Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное учреждение высшего образования

«Пермский национальный исследовательский политехнический университет»

ПНИПУ

Отчет по лабораторной работе

**«Работа с одномерными массивами»**

Выполнила:

студентка группы РИС-23-2б

Виноградова Юлия Дмитриевна

Проверила:

доцент кафедры ИТАС

О.А. Полякова

2024 г.

**Создание алгоритма для выполнения задачи**

**Постановка задачи:**

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

**Словесный алгоритм:**

1. Изначально формируется большой массив и происходит его частичное заполнение.

Реализация первого и второго пунктов задания:

1. С использованием цикла for, все элементы массива mas просматриваются поочередно.

2. Для каждого элемента при текущем индексе k выводится значение элемента.

3. Индекс k уменьшается на 1.

4. Происходит проверка условия: если k становится -1, то устанавливается k равным n - 1 для продолжения итераций.

Объясняется работа условия if: цикл работает в режиме «кольцевого» обхода, просматривая элементы массива справа налево и возвращаясь к концу, если достигнуто начало массива. Цикл продолжается, пока k >= 0, этим гарантируется, что цикл будет продолжен с конца массива, начиная с элемента с индексом n – 1.

6. Данные действия повторяются n раз, чтобы обеспечить проход по всем элементам массива.

Реализация третьего пункта задания:

Использование алгоритма сортировки пузырьком:

1. Начинается сравнение первого и второго элементов списка.

2. Если первый элемент оказывается меньше второго, они обмениваются местами.

3. После этого происходит сравнение второго и третьего элементов и так далее, до конца списка.

4. После первого прохода самый маленький элемент оказывается в конце списка.

5. Шаги 1-4 повторяются для всех элементов до тех пор, пока список полностью не будет отсортирован.

Реализация четвертого пункта задания:

1. Начинается цикл, в котором индекс i увеличивается от 0 до n.

2. Проверяется, является ли текущий элемент массива с индексом i нечётным (имеет остаток от деления на 2 равный 1).

3. Если это так, переменная n уменьшается на 1 (что означает, что длина массива уменьшилась на 1).

4. Запускается второй цикл, в котором индекс j увеличивается от i до n.

5. В этом цикле каждый элемент массива сдвигается на одну позицию влево, начиная с элемента с индексом i.

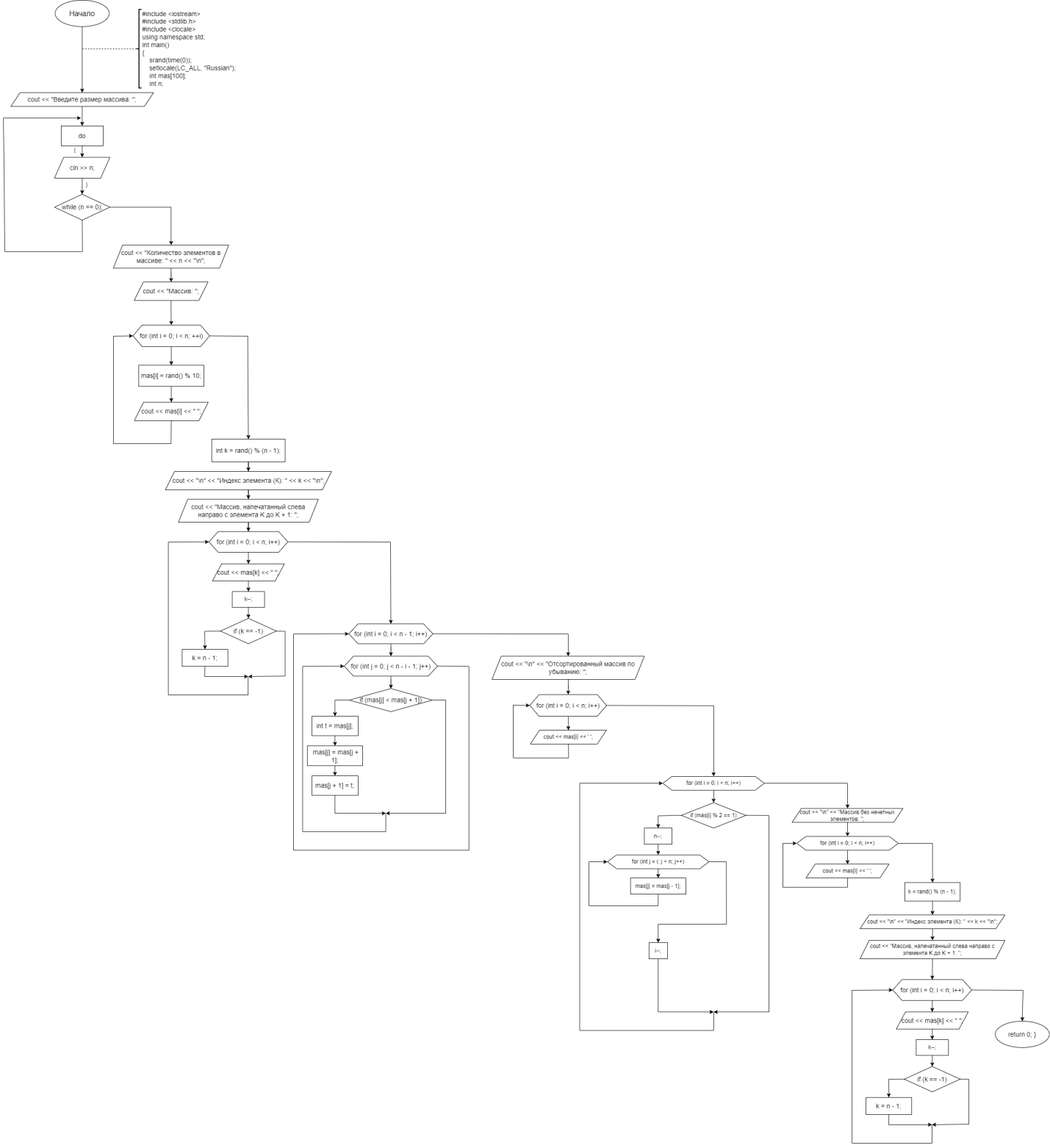
6. После этого переменная i уменьшается на 1, чтобы вернуться к предыдущему элементу в массиве.

