Day 25/180 Selection Sort in C++

1: Use a Selection Sort Algorithm to sort the array of integers in decreasing order.

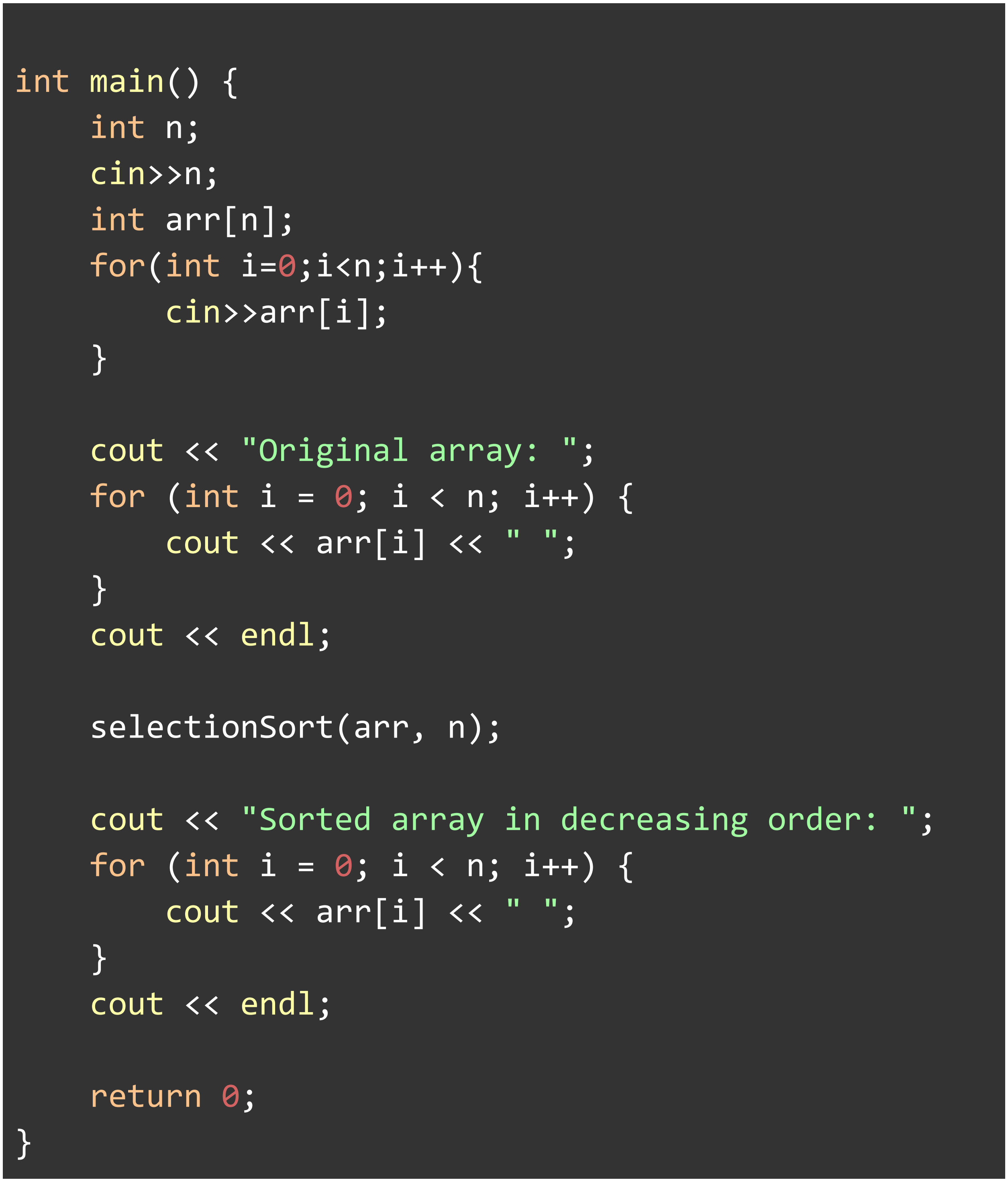
Solution:

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

void selectionSort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++) {
        // Assume the current element is the maximum
        int maxIndex = i;

        // Find the index of the maximum element in the unsorted part of the array
        for (int j = i + 1; j < n; j++) {
            if (arr[j] > arr[maxIndex]) {
                maxIndex = j;
            }
        }

        // Swap the maximum element with the current element (if they are different)
        if (maxIndex != i) {
            std::swap(arr[i], arr[maxIndex]);
        }
    }
}
```



