

```
#include <iostream>

void insertionSort(int arr[], int size) {
    for (int i = 1; i < size; i++) {
        int key = arr[i];
        int j = i - 1;

        // 将大于key的元素向右移动
        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j--;
        }

        // 将key插入正确的位置
        arr[j + 1] = key;
    }
}
```

这是插入排序函数，它接收一个数组和数组的大小作为参数。插入排序通过将未排序部分的元素逐个插入到已排序部分的正确位置来实现排序。外部循环从第二个元素开始到最后一个元素，选择一个元素作为key。内部循环将大于key的元素向右移动，为key找到合适的位置。在内部循环中，将每个比key大的元素向右移动一个位置，直到找到合适的位置。然后，将key插入到该位置。通过多次的插入操作，将元素逐步插入到已排序部分的正确位置，实现排序。

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

    insertionSort(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，用于测试插入排序算法。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insertionSort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插入排序算法的思路和实现。