

《数据结构与算法B》随堂摸底测试（一）

学号_____ 姓名_____ 教师/教室_____

一、 选择题

- 1 在插入排序算法中，如果待排序的序列已经是有序的，则插入排序算法的时间复杂度为（ ）。
A. $O(n^2)$ B. $O(n \log n)$ C. $O(n)$ D. $O(1)$
- 2 在二分查找算法中，每次比较后都将搜索范围缩小一半，若要在长度为 n 的数组中查找一个元素，最坏情况下需要比较多少次（ ）。
A. n B. $\lfloor n/2 \rfloor$ C. $\lfloor \log n \rfloor$ D. $\lfloor \log n \rfloor + 1$
- 3 下列排序算法中，某一趟结束后未必能选出一个元素放在其最终位置上的是：（ ）
A. 堆排序 B. 冒泡排序 C. 快速排序 D. 直接插入排序
- 4 下述几种排序方法中，要求内存量（空间复杂性）最大的是（ ）
A. 插入排序 B. 选择排序 C. 快速排序 D. 归并排序

二、 判断（对填写“Y”，错填写“N”）

1. （ ） 分治法和动态规划法都运用了将问题分解为规模较小的子问题的思想。
2. （ ） 归并排序算法一定比简单插入排序算法的执行效率高。
3. （ ） 在待排序元素为正序情况下，直接插入排序可能比快速排序的时间复杂度更小。

三、 简答题

1. 对数组进行**递增排序**过程中，需要进行多次扫描，采取不同排序方法会产生不同的排序中间结果。请表示出不同排序算法对下列字符数组第一次扫描后的排序结果。（注：快速排序使用第一个元素作为分界点）

	E, C, A, M, S, R, D, F
直接插入排序	
Shell 排序 ($d_1=N/2$)	
直接选择排序	
起泡排序	
快速排序	
二路归并排序	

2. 对待排序的关键码集合 { E, C, A, B, S, R, D, F } 进行堆排序, 得到递增序列。请给出初始建堆的树结构已经对应的关键集合, 并简单叙述堆排序的基本过程。

- 建立大根堆后: _____
- 互换, 重建堆后:

四、 算法填空题

//完成下列快速排序算法, 并补充相应的代码注释。

void quickSort(SortObject *pvector, int l, int r) // SortObject 为记录类型 RecordNode 的数组

```
{
    int i, j;
    RecordNode temp;    // RecordNode 为记录类型, 排序码为 RecordNode 的成员 key
    if(l>=r)    return;    /*只有一个记录或无记录, 则无需排序*/
    i=l;    j=r;
    temp=pvector->record[i];
    while( i != j )    /*寻找 temp 的最终位置*/
    {
        while( ( _____ )&&( _____ ) )
            j--;    /*从右到左扫描, 查找第排序码小于 temp.key 的记录*/
        if(i<j)    _____ = pvector->record [j];
        while( ( _____ )&&( _____ ) )
            i++;    /*从左到右扫描, 查找第排序码大于 temp.key 的记录*/
        if(i<j)    _____ = pvector->record[i];
    }
    pvector->record[i]=temp;    /*找到 temp 的最终位置*/
    _____;    /*找到递归处理左区间*/
    quickSort(pvector,i+1,r);    /*找到递归处理右区间*/
}
```