7. Этот процесс повторяется для всех элементов массива до тех пор, пока итератор i не достигнет значения n.

Пятый пункт аналогичен первому и второму.

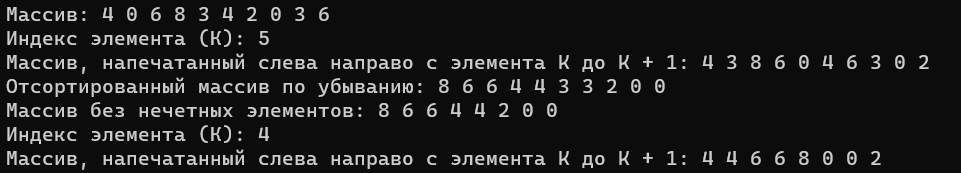
**Смысловые значения переменных:** *mas* – имя массива; *n* – размер массива; *k* – индекс элемента; *i, j* – переменные цикла.

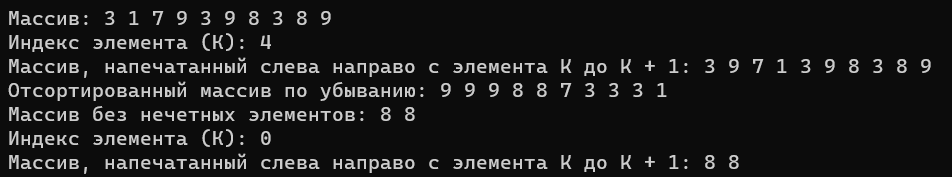
**Блок-схема**

**Код**

|  |
| --- |
| #include <iostream>  #include <stdlib.h>  #include <clocale>  using namespace std;  int main()  {  srand(time(0));  setlocale(LC\_ALL, "Russian");    int mas[100];  int n;  do {  cin >> n;  } while (n == 0);    cout << "Количество элементов в массиве: " << n << "\n";    cout << "Массив: ";    for (int i = 0; i < n; i++)  {  mas[i] = rand() % 10;  cout << mas[i] << " ";  }    int k = rand() % (n - 1);    cout << "\n" << "Индекс элемента (К): " << k << "\n";  cout << "Массив, напечатанный слева направо с элемента K до K + 1: ";    for (int i = 0; i < n; i++)  {  cout << mas[k] << " ";  k--;  if (k == -1) k = n - 1;  }  for (int i = 0; i < n - 1; i++)  {  for (int j = 0; j < n - i - 1; j++)  {  if (mas[j] < mas[j + 1])  {  int t = mas[j];  mas[j] = mas[j + 1];  mas[j + 1] = t;  }  }  }    cout << "\n" << "Отсортированный массив по убыванию: ";    for (int i = 0; i < n; i++)  {  cout << mas[i] << ' ';  }  for (int i = 0; i < n; i++)  {  if (mas[i] % 2 == 1)  {  n--;  for (int j = i; j < n; j++)  {  mas[j] = mas[j + 1];  }  i--;  }  }    cout << "\n" << "Массив без нечетных элементов: ";    for (int i = 0; i < n; i++)  {  cout << mas[i] << ' ';  }  k = rand() % (n - 1);    cout << "\n" << "Индекс элемента (К): " << k << "\n";  cout << "Массив, напечатанный слева направо с элемента K до K + 1: ";    for (int i = 0; i < n; i++)  {  cout << mas[k] << ' ';  k--;  if (k == -1) k = n - 1;  }    return 0;  } |

**Работа программы при разных значениях**

****

****

**Скриншот из гита**

**Вывод**

В результате выполнения лабораторной работы по работе с одномерным массивом были успешно реализованы алгоритмы однонаправленного кольца и удаления нечётных элементов. Алгоритм однонаправленного кольца позволяет эффективно просматривать и обрабатывать элементы массива в кольцевом режиме, начиная справа и возвращаясь к началу, что обеспечивает полный проход по всем элементам массива. Также был разработан алгоритм удаления нечётных элементов, который позволяет эффективно удалять элементы с нечётными индексами из массива, сокращая его длину на 1 для каждого удалённого элемента.

Результаты выполнения лабораторной работы показали, что реализованные алгоритмы позволяют эффективно работать с одномерным массивом, обеспечивая нужную функциональность и производительность при обработке массива данных.