Write a function template for selection sort that inputs, sorts and outputs an integer array and a float array

Program-

```
#include <iostream>
using namespace std;

// Function template for selection sort
template <typename T>
void selectionSort(T arr[], int size) {
    // Selection Sort algorithm
    for (int i = 0; i < size - 1; i++) {
        int minIndex = i;  // Assume the current position is the minimum
        for (int j = i + 1; j < size; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;  // Update minIndex if a smaller element is found
            }
        }
        // Swap the found minimum element with the first element
        T temp = arr[minIndex];
        arr[minIndex] = arr[i];
        arr[i] = temp;
    }
}

// Function to print an array
template <typename T>
void printArray(T arr[], int size) {
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
}

int main() {
    // Integer array
    int intArray[] = {64, 25, 12, 22, 11};
    int intSize = sizeof(intArray) / sizeof(intArray[0]);

    // Float array
    float floatArray[] = {64.5, 25.3, 12.1, 22.8, 11.7};
    int floatSize = sizeof(floatArray) / sizeof(floatArray[0]);

    // Sorting integer array
    cout << "Original Integer Array: ";
    printArray(intArray, intSize);

    selectionSort(intArray, intSize);
    cout << "Sorted Integer Array: ";
    printArray(intArray, intSize);

    // Sorting float array
    cout << "\nOriginal Float Array: ";
    printArray(floatArray, floatSize);

    selectionSort(floatArray, floatSize);

    cout << "Sorted Float Array: ";
    printArray(floatArray, floatSize);

    return 0;
}
```