

数据结构作业

计科 陈昕琪 PB22111711

1. 逻辑结构:

数据元素间的逻辑关系, 抽象化的相互关系

与数据的存储无关, 独立于计算机, 从具体问题抽象出来的数学模型。

物理结构:

数据元素及其关系, 在计算机存储器中的存储形式。

区别:

逻辑结构是比较抽象的, 数据间的关系

而物理结构是数据在存储器中的存储形式, 更加具象化

数据的逻辑结构和物理结构是密不可分的两个方面, 一个算法的设计取决于所选定的逻辑结构, 而算法的实现依赖于所采用的存储结构。

联系:

算法的设计取决于选定的逻辑结构。

算法的实现依赖于所采用的存储结构。

2.

```
(1)
Sum1(int n)
{
    int p=1, sum=0, m;
    for (m=1; m<=n; m++){
        p*=m; sum+=p;
    }
    return (sum);
}
```

1
n+1
n
n

$$T(n) = 3n + 3$$

复杂度最大 $T(n) = \sum_{i=1}^n 1 = n$

$$T(n) = O(n)$$

```
(2)
Sum2(int n)
{
    int sum=0, m, t;
    for (m=1; m<=n; m++){
        p=1;
        for (t=1; t<=m; t++) p*=t;
        sum+=p;
    }
    return (sum);
}
```

1
n+1
n
n(m+1), mn
n
1

$$T(n) = 2mn + 4n + 3$$

复杂度最大的内层 $T(n) = \sum_{m=1}^n \sum_{t=1}^m 1 = mn$

$$T(n) = O(mn)$$

```
(3)
Fun(int n)
{
    int s=i=0;
    while (s < n){
        s=s+i; i++;
    }
    return (s);
}
```

1
n+1
n, n

$$T(n) = 3n + 3$$

复杂度最大 $T(n) = \sum_{s=0}^{n-1} 1 = n$

$$T(n) = O(n)$$

```
(4) 递归函数
fact(int n)
{
    if (n<=1) return(1);
    else return(n*fact(n-1));
}
```

n
n-1

$$T(n) = 2n - 1$$

复杂度最大 $T(n) = n$

$$T(n) = O(n)$$

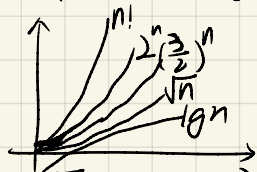
3. $(\frac{2}{3})^n < 2^{10} < \lg n < \sqrt{n} < n^{\frac{1}{2}} < (\frac{3}{2})^n < 2^n < n! < n^{10n} < n^n$

① $(\frac{2}{3})^n$ 递减, 2^{10} 不增长

② 分为 $\{(\frac{2}{3})^n, n^n, 2^n, n^{10n}\}$ $\{\sqrt{n}, n!, \lg n\}$ $\{n^{\frac{1}{2}}\}$ 三部分

$$n^n > n^{10n} > 2^n > (\frac{2}{3})^n$$

$$n! > \sqrt{n} > \lg n$$



4. (1) $T_1(n) = 5n^2 + O(n)$

$$T_2(n) = 3n^2 + O(n)$$

$T_2(n)$ 优于 $T_1(n)$

对于 T_2, T_3

$$T_2(n) = 3n^2 + 1000n + O(\lg n)$$

$$T_3(n) = 8n^2 + O(\lg n)$$

$T_2(n)$ 优于 $T_3(n)$

对于 T_2, T_4

$$T_2(n) = 3n^2 + O(n)$$

$$T_4(n) = 1.5n^2 + O(n \lg n)$$

T_2 劣于 $T_4(n)$

对于 T_1, T_3

$$T_1(n) = 5n^2 + O(n)$$

$$T_3(n) = 8n^2 + O(\lg n)$$

T_1 优于 T_3

优 \rightarrow 劣

$$T_4(n), T_2(n), T_1(n), T_3(n)$$