



C++ для олимпиадного программирования

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Библиотеки

```
1 #include <iostream> - ввод/вывод
2 #include <algorithm> - sort, lower_bound, swap, merge, min, max, ...
3 #include <cmath> - sin, cos, sqrt ...
4
5 #include <vector>
6 #include <string>
7 #include <map>
8 #include <set>
9 #include <queue>
10 #include <unordered_set>
11 #include <complex>
12 #include <unordered_map>
13 #include <bitset>
14
15 #include <ctime>
16 #include <cassert>
17 #include <random>
18
19 #include <bits/stdc++.h> - подключит все, что вам нужно
```



std

```
using namespace std;
```

Теперь:

- `cin` вместо `std::cin`;
- `sort` вместо `std::sort` и т. д.



Пример кода

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 int main() {
6     int a, b;
7     cin >> a >> b;
8     cout << a + b << endl;
9     return 0;
10 }
```



Стандартные типы и классы

```
1 int, long long, char, string, double, long double
2
3 vector, set, map
4 vector<int>, vector<string>
5 set<int>, set<string>
6 map<int, int>, map<string, char>
7
8 pair<int, int>
9 make_pair(a, b)
10
11 vector<pair<int, pair<set<int>, string>>>
```



vector

```
1 vector<int> a, b(5), c(6, 0);
2 b[0] = 5;
3 a.push_back(4);
4 a.size()
5 i < (int)a.size() - 1
6 c.clear()
7 c.resize(10);
8 b.assign(4, 7);
9 int ans = 0;
10 for (int x : b) {
11     ans += x;
12 }
13 for (int i = 0; i < a.size(); i++) {
14     ans += a[i];
15 }
```



set

```
1 set<int> a;
2 a.insert(5);
3 a.insert(5);
4 a.count(5);
5 a.erase(5);
6 a.erase(5);
7 a.size()
8 a.clear()
9 int ans = 0;
10 for (int x : a) {
11     ans += x;
12 }
```



map

```
1 map<string, int> a;
2 a["Apple"] = 5;
3 a["Orange"] = 2;
4
5 int x = a["Apple"];
6 int y = a["Avocado"];
7 a.count("Avocado");
8 a.size()
9 a.clear()
10 int ans = 0;
11 vector<string> names;
12 for (pair<string, int> x : a) {
13     names.push_back(x.first);
14     ans += x.second;
15 }
```



Санкт-Петербургский
государственный
университет
www.spbu.ru



О-нотация, асимптотика алгоритмов

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Время работы программы

C++ выполняет 10^8 операций в секунду.



Как считать время работы?

```
1 int ans = 0;
2 for (int i = 0; i < n; i++) {
3     ans += a[i];
4 }
5 for (int j = 0; j < m; j++) {
6     ans += b[j];
7 }
8 -----
9 int ans = 0;
10 for (int i = 0; i < n; i++) {
11     for (int j = 0; j < m; j++) {
12         ans += a[i][j];
13     }
14 }
```



Асимптотика

$n, 2n, 5n, n/2$ — одно и то же для нас.

$$f(n) = O(g(n)) \iff f(n) \leq C \times g(n)$$

$$O(n) \quad n \leq 10^8$$

$$O(n \times \log n) \quad n \leq 10^6$$

$$O(n^2) \quad n \leq 10^4$$

$$O(n^3) \quad n \leq 500$$



Санкт-Петербургский
государственный
университет
www.spbu.ru

Постановка задачи о сортировке

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Сортировка массива

Сортировка — упорядочивание элементов в возрастающем (или убывающем) порядке.

(5, 100, 6, 1, 2) -> (1, 2, 5, 6, 100)

(5, 100, 6, 1, 2) -> (100, 6, 5, 2, 1)

(5, 100, 6, 1, 100, 2) -> (1, 2, 5, 6, 100, 100)

(1) -> (1)

() -> ()



Стабильность сортировки

Сортировка называется стабильной, если она оставляет равные элементы в том же порядке, в каком они были.

((25, София), (28, Анна), (25, Павел))

((25, София), (25, Павел), (28, Анна)) — стабильно.

((25, Павел), (25, София), (28, Анна)) — НЕ стабильно.



Санкт-Петербургский
государственный
университет
www.spbu.ru



Сортировка пузырьком, вставками и выбором

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Сортировка пузырьком

5, 2, 100, 1, 4, 6



Сортировка вставками

5, 2, 100, 1, 4, 6



Сортировка выбором

5, 2, 100, 1, 4, 6



Санкт-Петербургский
государственный
университет
www.spbu.ru



Сортировка слиянием

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Сортировка слиянием

5, 2, 100, 1, 4, 6, 3, 7



Санкт-Петербургский
государственный
университет
www.spbu.ru



Быстрая сортировка

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Быстрая сортировка (Quick Sort)

5, 2, 4, 1, 4, 6, 1, 4



Санкт-Петербургский
государственный
университет
www.spbu.ru



Сортировка подсчетом

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Сортировка подсчетом

1, 5, 2, 7, 2, 5, 1, 4, 7, 1, 1

4 единицы

2 двойки

0 троек

1 четверка

2 пятерки

0 шестерок

2 семерки

1, 1, 1, 1, 2, 2, 4, 5, 5, 7, 7

$O(n + C)$



Санкт-Петербургский
государственный
университет
www.spbu.ru



Встроенная сортировка, избыточность сортировки

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Встроенная сортировка

