Федеральное государственное автономное

образовательное учреждение

высшего профессионального образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра вычислительной техники

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №9**

(вариант 7)

Тема работы

Массивы и указатели

Преподаватель \_\_\_\_\_\_\_\_\_\_\_

Покидышева Л. И.

подпись, дата

Студент гр. КИ-15 08 Б, Войченко В.В.

\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Красноярск 2015

**Цель работы**

•научиться обрабатывать одномерные массивы с использованием указателей;

•познакомиться с механизмом

динамического выделения памяти;

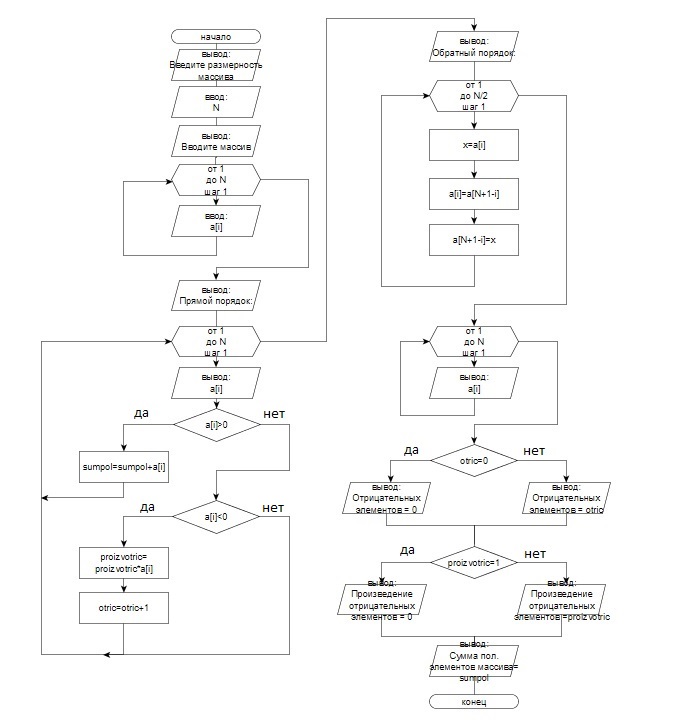
**Задание 1**

1. Разработать графическую схему алгоритма и два варианта программы (первый вариант – со статическим выделением памяти под массив, второй – с динамическим), которая получает с клавиатуры натуральное число N - размерность массива, а также N целых чисел – элементы этого массива, выводит этот массив на экран монитора в прямом и обратном порядке, а также определяет сумму положительных, произведение и количество отрицательных элементов.

2. Выполнить пункты 2-4 из упр.№1. Провести трассировку программы с помощью встроенного в среду программирования отладчика, анализируя значения переменных после каждого оператора присваивания.

3. Выполнить несколько запусков программы для заранее подготовленных тестовых примеров для проверки работоспособности всех ветвей программы. Полученные результаты проанализировать.

**Схема алгоритма программы 1**



**Исходный код программы 1 (Вариант 1: динамическое выделение массива)**

#include <stdio.h>

#include <iostream>

#include <clocale>

#include <cstdlib>

using namespace std;

int vvodint ()

{

int n;

cout<<"Введите размерность массива: \n\n";

cin>>n;

cout<<"\n\n";

return n;

}

//------------------------------------

void vvod (int \*ptr\_a, int n)

{

cout<<"Вводите массив: ";

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

cin>>\*(ptr\_a+i);

cout<<"\n\n";

}

//------------------------------------

void vivod (int \*ptr\_a, int n)

{

cout<<"Прямой порядок: ";

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

{

cout<<\*(ptr\_a+i)<<" ";

}

cout<<"\n\n";

}

//------------------------------------

void vivodobr (int \*ptr\_a, int n)

{

int x;

cout<<"Обратный порядок: ";

if (n==2)

{

x=\*ptr\_a;

\*ptr\_a=\*(ptr\_a+1);

\*(ptr\_a+1)=x;

cout<<\*ptr\_a<<" "<<\*(ptr\_a+1)<<endl;

}

else

{

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

cout<<\*(((ptr\_a+n)-1)-i)<<" ";

}

cout<<"\n\n";

}

//------------------------------------

int main ()

{

int n;

int otric=0, sumpol=0, proizvotric=1;

setlocale (LC\_ALL, "russian");

n=vvodint (); //ввод числа

int \*a= new int [n];

int \*ptr\_a=a;

vvod (a, n); //ввод массива

vivod (a, n); //вывод массива

vivodobr (a, n); //вывод обратного массива

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

{

if (\*(ptr\_a+i)>0)

sumpol+=\*(ptr\_a+i);

if (\*(ptr\_a+i)<0)

