

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
Кафедра вычислительной техники

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ № 8

Одномерные массивы
Вариант № 19

Преподаватель

подпись, дата

Пушкарев К. В.

Студент КИ18-096, 031831293

подпись

18.11.2018

Овсянников В.А.

Красноярск 2018

1 Дополнительное упражнение № 2 (вариант 9)

Разработать, отладить, продемонстрировать и защитить преподавателю графическую схему алгоритма и составить программу, использующую динамическое выделение памяти под массив, для решения следующей задачи: из заданного одномерного массива $A[N]$ формирует новый массив B по правилам: заменить все нулевые элементы, предшествующие первому отрицательному, единицей. Если массив сформировать невозможно, вывести соответствующее сообщение.

2 Цели работы

1. получить навыки построения графических схем алгоритма и программ обработки одномерных массивов;
2. Познакомиться с механизмом динамического выделения памяти.

3 Подход к решению задачи

Согласно условию задачи, требуется разработать графическую схему алгоритма и программу, которая использует динамическое выделение памяти под массив, для формирования нового массива B из заданного одномерного массива $A[N]$, заменяя в новом массиве единицей все нулевые элементы, предшествующие первому. Для этого создадим переменные i , j и N типа int и одномерный массив A размерностью N . Значение N пользователь будет вводить с клавиатуры. Создадим функцию `find_negative_el()`, которая будет осуществлять поиск первого отрицательного элемента и возвращать его индекс. Добавим проверку на возможность сформировать массив, если первому отрицательному элементу не предшествуют другие элементы или если отрицательных элементов нет вообще на экран будет выведено сообщение о том, что массив сформировать невозможно, иначе будет сформирован новый массив B и выведен на экран.

4 Графическая схема алгоритма

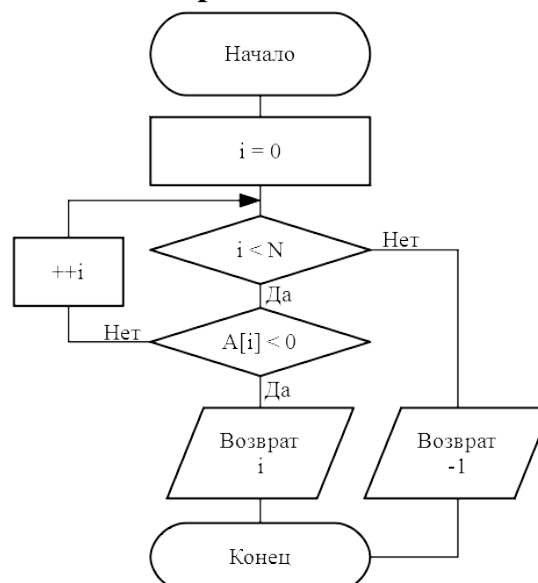


Рисунок 1 – Функция `int find_negative_el(int N, int A[])`.