```
1 vector<int> a;
2 sort(a.begin(), a.end());
3
4 int b[N];
5 sort(b, b + N);
6
7 sort(a.begin() + 2, a.begin() + 8);
8
9 reverse(a.begin(), a.end());
10 sort(a.rbegin(), a.rend());
11
12 bool my_cmp(int x, int y) {
13     return x % 3 < y % 3;
14 }
15
16 sort(a.begin(), a.end(), my_cmp);
```



Избыточность сортировки

```
1 int min_val = a[0];
2 for (int i = 1; i < a.size(); i++) {
3     min_val = min(min_val, a[i]);
4 }
5
6
7 int min1 = 1000000000, min2 = min1;
8 for (int i = 0; i < a.size(); i++) {
9     if (a[i] < min1) {
10         min2 = min1;
11         min1 = a[i];
12     } else if (a[i] < min2) {
13         min2 = a[i];
14     }
15 }
```



Санкт-Петербургский
государственный
университет
www.spbu.ru



Бинарный поиск

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Бинарный поиск: идея

0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1

- Найти первую единицу
- Делим каждый раз отрезок пополам,
переходим в одну из половин

$O(\log n)$



Бинарный поиск: алгоритм

1, 2, 5, 8, 10, 15, 100, 250, 403



Бинарный поиск: код

```
1 int find_elem_pos(int elem) {  
2     int l = 0, r = a.size();  
3     while (r - l > 1) {  
4         int mid = (r + l) / 2;  
5         if (a[mid] > elem) {  
6             r = mid;  
7         } else {  
8             l = mid;  
9         }  
10    }  
11    return l;  
12 }
```



Санкт-Петербургский
государственный
университет
www.spbu.ru

Встроенный бинарный поиск, вещественный бинарный поиск

Горбачев Егор Андреевич

преподаватель олимпиадного программирования





Встроенный бинарный поиск

```
1 int pos1 = lower_bound(a.begin(), a.end(), 5) - a.begin();
2 int pos2 = upper_bound(a.begin(), a.end(), 5) - a.begin();
3 bool have_5 = binary_search(a.begin(), a.end(), 5);
```



Вещественный бинарный поиск

```
1 double l = 0, r = n + 1;
2 while (r - l > 1e-9) {
3     double mid = (r + l) / 2;
4     if (mid * mid > n) {
5         r = mid;
6     } else {
7         l = mid;
8     }
9 }
10 -----
11 double l = 0, r = n + 1;
12 for (int i = 0; i < 100; i++) {
13     double mid = (r + l) / 2;
14     if (mid * mid > n) {
15         r = mid;
16     } else {
17         l = mid;
18     }
19 }
20
```



Санкт-Петербургский
государственный
университет
www.spbu.ru



Источники

1. В онлайн-курсе использованы материалы и схемы, разработанные автором.