

# 数据结构习题 第2章

---





## 2-1

- 以学生的成绩表为例，按照逻辑结构、存储结构和相关操作三个方面，讨论他的数据结构。



## 参考答案

- 逻辑结构：线性表
- 存储结构：顺序结构/链接结构/散列结构
- 操作：增、删、查、改，排序

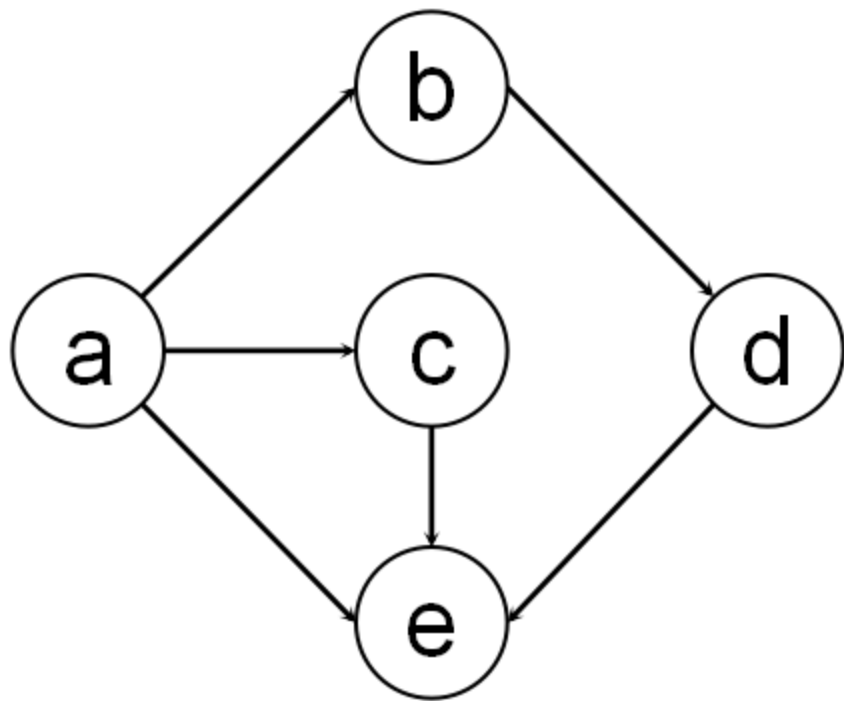


## 2-2

- 设数据的逻辑结构为 $L=(N, R)$ , 其中,  
 $N = \{a, b, c, d, e\}$ ,  
 $R = \{<a, b>, <a, c>, <a, e>, <b, d>, <c, e>, <d, e>\}$   
请画出对应的逻辑结构, 说明是何种结构



## 参考答案



图型结构：a有多个后继，e有多个前驱。



## 2-5

- 题目描述

试用ADL语言编写一个算法，判断任一整数  $n$  是否为素数



## 参考答案

算法CheckPrime (n. flag)

/\*判断整数n是否为素数，将结果保存到变量flag\*/

CP1[n≤1? ]

IF n≤1 THEN (flag←false. RETURN.)

CP2[初始化]

i←2. flag←true.

CP3[求余判断]

$i \leq n / 2$

WHILE  $i \leq n - 1$  DO (

IF (n MOD i)=0 THEN (flag←false. RETURN.)

i←i+1.

)



# 扩展

求n以内的所有素数  
素数筛选法





## 2-6

- 分析下面程序段的时间复杂性

```
int s=0, i, j, k;  
for (i=0; i<=n; i++)  
    for (j=0; j<=i; j++)  
        for (k=0; k<j; k++)  
            s++;
```



## 参考答案

- 以s++为基本运算
- 对每个i, 分析(j,k)两重循环的次数
  - j=0 循环次数为 0
  - j=1 循环次数为 1
  - .....
  - j=i 循环次数为 i
  - 因此对每个i, (j,k)二重循环的次数为 $i*(i+1)/2$
- 总循环次数为 $\text{sigma}(i*(i+1)/2) \ i=0..n$
- $T(n)=n*(n+1)*(n+2)/6$ , 算法的阶为 $O(n^3)$



## 2-9

- 将下列算法时间复杂性：  
 $O(n)$ ,  $O(2^n)$ ,  $O(\log_2 n)$ ,  $O(n \log_2 n)$ ,  $O(n^5)$ ,  
 $O(n^2+1)$ ,  $O(n^3-n^2)$   
按由低到高的顺序排列。其中， $n$ 是输入数据的规模。



## 参考答案

- $O(\log_2 n)$   $O(n)$   $O(n \log_2 n)$   $O(n^2 + 1)$   $O(n^3 - n^2)$   
 $O(n^5)$   $O(2^n)$



## 作业2-11

- 题目描述  
证明对正整数 $n \geq 3$ , 算法BS的元素比较次数 $T(n) \leq 5n/3 - 2$ 。
- 已知信息

$$T(n) = \begin{cases} 0 & n = 1 \\ 1 & n = 2 \\ T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 2 & n > 2 \end{cases}$$



算法BS(A, i, j, fmax, fmin)

/\* 在数组 A 的第 i 至第 j 个元素间寻找最大和最小元素, 已知  $i \leq j$ ; 假定 A 中元素互异 \*/

BS1. [递归出口]

IF  $i = j$  THEN (fmax  $\leftarrow$  fmin  $\leftarrow$  A[i]. RETURN.)