```
int main() {
    int n;
    cin>>n;
    int arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }

    cout << "Original array: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    selectionSort(arr, n);

    cout << "Sorted array in decreasing order: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}
```

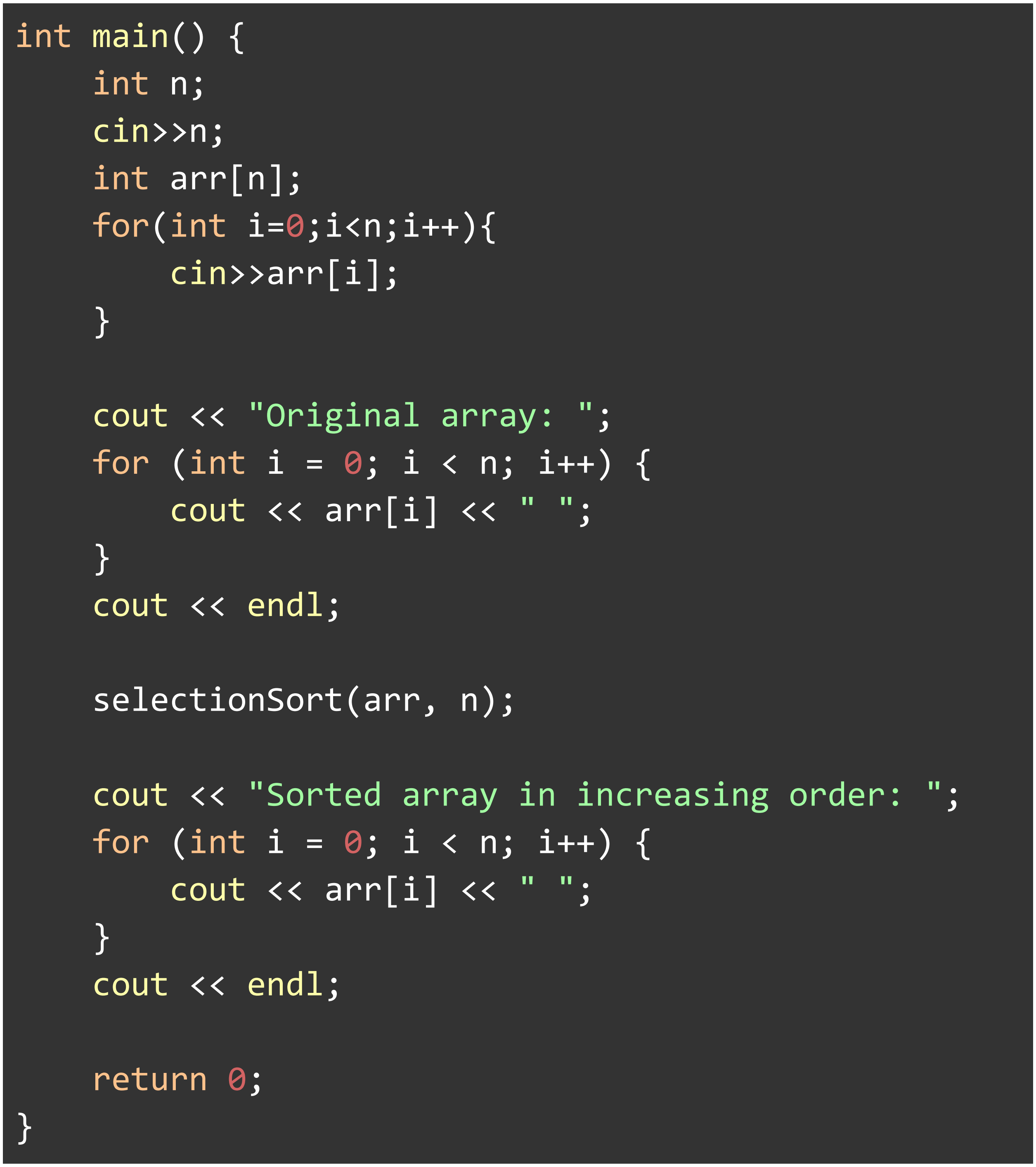
2: Use a Selection Sort Algorithm to sort the array of integers in increasing order by taking the highest number to last place. Question was explained in the class.

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

void selectionSort(int arr[], int n) {
    for (int i = n - 1; i > 0; i--) {
        // Assume the current element is the maximum
        int maxIndex = i;

        // Find the index of the maximum element in
the unsorted part of the array
        for (int j = 0; j < i; j++) {
            if (arr[j] > arr[maxIndex]) {
                maxIndex = j;
            }
        }

        // Swap the maximum element with the last
element in the unsorted part
        if (maxIndex != i) {
            swap(arr[i], arr[maxIndex]);
        }
    }
}
```



```
int main() {
    int n;
    cin>>n;
    int arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }

    cout << "Original array: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    selectionSort(arr, n);

    cout << "Sorted array in increasing order: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}
```

3: Use a Selection Sort Algorithm to sort the array of char in ascending order.