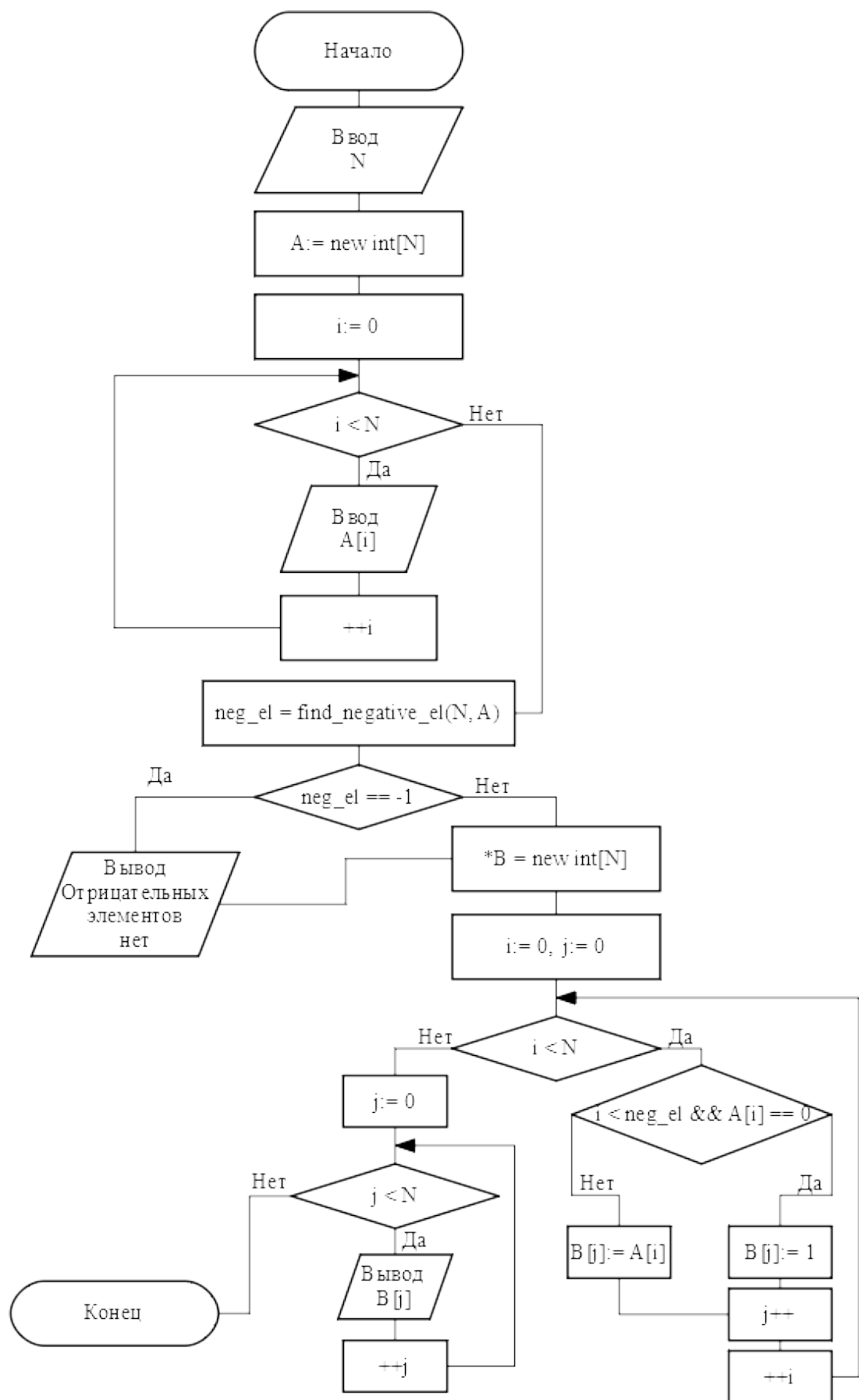


Рисунок 2 – функция *main*

5 Код программы

```
1 #include "pch.h"
2 #include <iostream>
3 #include <locale>
4
5 using namespace std;
6
7 int find_negative_el(int N, int A[]);
8
9 int main(){
10     setlocale(LC_ALL, "rus");
11
12     int N, i, j;
13
14     cout << "Введите размерность массива: ";
15     cin >> N;
16
17     int *A = new int[N];
18
19     cout << "Введите массив A: ";
20     for (i = 0; i < N; i++) {
21         cin >> A[i];
22     }
23
24     int neg_el = find_negative_el(N, A);
25
26     if (neg_el == -1) {
27         cout << "\nОтрицательных элементов нет\n";
28     }
29
30     int *B = new int[N];
31
32     for (i = 0, j = 0; i < N; i++) {
33         if (i < neg_el && A[i] == 0) {
34             B[j] = 1;
35         }
36         else B[j] = A[i];
37         j++;
38     }
39     cout << "\nМассив B = {";
40     for (j = 0; j < N; j++) {
41         cout << " " << B[j];
42     }
43     cout << " }" << endl;
44     return 0;
45 }
46
47 int find_negative_el(int N, int A[]) {
48     int i;
49     for (i = 0; i < N; i++) {
50         if (A[i] < 0) {
51             return i;
52         }
53     }
54     return -1;
55 }
```

6 Результаты выполнения экспериментальной части работы

Результаты запуска программы с различными входными значениями приведены в таблице 1.

Таблица 1 – Работа программы на тестовых наборах данных

Входные данные	Результат
$N = 5,$ $A = \{0, 0, 0, -2, 0\}$	Массив $B = \{1, 1, 1, -2, 0\}$
$N = 3$ $A = \{-1, -2, -3, -4, -5\}$	Массив $B = \{-1, -2, -3, -4, -5\}$
$N = 3$ $A = \{1, 0, 1\}$	Отрицательных элементов нет Массив $B = \{1, 0, 1\}$