1．Think of an algorithm that uses a Stack to eﬃciently check for unbalanced brackets. What is the maximum number of characters that will appear on the stack at any time when the algorithm analyzes the string ([]()[()])? ( )

(a) 3 (b) 4 (c) 5 (d) 6 (e) None of these is correct.

2. Consider a sequence of push and pop operations used to push the integers 0 through 9 on a stack. The numbers will be pushed in order, however the pop operations can be interleaved with the push operations, and can occur any time there is at least one item on the stack. When an item is popped, it is printed to the terminal. Which of the following could NOT be the output from such a sequence of operations? ( )

(a) 0 1 2 3 4 5 6 7 8 9

(b) 4 3 2 1 0 5 6 7 8 9

(c) 5 6 7 8 9 0 1 2 3 4

(d) 4 3 2 1 0 9 8 7 6 5

(e) All of these output sequences are possible.

3. 一个中缀表达式为a+(3-x/4)\*(y-2)，则其对应的后缀表达式：( )。

4. 写出后缀表达式3 4.6 5 \* 2 - \*的值：( )。

5. Write an algorithm to deletes all occurrences (if any) of x from s and leaves the remaining entries in s in the same relative order.

要求：算法不能假设栈的具体实现方法，需直接调用栈的基本操作来完成算法。函数原型参考**：int delete\_all(Stack &s, Stack\_entry x)，**函数功能为将s中的x删除，并返回被删除元素的个数。

**int delete\_all(Stack &s, Stack\_entry x){**

**Stack temps;**

**Stack\_entry item;**

**int count=0;**

**while (!s.empty()){**

**s.top(item);**

**s.pop();**

**if (item!=x)**

**temps.push(item);**

**else**

**count++;**

**}**

**while (!temps.empty()){**

**temps.top(item);**

**temps.pop();**

**s.push(item);**

**}**

**return count;**

**}**

**}**