МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Воронежский государственный технический университет»

**Факультет** экономики, менеджмента и информационных технологий

**Кафедра** систем управления и информационных технологий в строительстве

**Отчет по лабораторной работе**

Тема: **«Работа с массивами данных»**

По дисциплине: Основы программирования и алгоритмизации

Выполнил студент: Гладнева Евгения Геннадьевна

Группа: бИСТ-222

Руководитель: доцент, к.т.н. Курипта О. В.

Работа защищена « » \_\_\_\_\_\_\_\_\_ 2022г.

С оценкой \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Воронеж 2022

**Постановка задачи**

**Условие задачи:** В одномерном массиве, состоящим из n вводимых с клавиатуры положительных и отрицательных целых чисел, вычислить сумму всех элементов массива, расположенных до последнего положительного элемента.

**Исходные данные:**

n.

**Алгоритм решения:**

1. В цикле запрашивать у пользователя значение очередного элемента.
2. Если элемент положительный, прибавить сумму предыдущих элементов (начиная с положительного) к реальной сумме. Обнулить возможную сумму.
3. Увеличить возможную сумму на значение введённого элемента.

**Контрольный пример**:

Входные данные: 1, 2, -4, 5, -6, 7, 3, -7, -8, -9

Результат: 5

**Словесный алгоритм**

**Алгоритм представлен пошаговой детализацией:**

**Шаг 1:** установка директивы препроцессора для функции scanf, подключение необходимых заголовочных файлов

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <locale.h>

#include <math.h>

**Шаг 2:** определение размера массива через константу

#define N 10

**Шаг 3:** объявление функции main

void main(void) {};

**Шаг 4:** установка локализации

setlocale(LC\_ALL, "RUS");

**Шаг 5:** объявление переменных

int A[N];

int x;

int sum = 0, possible\_sum = 0;

**Шаг 5:** ввод значений пользователем

for (int i = 0; i < N; i++)

{

printf("A[%d] = ", i);

scanf("%d", &x);

A[i] = x;

…

}

**Шаг 6:** изменение сумм в зависимости от условия

if (x > 0) {

sum += possible\_sum;

possible\_sum = 0;

}

possible\_sum += x;

**Шаг 7:** вывод результата

printf("Сумма элементов до последнего положительного = %d", sum);

**Блок – схема программы**

Блок – схема программы, которая рассчитывает сумму всех элементов массива, расположенных до последнего положительного элемента, представлена на рисунке 1.

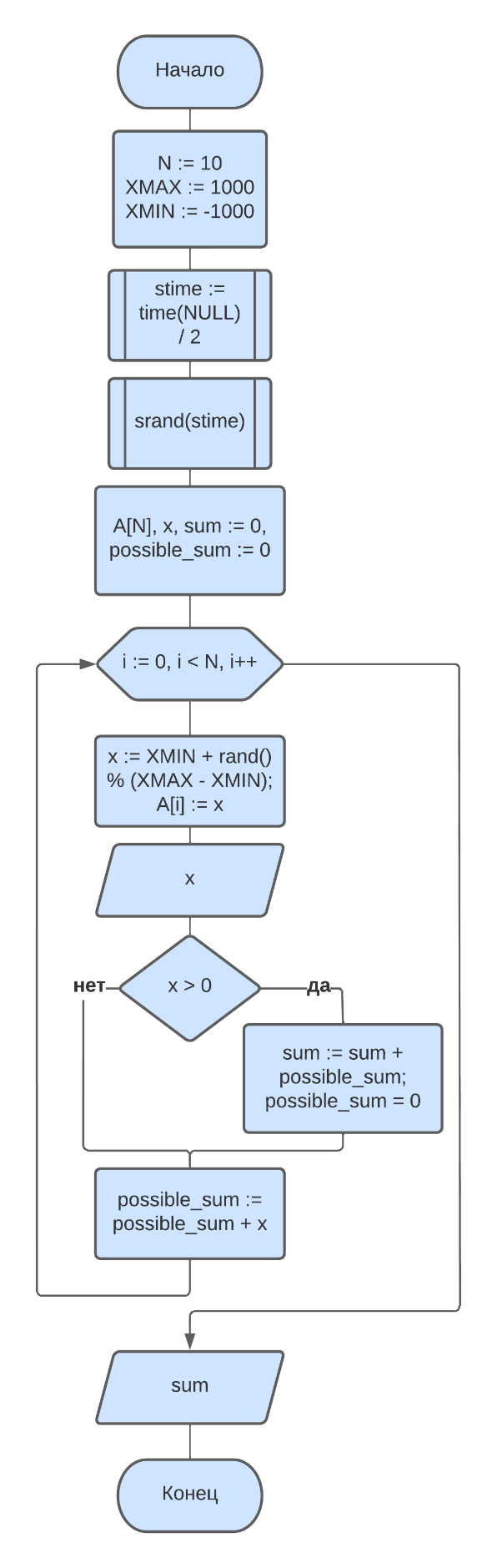


Рисунок 1 — Блок-схема программы

**Результат работы программы**

На рисунке 2 представлено окно ввода данных

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2 — Окно ввода данных

На рисунке 3 представлена работа программы с данными из контрольного примера.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3 — Окно с выполненной программой

**Код программы**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <locale.h>

#include <math.h>

#define N 10

void main()

{

setlocale(LC\_ALL, "RUS");

int A[N];

int x;

int sum = 0, possible\_sum = 0;

for (int i = 0; i < N; i++)

{

printf("A[%d] = ", i);

scanf("%d", &x);

A[i] = x;

if (x > 0) {

sum += possible\_sum;

possible\_sum = 0;

}

possible\_sum += x;

}

printf("Сумма элементов до последнего положительного = %d", sum);

}