Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное учреждение высшего образования «Пермский национальный исследовательский политехнический университет»

Лабораторная работа

**“Работа с одномерными массивами”**

Выполнил:

студент группы РИС-23-1б

Жуланов Никита Андреевич

Проверила:

доцент кафедры ИТАС

О. А. Полякова

2023 г.

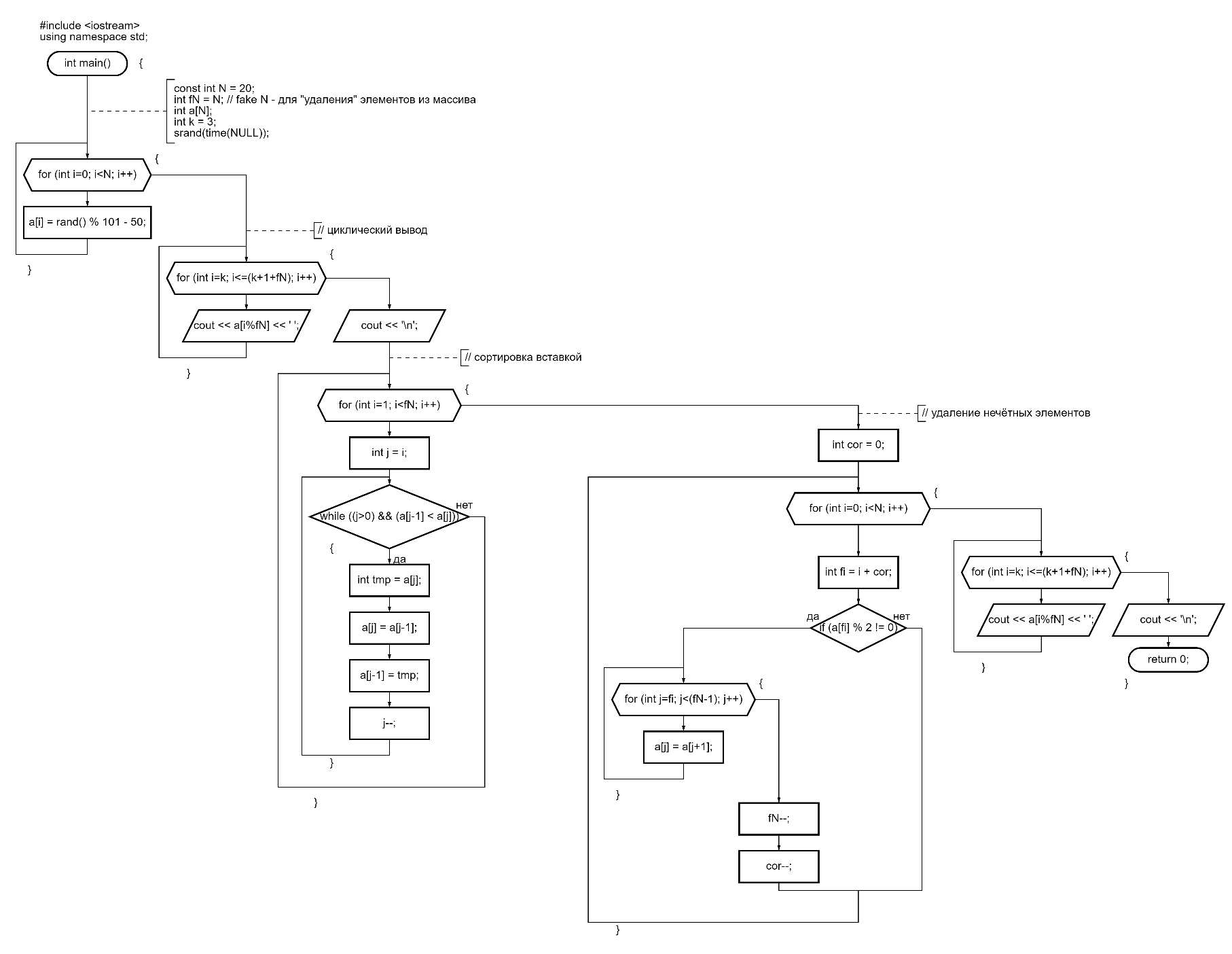
**Разработка алгоритма** **кольцевого сдвига элементов в массиве**

**Постановка задачи (25 вариант):**

1. Реализовать с использованием массива однонаправленное кольцо (просмотр возможен справа налево, от первого элемента можно перейти к последнему).
2. Распечатать полученный массив, начиная с k-ого элемента и до k+1.
3. Упорядочить элементы по убыванию.
4. Удалить из кольца нечетные элементы.
5. Распечатать полученный массив, начиная с k-ого элемента и до k+1.

**Анализ задачи:**

1. Кольцевой просмотр реализуется при помощи операции взятия остатка от деления на размер массива (i % N = 0, где i=N).
2. Сортировка по убыванию реализуется через сортировку вставкой.
3. Удаление нечётных элементов. Как только встретился нечётный элемент:
   1. Элементы справа от него сдвигаются влево, “затирая” его.
   2. Граница массива уменьшается на 1.
   3. Индекс текущего элемента корректируется уменьшением на 1.

**Блок-схема:**

**Код на языке C++:**

#include <iostream>

#include <ctime>

using namespace std;

int main() {

    const int N = 20;

    int fN = N; // fake N - для "удаления" элементов из массива

    int a[N];

    int k = 3;

    srand(time(NULL));

    for (int i=0; i<N; i++) {

        a[i] = rand() % 101 - 50;

    }

    // циклический вывод

    for (int i=k; i<=(k+1+fN); i++) {

        cout << a[i%fN] << ' ';

    }

    cout << '\n';

    // сортировка вставкой

    for (int i=1; i<fN; i++) {

        int j = i;

        while ((j>0) && (a[j-1] < a[j])) {

            int tmp = a[j];

            a[j] = a[j-1];

            a[j-1] = tmp;

            j--;

        }

    }

    // удаление нечётных элементов

    int cor = 0;

    for (int i=0; i<N; i++) {

        int fi = i + cor;

        if (a[fi] % 2 != 0) {

            for (int j=fi; j<(fN-1); j++) {

                a[j] = a[j+1];

            }

            fN--;

            cor--;

        }

    }

    for (int i=k; i<=(k+1+fN); i++) {

        cout << a[i%fN] << ' ';

    }

    cout << '\n';

    return 0;

}

**Результаты:**

|  |  |  |
| --- | --- | --- |
| N | k | Результат выполнения кода |
| 5 | 0 |  |
| 20 | 0 |  |
| 20 | 3 |  |
| 20 | 10 |  |
| 25 | N\*N |  |

**Вывод:**

Я смог реализовать алгоритм, решающий поставленные задачи.