分治法作业

Edward

2019 年 9 月 18 日

1 section name

2 求渐近表示

1. $T(n)=T\left(\frac{9n}{10}\right)+n$

   $a=1, b=\frac{10}{9}, \log_b a=0$ ，需要逐项递推，设 $n=\frac{10}{9}^k$，则 $k=\log_{\frac{10}{9}} n$

$$
\begin{aligned}
T(n) &= T\left(\frac{9n}{10}\right)+n \\
&= T\left(\frac{10}{9}^{k-1}\right)+\frac{10}{9}^{k} \\
&= T\left(\frac{10}{9}^{k-2}\right)+\frac{10}{9}^{k-1}+\frac{10}{9}^{k} \\
&= \cdots \\
&= T(1)+\sum_{i=1}^{k}\frac{10}{9}^{i} \qquad\qquad . \\
&= T(1)+\frac{\frac{10}{9}\left(1-\frac{10}{9}^{k}\right)}{1-\frac{10}{9}} \\
&= T(1)+10\left(\frac{10}{9}^{k}-1\right) \\
&= T(1)+10(n-1) \\
&= \Theta(n)
\end{aligned}
$$

2. $T(n)=2T\left(\frac{n}{2}\right)+n^3$

$a=2, b=2, \log_b a=1$ ，由 master 定理知，$T(n)=\Theta(n^3)$

3. $T(n)=16T\left(\frac{n}{4}\right)+n^2$

   $a=16, b=4, \log_b a=2$ ，由 master 定理知，$T(n)=\Theta(n^2\log n)$

4. $T(n)=7T\left(\frac{n}{3}\right)+n^2$

   $a=7, b=3, \log_b a<2$ ，由 master 定理知，$T(n)=\Theta(n^2)$

5. $T(n)=7T\left(\frac{n}{2}\right)+n^2$

   $a=7, b=2, \log_b a>2$ ，由 master 定理知，$T(n)=\Theta\left(n^{\log_2 7}\right)$

6. $T(n)=2T\left(\frac{n}{4}\right)+\sqrt{n}$

   $a=2, b=4, \log_b a=\frac{1}{2}$ ，由 master 定理知，$T(n)=\Theta(\sqrt{n}\log n)$

7. $T(n)=T\left(\frac{2n}{3}\right)+T\left(\frac{n}{3}\right)+2n$

   $a^1=1, a^2=1, b^1=\frac{3}{2}, b^2=3$, 则满足 $(\frac{2}{3})^p+(\frac{1}{3})^p=1$ 的 p 有唯一值，且为 1。因此，由 master 定理的推广形式，$T(n)=\Theta(n\log n)$

3 设计算法寻找单峰分布的峰值

solution:

算法如下：

Algorithm 1 寻找单峰序列的峰值

```
1: procedure FindPeak(A(1:m))
2:     if m == 1 then
3:         return A(1)
4:     end if
5:     if m == 2 then
6:         if A(1) > A(2) then
7:             return A(1)
8:         else
9:             return A(2)
10:        end if
11:    end if
12:    if m == 3 then
13:        if A(1) > A(2) and A(2) > A(3) then
14:            return A(1)
15:        else if A(1) < A(2) and A(2) > A(3) then
16:            return A(2)
17:        else if A(1) < A(2) and A(2) < A(3) then
18:            return A(3)
19:        end if
20:    end if
21:    if m > 3 then k ← ⌊m/2⌋
22:        if A(k−1) > A(k) and A(k) > A(k+1) then
23:            return FindPeak(A(1 : k) )
24:        else if > (k−1) < A(k) and A(k+1) > A(k) then
25:            return FindPeak(A(k : m) )
26:        else if > (k−1) < A(k) and A(k+1) < A(k) then
27:            return A(k)
28:        end if
29:    end if
30: end procedure
```