{

proizvotric\*=\*(ptr\_a+i);

otric++;

}

}

(otric==0 ? cout<<"Количество отрицательных элементов массива А= 0 \n\n" :

cout<<" Количество отрицательных элементов массива А = "<<otric<<"\n\n");

(proizvotric==1 ? cout<<"Произведение отрицательных элементов массива А= 0 \n\n" :

cout<<" Произведение отрицательных элементов массива А = "<<proizvotric<<"\n\n");

cout<<"Сумма положительных элементов массива А= "<<sumpol<<"\n\n";

system ("pause");

delete [] a;

return 0;

}

**Исходный код программы 1 (Вариант 2: статическое выделение массива)**

#include <stdio.h>

#include <iostream>

#include <clocale>

#include <cstdlib>

using namespace std;

int vvodint ()

{

int n;

cout<<"Введите размерность массива: (n<=30)\n\n";

cin>>n;

while ((n>30) || (n<0))

{

cout<<"Выход за интервал размерности массива\nповторите ввод!\n";

cin>>n;

}

cout<<"\n\n";

return n;

}

//------------------------------------

void vvod (int \*ptr\_a, int n)

{

cout<<"Вводите массив: ";

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

cin>>\*(ptr\_a+i);

cout<<"\n\n";

}

//------------------------------------

void vivod (int \*ptr\_a, int n)

{

cout<<"Прямой порядок: ";

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

{

cout<<\*(ptr\_a+i)<<" ";

}

cout<<"\n\n";

}

//------------------------------------

void vivodobr (int \*ptr\_a, int n)

{

int x;

cout<<"Обратный порядок: ";

if (n==2)

{

x=\*ptr\_a;

\*ptr\_a=\*(ptr\_a+1);

\*(ptr\_a+1)=x;

cout<<\*ptr\_a<<" "<<\*(ptr\_a+1)<<endl;

}

else

{

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

cout<<\*(((ptr\_a+n)-1)-i)<<" ";

}

cout<<"\n\n";

}

//------------------------------------

int main ()

{

int otric=0, sumpol=0, proizvotric=1, n;

setlocale (LC\_ALL, "russian");

n=vvodint (); //ввод числа

int a[n];

vvod (a, n); //ввод массива

int \*ptr\_a=a;

vivod (a, n); //вывод массива

vivodobr (a, n); //вывод массива наоборот

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

{

if (\*(ptr\_a+i)>0)

sumpol+=\*(ptr\_a+i);

if (\*(ptr\_a+i)<0)

{

proizvotric\*=\*(ptr\_a+i);

otric++;

}

}

(otric==0 ? cout<<"Êîëè÷åñòâî îòðèöàòåëüíûõ ýëåìåíòîâ ìàññèâà À= 0 \n\n" :

cout<<"Êîëè÷åñòâî îòðèöàòåëüíûõ ýëåìåíòîâ ìàññèâà À= "<<otric<<"\n\n");

(proizvotric==1 ? cout<<"Ïðîèçâåäåíèå îòðèöàòåëüíûõ ýëåìåíòîâ ìàññèâà À= 0 \n\n" :

cout<<"Ïðîèçâåäåíèå îòðèöàòåëüíûõ ýëåìåíòîâ ìàññèâà À= "<<proizvotric<<"\n\n");

cout<<"Ñóììà ïîëîæèòåëüíûõ ýëåìåíòîâ ìàññèâà À= "<<sumpol<<"\n\n";

system ("pause");

return 0;

