

## Ordenamiento con cubetas

```
#include <algorithm>
#include <cstdlib>
#include <ctime>
#include <iostream>
```

using namespace std;

```
const int NUMERO_BUCKETS = 10;
const int MAXIMO_NUM = 100;
```

```
void bucket_sort(int arr[], int n) {
    int buckets[NUMERO_BUCKETS] = {MAXIMO_NUM / NUMERO_BUCKETS};
    int bucket_count[NUMERO_BUCKETS] = {0};

    for (int i = 0; i < n; ++i) {
        int bucket_index = arr[i] / (MAXIMO_NUM / NUMERO_BUCKETS);
        buckets[bucket_index] = bucket_count[bucket_index] + 1;
    }

    int index = 0;
    for (int i = 0; i < NUMERO_BUCKETS; ++i) {
        sort(buckets[i], buckets[i] + bucket_count[i]);
        for (int j = 0; j < bucket_count[i]; ++j) {
            arr[index++] = buckets[i][j];
        }
    }
}
```

```
int main() {
    srand(time(0));
    int arr[NUMERO_BUCKETS];

    for (int i = 0; i < NUMERO_BUCKETS; ++i) {
        arr[i] = rand() % MAXIMO_NUM;
    }

    cout << "ordenamiento no clasificado: ";
    for (int i = 0; i < NUMERO_BUCKETS; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;

    bucket_sort(arr, NUMERO_BUCKETS);
    cout << "ordenamiento clasificado: ";
    for (int i = 0; i < NUMERO_BUCKETS; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;
    return 0;
}
```