Saltar al contenido principal

Recursión y teorema maestro

Introducción

El teorema maestro

En mucho algoritmos del estilo de divide y conquista se hace una recursión del siguiente estilo

```
problema(n):
    dividir el problema en a sub-problemas de tamaño b/n
    resolver problema para cada uno de ellos (recursión)
    combinar las respuestas
```

El teorema maestro ayuda a saber cuánto tiempo se tardan este tipo de algoritmos bajo ciertas hipótesis sobre a, b y el tiempo de dividir el problema en subproblemas y de combinar las respuestas.

Teorema. Supongamos que tenemos una función que satisface la siguiente ecuación recursiva:

$$T(n)=aT(n/b)+f(n).$$

en donde $a\geq 1$ y $b>1$ son constantes y $f(n)$ es una función positiva. Definamos $d=\log_b a=\log a / \log b$, al cual le llamaremos el **exponente crítico**. Entonces, tenemos los siguientes tres casos:

1. Si $f(n)=O(n^{d-\epsilon})$ para alguna constante $\epsilon>0$, entonces $T(n)=\Theta(n^d)$.
2. Si $f(n)=\Theta(n^d \log ^k n)$ para alguna $k\geq 0$, entonces $T(n)=\Theta(n^d\log^{k+1}n)$.
3. Si $f(n)=\Omega(n^{d+\epsilon})$ para alguna constante $\epsilon>0$ y además $f(n)$ satisface la **condición de regularidad** $af(n/b)<cf(n)$ para alguna constante $c<1$ y n suficientemente grande, entonces $T(n)=\Theta(f(n))$.

Ejemplo. Usa el teorema maestro para resolver asintóticamente las siguiente recursiones. Hay una de ellas a la que no se le puede aplicar el teorema. Di cuál es y por qué.

a) $T(n)=3T(n/2)+n^2$.

b) $T(n)=4T(n/2)+n^2$.

c) $T(n)=2^nT(n/2)+n^n$.

d) $T(n)=16T(n/4)+n$.

a) En efecto $a=3\geq1$, $b=2> 1$ y $f(n)=n^2$, así que las hipótesis iniciales se cumplen. En este problema, el exponente crítico es $d=\log 3/ \log 2 \approx 1.585$. Notemos que

$$f(n)=n^2=\Omega(n^2)=\Omega(n^{d+\epsilon}).$$

Además, tenemos que $3f(n/2)=\frac{3}{4}n^2<\frac{4}{5}n^2$, por lo que se cumple la regularidad con $c=4/5$. Así, estamos en el caso (3) del teorema maestro. Por lo tanto, tenemos que $T(n)=\Theta(n^2)$.

b) En efecto $a=4\geq1$, $b=2>1$ y $f(n)=n^2$ es positiva, así que las hipótesis iniciales se cumplen. En este problema, el exponente crítico es $d=\log 4/ \log 2 = 2$. Notemos que

$$f(n)=n^2=\Theta(n^d).$$
Así, estamos en el caso (2) del teorema maestro, con $k=0$. Por lo tanto, tenemos que $T(n)=\Theta(n^2\log n)$.

c) Notemos que $a=2^n$, que no es una constante, de modo que no podemos aplicar el teorema maestro.

d) En efecto $a=16\geq 1$, $b=4>1$ y $f(n)=n$ es positiva, así que las hipótesis iniciales se cumplen. En este problema, el exponente crítico es $d=\log 16/ \log 4 = 2$. Notemos que

$$f(n)=n=O(n^d).$$

Así, estamos en el caso (1) del teorema maestro y por lo tanto $T(n)=O(n^2)$.

```
import numpy as np

print(np.log(3)/np.log(2))
print(np.log(4)/np.log(2))
print(np.log(16)/np.log(4))
```

```
1.5849625007211563
2.0
2.0
```

Ejercicio. Resuelve las siguiente recursiones usando el teorema maestro o explica por qué no se puede usar. 10, 13, 16, 19

a) $T(n)=16T(n/4)+n!$.

b) $T(n)=3T(n/3)+\sqrt{n}$.

c) $T(n)=3T(n/3)+n/2$.

d) $T(n)=64T(n/8)-n^2\log n$.

Revisar respuestas en https://people.csail.mit.edu/thies/6.046-web/master.pdf

Cuando estamos ordenando por recursión, lo que hacemos es partir el problema de n números en 2 problemas con $n/2$ números. La división y recombinación toma tiempo lineal, así que el tiempo total es

$$T(n)=2T(n/2)+cn.$$

Notemos que $a=b=2>1$ y que cn es positiva, de modo que podemos usar el teorema maestro. El exponente crítico es $d=\log 2 / \log 2 = 1$. Tenemos que $f(n)=cn=O(n^d)$, de modo que estamos en el caso (2) del teorema maestro. Entonces $T(n)=\Theta(n\log n)$.

Cuando estamos haciendo búsqueda binaria, lo que hacemos es partir el problema en un problema de $n/2$ números y para hacer esto necesitamos tiempo constante. Así que el tiempo de ejecución es recursivamente

$$T(n)=T(n/2)+f(n).$$

Tenemos $a=1\geq 1$, y $b=2>2$. Además la función $f(n)$ es positiva y $O(1)$. El exponente crítico es $d=\log 1 \ log 2 = 0$ y tenemos entonces que $f(n)$ es comparable a $n^d=1$. Estamos de nuevo en el caso 2 del teorema maestro y por lo tanto el tiempo total de ejecución satisface $T(n)=\Theta(\log n)$.

Sección 1

Sección 2

Sección 3

Tarea moral

Los siguientes problemas te ayudarán a practicar lo visto en esta entrada. Para resolverlos, necesitarás usar herramientas matemáticas, computacionales o ambas.

1. Problema
2. Problema
3. Problema
4. Problema
5. Problema

Anterior
Divide y conquista

Siguiente
Backtrack en búsquedas combinatorias