

7. 对下述模式计算故障函数：

80

(a) a a a a b;

(b) a b a b a a;

(c) a b a a b a a b b.

(a)

j	0	1	2	3	4
pat	a	a	a	a	b
f	-1	0	1	2	-1

(b)

j	0	1	2	3	4	5
pat	a	b	a	b	a	a
f	-1	-1	0	1	2	0

(c)

j	0	1	2	3	4	5	6	7	8
pat	a	b	a	a	b	a	a	b	b
f	-1	-1	0	0	1	2	3	4	-1

### 练习

71

1. 在矩阵的数组表示中, 确定一个任意元素  $A[i][j]$  的位置, 并改变该元素的值, 将需要多少时间?

2. 对函数 `FastTranspose` (程序 2.11) 仔细分析计算时间和存储需求。你是否认为还可能有更快的算法?

### 练习1:

**O(1)**, 当给定了下标时, 可以直接计算出矩阵中第*i*行第*j*列所在的位置, 有对应的公式, 用O(1)的时间定位过去不需要遍历。

若考试中遇到这样的大题, 请写出步骤(公式)。

学了大O表达法之后, 一般题目问需要多长时间都是说求时间复杂度, 为了知道一个算法用多长时间而去专门写代码跑一下太麻烦, 而且由于机器性能等不同, 不同机器跑出来的时间也不一样, 这时候用类似秒之类的计时方法来计算算法的时间没有特别大的意义。

## 练习2:

自由发挥

1. 写出下列表达式的后缀形式:

(a)  $A * B * C$

(b)  $-A + B - C + D$

(c)  $A * -B + C$

(d)  $(A + B) * D + E / (F + A * D) + C$

(e)  $A \&\& B \parallel C \parallel ! (E > F)$  (assuming C++ precedence)

(f)  $!(A \&\& !((B < C) \parallel (C > D))) \parallel (C < E)$

说明: 书中给出的算术运算符的优先级在文档的最后面, (b)(c)中的-A, -B前的负号是单目运算符, 优先级最高, 方法就用平常的栈求解即可。

(a) 前缀:  $**ABC$

后缀:  $AB * C *$

(b) 前缀:  $+ - + - ABCD$

后缀:  $A - B + C - D +$

(c) 前缀:  $+ * A - BC$

后缀:  $AB - * C +$

(d) 前缀:  $++ * + ABD / E + F * ADC$

后缀:  $AB + D * EFAD * + / + c +$

(e) 前缀:  $|| || \&\& ABC! > EF$

后缀:  $AB \&\& c || EF > ! ||$

(f) 前缀:  $|| ! \&\& A ! || < BC > CD < CE$

后缀:  $ABC < CD > || ! \&\& ! CE < ||$

2. 根据本节所述的优先级, 回答下列问题:

(a) 在函数 postfix 中(程序 3.19), 如果表达式 e 含有 n 个操作符和分界符, 栈中可能存入元素的最大数目是多少?

(b) 如果 e 含有 n 个操作符, 并且括号嵌套的最大深度为 6 层, 栈中可能存入元素的最大数目是多少?

## 练习2:

(a) 假设表达式 e 中有 n 个操作符和分界符, 书中的给出的优先级有 7 种, 加上小括号总共有八种优先级的操作符, 由于栈内操作符需要严格单调递增, 所以当栈中存放最大数目的操作符的情况为:

$|| \&\& == < + * ! (|| \&\& == < + * ! (|| \&\& == < + * ! \dots$

设有 m 个括号, 则

$$m * 9 + 7 = n$$

$$m = (n - 7) / 9$$

说明: 9 为一个左括号, 7 个不同优先级的操作符和一个右括号, 7 是最前面的 7 个优先级不同的操作符

栈中能存储的操作符有

$$8 * m + 7 + 1 = (8n + 16) / 9 \quad // \text{栈底部需要有 \# 号, 所以会多加一个 1}$$

(b)  $1 + 7 + 8 * 6 = 56$

-----

优先级	操作符
1	unary minus, !
2	*, /, %
3	+, -
4	<, <=, >=, >
5	==, !=
6	&&
7	

图 3.12 C++中操作符的优先级

答案可能有问题，可以提出来。