}

**Тестовые наборы данных для программы 1:**

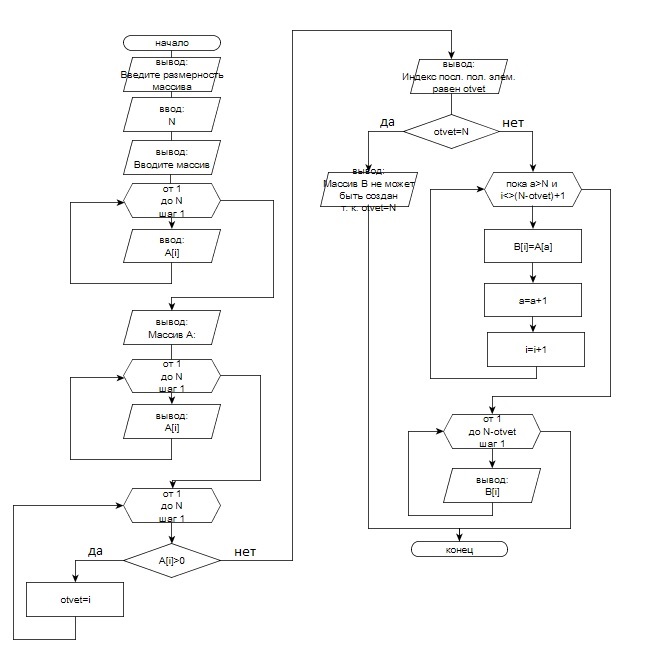
|  |  |
| --- | --- |
| Входные данные | Результат |
| Размерность массива А: 5  Массив А: 1 2 -4 -5 -6 | Массив: 1 2 -4 -5 -6  Прямой порядок: 1 2 -4 -5 -6  Обратный порядок: -6 -5 -4 2 1  Кол-во отриц. эл-ов массива А= 3  Произв-е отриц. эл-ов массива А= -120  Сумма пол. эл-ов массива А= 3 |
| Размерность массива А: 3  Массив А: 45 46 47 | Массив: 45 46 47  Прямой порядок: 45 46 47  Обратный порядок: 47 46 45  Кол-во отриц. эл-ов массива А= 0  Произв-е отриц. эл-ов массива А= 0  Сумма пол. эл-ов массива А= 138 |
| Размерность массива А: 1  Массив А: 58 | Массив: 58  Прямой порядок: 58  Обратный порядок:58  Кол-во отриц. эл-ов массива А= 0  Произв-е отриц. эл-ов массива А= 0  Сумма пол. эл-ов массива А= 58 |
| Размерность массива А: 2  Массив А: 16 -88 | Массив: 16 -88  Прямой порядок: 16 -88  Обратный порядок: -88 16  Кол-во отриц. эл-ов массива А= 1  Произв-е отриц. эл-ов массива А= -88  Сумма пол. эл-ов массива А= 16 |

**Задание 2**

1) 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, указанным в таблице №8.2. Если массив сформировать не возможно, вывести соответствующее сообщение.

Условие: сформировать массив В из массива А из элементов заданного массива, расположенных после последнего положительного элемента; найти в массиве А индекс последнего положительного элемента в массиве А и оформить как функцию.

**Схема алгоритма программы 2**



**Исходный код программы 2**

#include <iostream>

#include <clocale>

#include <cstdlib>

using namespace std;

//----------------------------------------------------

void view (int \*ptr\_a, int n)

{

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

cout<<\*(ptr\_a+i)<<" ";

cout<<"\n\n";

}

//----------------------------------------------------

void vvodmass (int \*ptr\_a, int n)

{

cout<<"Введите массив:";

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

{

cin>>\*(ptr\_a+i);

}

cout<<"\n\n";

}

//----------------------------------------------------

int finder (int \*ptr\_a, int n)

{

int otvet=0;

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

if (\*(ptr\_a+i)>0)

otvet=i;

otvet++;

cout<<"Индекс последнего положительного элемента: "<<otvet;

cout<<"\n\n";

return otvet;

}

//----------------------------------------------------

int main ()

{

int n, p, i;

int otvet=0;

setlocale (LC\_ALL, "russian");

cout<<"Введите размерность массива: ";

cin>>n ;

int \*a=new int [n];

int \*ptr\_a=a;

vvodmass (a, n); //ввод массива

cout<<"Массив А: ";

view (a, n); //вывод массива

otvet=finder (a, n); //поиск последнего пол. элемента массива

if (otvet==n)

cout<<"Массив В не может быть создан т.к. последний положительный элемент в массиве А \n стоит в конце массива\n";

else

{

cout<<"Массив В: ";

for (i=otvet; i<n; i++)

cout<<\*(ptr\_a+i)<<" ";

}

cout<<"\n";

delete [] a;

system ("pause");

return 0;

}

**Тестовые наборы данных для программы 2:**

|  |  |
| --- | --- |
| Входные данные | Результат |
| Размерность массива А: 5  Массив А: 1 2 -4 -5 -6 | Индекс последнего пол. эл.: 2  Массив В: -4 -5 -6 |
| Размерность массива А: 3  Массив А: 45 46 47 | Индекс последнего пол. эл.: 3  Массив В не может быть создан т.к. последний пол. эл. в массиве А стоит в конце массива А |
| Размерность массива А: 1  Массив А: 58 | Индекс последнего пол. эл.: 1  Массив В не может быть создан т.к. последний пол. эл. в массиве А стоит в конце массива А |
| Размерность массива А: 2  Массив А: 16 -88 | Индекс последнего пол. эл.: 1  Массив В: -88 |