第一章 概论习题

我保证没有抄袭他人作业

1. 请计算下面程序中循环语句的执行次数，并给出计算过程（要求是计算准确的次数而不是算法复杂度的阶，其中 n 为正整数）。

(1)

```
for (i = 1; i < n - 1; i++)
    for (j = n; j >= i; j--)
    {
        a[i][j] = 1;
    }
```

解：外层循环共执行 n - 2 次,对于每一个 i,内层循环执行 n - i + 1 次,共执行

n + n - 1 + n - 2 + ... + 3 = (n + 3)(n - 2) / 2 次

但 n = 1 时执行 0 次

(2)

```
int i = n * n;
while (i != 1)
{
    i = i / 2;
}
```

解：循环执行至 i = 1,设 $2^k <= n^2 < 2^{k+1}$, 则共执行 k 次,又有 $2\log_2 n - 1 < k <= 2\log_2 n$,即 $k = [2\log_2 n]$(向下取整),

故共执行$[2\log_2 n]$(向下取整)次

2. 证明

1) 对于任意实数 ， ，但 。

2) $n! = O(n^n)$，但 $n^n \neq O(n!)$

证明:

1) 存在 c = 1,对任意 n >= 1 有 $b^n < c * a^n$,即 $b^n = O(a^n)$

又,若 $a^n = O(b^n)$,则存在 c 使得存在一个 k 对任意的 n > k , $a^n < c * b^n$, 即 $c > (a / b)^n$,令 n 趋于正无穷,得到 c > +∞,矛盾

2) 因为 $n! = n * (n - 1) * ... * 1 <= n * n * ... * n = n^n$,所以 $n! = O(n^n)$,

若有 $n^n = O(n!)$,则存在 c 使得存在一个 k 对任意的 n > k , $n^n < c * n!$,即 c > (n / n) * (n / (n - 1)) * ... * (n / [n / 2]) * ... * (n / 1) >= (n / [n / 2]) * ... * (n / 1) >= $2^{[n / 2]}$,后者在 n 趋于无穷时也趋向正无穷,矛盾