

```

#include <stdio.h>
#include <malloc.h>

void shellSort(int *a, int len); // 函数声明

int main(void)
{
    int i, len, * a;
    printf("请输入要排的数的个数: ");
    scanf("%d",&len);
    a = (int *)malloc(len * sizeof(int)); // 动态定义数组
    printf("请输入要排的数: \n");
    for (i = 0; i < len; i++) { // 数组值的输入
        scanf("%d",&a[i]);
    }
    shellSort(a, len); // 调用希尔排序函数
    printf("希尔升序排列后结果为: \n");
    for (i = 0; i < len; i++) { // 排序后的结果的输出
        printf("%d\t",a[i]);
    }
    printf("\n");

    return 0;
}

void shellSort(int *a, int len)
{
    int i, j, k, tmp, step; // step 为步长
    for (step = len / 2; step > 0; step /= 2) { // 步长初始化为数组长度的一半，每次遍历后步长减半，
        for (k = 0; k < step; k++) { // 变量 i 为每次分组的第一个元素下标
            for (i = k + step; i < len; i += step) { //对步长为step的元素进行直插排序，当step为1时，就
是直插排序
                tmp = a[i]; // 备份a[i]的值
                j = i - step; // j初始化为i的前一个元素（与i相差step长度）
                while (j >= 0 && a[j] > tmp) {
                    a[j + step] = a[j]; // 将在a[i]前且比tmp的值大的元素向后移动一位
                    j -= step;
                }
                a[j+step] = tmp;
            }
        }
    }
}

```