Козунов Алексей, 12 группа, вариант 14

Отчет

Алгоритм: std::remove_if

Задача: удалить из случайного массива числа, меньшие 18

Код программы:

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <execution>
#include <ctime>
#include <random>

using namespace std;

vector<int> random_numbers(long size) {
	vector<int> numbers;
	for (int i = 0; i < size; i++) {
		numbers.push_back(rand() % 101);
	}
	return numbers;
}

bool remove_number(int number) {
	return number < 18;
}

double sequential_execution(long size) {
	auto mas = random_numbers(size);

	auto start_time = clock();
	std::remove_if(mas.begin(), mas.end(), remove_number);
	auto end_time = clock();
	double sequent_time = (double)(end_time - start_time) / CLOCKS_PER_SEC;
	std::cout << "Sequent time :" << sequent_time << endl;

	return sequent_time;
}

double parallel_execution(long size) {
	auto mas = random_numbers(size);

	auto start_time = clock();
	std::remove_if(std::execution::par, mas.begin(), mas.end(), remove_number);
	auto end_time = clock();
	double parallel_time = (double)(end_time - start_time) / CLOCKS_PER_SEC;
	std::cout << "Parallel time :" << parallel_time << endl;

	return parallel_time;
}

int main() {

	std::cout << "\nTask size : " << 1000000 << endl;

	double sequent_time = sequential_execution(1000000);
	double parallel_time = parallel_execution(1000000);

	double acceleration_factor = sequent_time / parallel_time;
	std::cout << "Acceleration : " << acceleration_factor << endl;
```

```
        std::cout << "\nTask size : " << 100000000 << endl;

        sequent_time = sequential_execution(100000000);
        parallel_time = parallel_execution(100000000);

        double acceleration_factor = sequent_time / parallel_time;
        std::cout << "Acceleration : " << acceleration_factor << endl;
}
```

Результат:

```
Task size : 1000000
Sequent time : 0.031
Parallel time : 0.009
Acceleration : 4.44444

Task size : 100000000
Sequent time :3.003
Parallel time : 0.44526
Acceleration : 6.74437
```