetivo, sumar los valores de las componentes, o bien ordenar las componentes, se suele decir que n es el tamaño de la entrada. Independientemente si el arreglo es de enteros, o de estructuras.

Complejidad temporal.

Se denomina complejidad temporal a la función $T(n)$ que mide el número de instrucciones realizadas por el algoritmo para procesar los n elementos de entrada. Cada instrucción tiene asociado un costo temporal.

Afecta al tiempo de ejecución el orden en que se procesen los elementos de entrada.

Podría considerarse que los valores de los n casos que se presentan como entrada son los correspondientes: a un caso típico, o a un caso promedio, o de peor caso. El peor caso es el más sencillo de definir (el que demore más para cualquier entrada), pero si se desea otros tipos de entrada habría que definir qué se considera típico, o la distribución de los valores en el caso promedio.

Bibliografía:

- Brasard, G. and Bratley, P. (2000). Fundamentos de Algoritmia. Prentice Hall.