

```
#include <iostream>

void shellSort(int arr[], int size) {
    // 使用希尔增量序列进行排序
    for (int gap = size / 2; gap > 0; gap /= 2) {
        // 对每个子序列进行插入排序
        for (int i = gap; i < size; i++) {
            int temp = arr[i];
            int j = i;

            // 在子序列中进行插入排序
            while (j >= gap && arr[j - gap] > temp) {
                arr[j] = arr[j - gap];
                j -= gap;
            }

            arr[j] = temp;
        }
    }
}
```

这是希尔排序函数，它接收一个数组和数组的大小作为参数。希尔排序使用希尔增量序列来确定子序列的间隔，并对每个子序列进行插入排序。外部循环根据希尔增量序列调整间隔大小，逐渐缩小间隔直到为1。内部循环对每个子序列进行插入排序，将当前元素与间隔之前的元素进行比较和交换，以确保子序列中的元素有序。

```
int main() {
    int arr[] = {64, 34, 25, 12, 22, 11, 90};
    int size = sizeof(arr) / sizeof(arr[0]);

    shellSort(arr, size);

    std::cout << "排序结果: ";
    for (int i = 0; i < size; i++) {
        std::cout << arr[i] << " ";
    }
    std::cout << std::endl;

    return 0;
}
```

这是主函数，用于测试希尔排序算法。它创建一个包含待排序元素的数组，并计算数组的大小。然后，它调用 shellSort 函数对数组进行排序。最后，它输出排序结果。

希望这些注释和解释能帮助您理解希尔排序算法的思路和实现。