高级算法设计与分析作业1
算法基础

傅彦璋 23S003008

2024.5.6

1

求a和b（$a \geq b$）的最大公约数的欧几里得算法描述如$GCD(a, b)$。算法的输入规模为$\log_2 a + \log_2 b$。

Algorithm 1: $GCD(a, b)$

Data: a, b满足$a \geq b$

Result: $gcd(a, b)$

```
1 r ← a mod b ;
2 if r ≠ 0 then
3 |   return GCD(b, r)
4 else
5 |   return b
6 end
```

记算法第k次递归调用$GCD(a, b)$时，b的值为b_k。算法第1行的$r = a \mod b$意味着$r \leq \frac{a}{2}$，所以，对任意k，一定有$b_{k+2} \leq \frac{b_k}{4}$。当某一次递归调用$GCD(a, 1)$，即$b = 1$的时候，算法一定停止。所以，算法的递归调用次数不超过$\log_2 b$。故求余操作不超过$\log_2 b$次，赋值操作不超过$3\log_2 b$次。

我们知道，计算$a \mod b$操作的时间代价是$O(\log_2 a)$的。所以所有求余操作产生的时间代价是$O(\log_2 a \cdot \log_2 b) \in O(\log_2^2 a)$的。

于是，对于规模为$O(n)$的输入规模，算法的时间复杂度为$O(n^2)$。

2

(a)

算法由两重循环构成。外层循环最多循环$n-1$次（变量i从2增长到n即停止），是有限的次数，一定会停止。在每个外层循环中，内层循环最多循环$i-1$次（变量j从$i-1$下降到1即停止），而$i-1<n$，也是有限的。所以算法在有限步内必然停止。

(b)

用循环不变量方法证明算法的正确性：

循环变量为i，初始时，$i=2$，$A[1]$来自输入且有序。当循环体执行前，若$A[1,2,...,i-1]$来自输入且有序，则循环体执行一遍后，$A[1,2,...,i]$来自输入且有序。终止时，$i=n+1$，故循环体执行最后一遍后，$A[1,2,...,n]$来自输入且有序。

(c)

最好情况：初始序列是有序的，算法的第5、6行不会执行。赋值操作仅出现在第2、3、7行。显然赋值操作次数和比较操作次数都是$O(n)$的。

最坏情况：初始序列是逆序的。对于每个外层循环，算法的内层循环都要执行$i-1$遍。赋值操作次数和比较操作次数都是$O(n^2)$的。

平均情况：所有输入服从均匀分布，我们知道，整个序列中逆序对数平均为$\frac{n(n-1)}{4}$，那么，在算法5、6行发生的赋值操作次数为$\frac{n(n-1)}{2}$，那么相对应的，在算法第4行发生的比较操作次数不低于$\frac{n(n-1)}{2}$。考虑上算法第2、3、7行的操作，总的赋值操作次数是$\frac{n^2}{2}+\frac{5n}{2}$。总之，比较次数和赋值次数皆是$O(n^2)$的。

3

算法的输入为整数n，输入规模为n的规模，在二进制模型下即$\log_2 n$，总之是$O(\log n)$的规模。循环将执行最多$n^{1/2}$次，即$\left(2^{\log_2 n}\right)^{1/2}$次。对于输入规模$n$，算法的基础操作次数是$2^{n/2}$，是指数时间算法。

4

算法的输入为整数n，输入规模为n的规模，在二进制模型下即$\log_2 n$，总之是$O(\log n)$的规模。当$n\geq 2$的时候，算法执行基本操作加法的次数是$n-1$次,即$2^{\log_2 n}-1$。即，对输入规模x，算法基本操作的次数是2^x-1，所以算法是指数时间算法。