

第 1 题直接插入排序算法部分代码参考如下：

```
#include <stdio.h>
#define TRUE 1
#define FALSE 0
#define OK 1
#define ERROR 0
#define MAXSIZE 20          //一个用作示例的小顺序表的最大长度
#define LT(a, b) (a<b)
#define EQ(a, b) (a==b)
typedef int Status;
typedef int KeyType;
typedef int InfoType;
typedef struct {
    KeyType key;           //关键字项
    InfoType otherinfo;    //其他数据项
}RedType;
typedef struct {
    RedType r[MAXSIZE+1]; //r[0]闲置或用作哨兵单元
    int length;           //顺序表长度
}SqList;                  //顺序表类型

//初始化顺序表
Status InitList(SqList &L) {
    // 构造一个空的顺序表 L。
    L.length = 0;         //空表长度为 0
    return OK;
} // InitList

//创建 n 个元素的顺序表
Status CreateList(SqList &L, int n) {
    int i;
    printf("输入%d 个元素: \n",n);
    for(i=1; i<=n; i++) {
        scanf("%d", &L.r[i].key);
        ++L.length;
    }
    if (L.length==0) return ERROR;    // 创建失败
    return OK;
}

//输出顺序表
Status DispList(SqList &L) { //输出排序结果
    int i;
    if (L.length==0) return ERROR;
    for(i=1; i<=L.length; i++)
        printf("%d ", L.r[i].key);
    printf("\n");
    return OK;
}
```

```
void main()
{
    SqlList L;
    int n;
    printf("初始化顺序表，");
    InitList(L);
    if (L.length==0)
        printf("顺序表为空！ \n");
    printf("输入顺序表的元素个数： ");
    scanf("%d", &n);
    CreateList(L, n);
    printf("输出%d 个元素的顺序表如下： \n", n);
    DispList(L);
    InsertSort(L);
    printf("直接插入排序结果如下： \n");
    DispList(L);
}
```