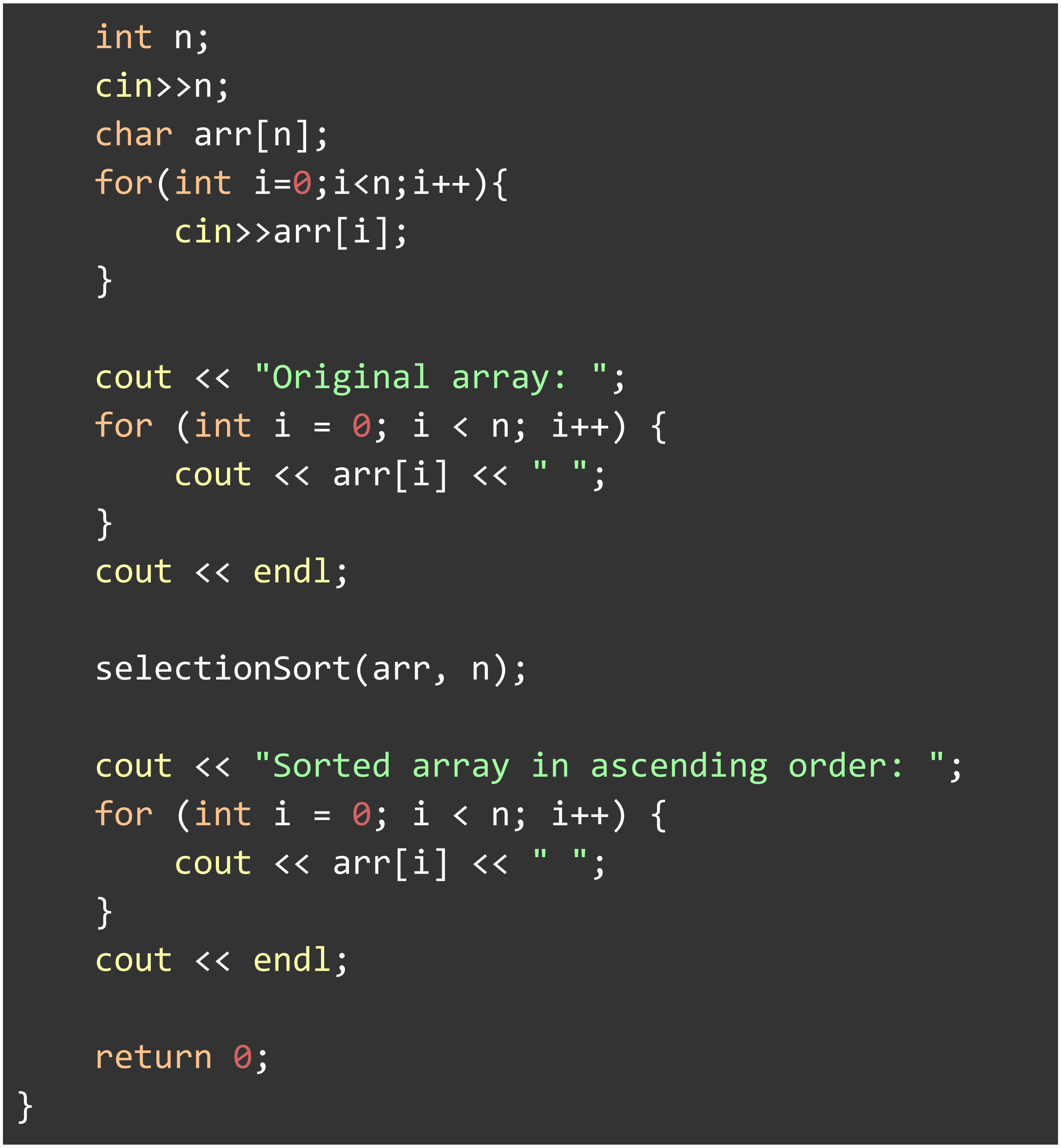
```
#include <iostream>
using namespace std;

void selectionSort(char arr[], int n) {
    for (int i = 0; i < n; i++) {
        // Assume the current element is the maximum
        int maxIndex = i;

        // Find the index of the maximum element in
the unsorted part of the array
        for (int j = i + 1; j < n; j++) {
            if (arr[j] < arr[maxIndex]) {
                maxIndex = j;
            }
        }

        // Swap the maximum element with the last
element in the unsorted part
        if (maxIndex != i) {
            swap(arr[i], arr[maxIndex]);
        }
    }
}


int main() {
```



```
    int n;
    cin>>n;
    char arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }

    cout << "Original array: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    selectionSort(arr, n);

    cout << "Sorted array in ascending order: ";
    for (int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}
```