IF  $i = j - 1$  THEN (

IF  $A[i] < A[j]$  THEN (fmax  $\leftarrow$  A[j]. fmin  $\leftarrow$  A[i]).

ELSE (fmax  $\leftarrow$  A[i]. fmin  $\leftarrow$  A[j]).

RETURN.

)

BS2. [取中值] mid  $\leftarrow \lfloor (i+j)/2 \rfloor$

BS3. [递归调用]

BS(A, i, mid, gmax, gmin).

BS(A, mid+1, j, hmax, hmin).

BS4. [合并]

fmax  $\leftarrow$  max{gmax, hmax}.

fmin  $\leftarrow$  min{gmin, hmin}.



# 解题思路

- 本题的数学归纳法证明思路
  - 证明  $n=3,4,5$  时成立
  - 假设  $n \leq k$  时都成立，证明  $n = k+1$  时也成立



## 参考答案

- $n=3$  时,  $T(3)=T(1)+T(2)+2=3 \leq 5*3/3-2$ , 命题成立。
- $n=4$  时,  $T(4)=T(2)+T(2)+2=4 \leq 5*4/3-2$ , 命题成立。
- $n=5$  时,  $T(4)=T(2)+T(3)+2=6 \leq 5*5/3-2$ , 命题成立。





- 假设 $n \leq k$ 时命题成立。

$$n=k+1 \text{ 时, } T(k+1)=T(\lfloor (k+1)/2 \rfloor)+T(\lceil (k+1)/2 \rceil)+2, \quad \dots(1)$$

当 $k \geq 3$ 时, 有  $\lfloor (k+1)/2 \rfloor \leq k$ ,  $\lceil (k+1)/2 \rceil \leq k$

根据假设有:  $T(\lfloor (k+1)/2 \rfloor) \leq 5 \cdot \lfloor (k+1)/2 \rfloor / 3 - 2$ ,

$$T(\lceil (k+1)/2 \rceil) \leq 5 \cdot \lceil (k+1)/2 \rceil / 3 - 2 \text{ 成立,} \quad \dots(2)$$

$$\text{又知 } k+1 = \lfloor (k+1)/2 \rfloor + \lceil (k+1)/2 \rceil, \quad \dots(3)$$

由(1-3)可知,

$$\begin{aligned} T(k+1) &= T(\lfloor (k+1)/2 \rfloor) + T(\lceil (k+1)/2 \rceil) + 2 \\ &\leq [5 \cdot \lfloor (k+1)/2 \rfloor / 3 - 2] + [5 \cdot \lceil (k+1)/2 \rceil / 3 - 2] + 2 \\ &= 5/3 \cdot [\lfloor (k+1)/2 \rfloor + \lceil (k+1)/2 \rceil] - 2 = 5(k+1)/3 - 2 \end{aligned}$$

综上, 命题得证。



## 2-12

- 给出算法BS的非递归算法，并说明算法最多需要多大的辅助空间。



## 参考答案

基本思路：使用堆栈模拟递归过程。栈中每个元素设计为一个二元组  $(L, R)$ ，分别表示当前文件的起始端和终止端。还可以考虑设置两个全局变量  $\max, \min$ ，保存当前得到的最大元和最小元。取出栈顶元素进行处理时，分情况考虑  $L$  和  $R$  的关系。当  $L$  和  $R$  所界定的文件包含元素个数小于等于 2 时，直接更新  $\max$  和  $\min$ ；当大于 2 时，对半生成两个新的二元组并入栈，分别对应左子文件的边界和右子文件的边界（考虑入栈次序对处理过程的影响）。初始时栈中只有一个元素  $(1, n)$ 。当栈为空时，算法结束，此时  $\max$  和  $\min$  即为所求。

空间复杂性分析：栈中保存的最大元素个数就是该算法辅助空间的大小。该分析类似汉诺塔栈空间的分析过程。对于包含  $n$  个元素的文件，以上算法过程需要连续弹栈和压栈的最大次数是  $\log_2 n$ （自己思考原因），每次弹出 1 个元素，增加 2 个元素，净增 1 个元素。因此，栈空间所需保存的最大元素个数就是  $\log_2 n$ 。



*THE END*