МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования

«Белорусский государственный университет транспорта»

Кафедра «Информационно-управляющие системы и технологии»

Отчет  
по практическим работам  
по дисциплине «Основы алгоритмизации и программирования»

Выполнил Проверил

студент группы ГИ-11 ст. преп. каф. ИУСиТ

Зайцев И. С. Голдобина Т. А.

Гомель, 2019

Содержание

[Практическая работа №11 Разработка алгоритма, составление, отладка и выполнение программы обработки одномерного массива 3](#_Toc26545793)

[Задания 3](#_Toc26545794)

[Контрольные вопросы 7](#_Toc26545795)

# Практическая работа №11 Разработка алгоритма, составление, отладка и выполнение программы обработки одномерного массива

### Цель

Разработать алгоритм и составить, отладить и выполнить программы обработки одномерного массива.

## Задания

### Задание 1

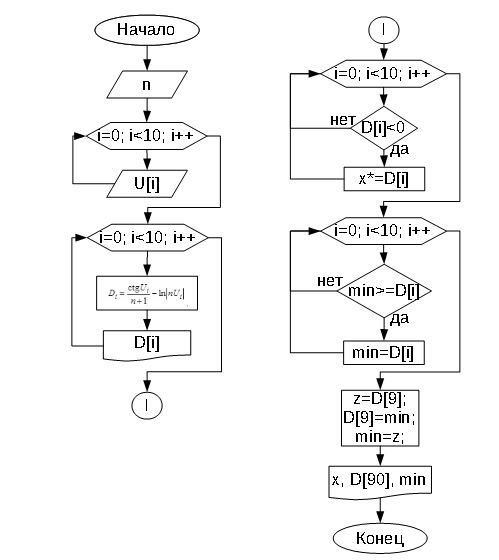
### Условие

Ввести одномерный массив *U* из десяти элементов и число *n*. Для каждого элемента массива вычислить . Вывести полученный массив. Определить произведение отрицательных элементов массива *D*. Поменять местами значения последнего и минимального элементов *D*.

Вывод результатов в процессе отладки программы осуществлять на экран поэлементно в цикле.

### Блок-схема алгоритма

### Программный код

  
Рисунок 1 – Блок-схема к заданию 1

#include <stdio.h>  
#include <stdlib.h>  
#include <math.h>   
  
int main()  
{  
int U[10], min=0;  
int i, D[10], n,x=1,z;  
printf ("Enter n:");  
scanf ("%d", &n);  
for (i=0; i<10; i++)  
{  
printf("U[%d] = ", i);  
scanf("%d", &U[i]);  
}  
for (i=0; i<10; i++)  
{D[i]=(1/tan(U[i]))/(n+1)-log(abs(n\*U[i]));  
printf("D[%d]=%d\n",i, D[i]);}  
for (i=0; i<10; i++) {  
if (D[i]<0)  
{  
x\*=D[i];  
}  
}  
for (i=0; i<10; i++){  
if (min>=D[i]) {min=D[i];}  
}  
z=D[9];  
D[9]=min;  
min=z;  
printf ("Multiplication=%d\nD[9]=%d\nmin=%d\n",x, D[9], min);  
  
  
  
return 0;}

  
Рисунок 2 – Решение задания 1

### Задание 2

### Условие

### Составить программу, в которой генерируется одномерный массив вещественных чисел, равномерно распределенных на заданном диапазоне. Границы диапазона ввести с клавиатуры или определить в виде глобальных констант. Найти разность между суммой положительных элементов и суммой модулей отрицательных элементов.

### Программный код

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

float s, n, G[10], Sa=0, Sp=0, m=1, min, max;

int t=0;

printf("enter minimal element: ");

scanf("%f", &min);

printf("enter maximal element: ");

scanf("%f", &max);

s=(max-min)/9;

for(n=min; n<=max; n+=s){

G[t]=n;

printf("g[%d]=%.2f\n", t, G[t]);

t++;

}

for(t=0; t<10; ++t){

if(G[t]>0) {

Sp+=G[t];

}

else if(G[t]<0) {

Sa+=fabs(G[t]);

}

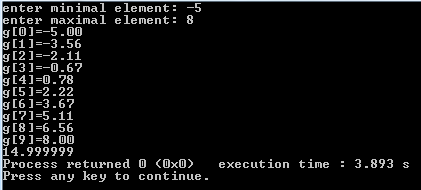
}

Sp-=Sa;

printf("%f", Sp);

return 0;

}

  
Рисунок 3 – Решение задания 2

## Контрольные вопросы

1. Как объявить одномерный массив из 10 целых чисел?

int a[10];

1. Алгоритм ввода элементов одномерного массива

for (i=0; i<10; i++)

{

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

scanf("%d", &U[i]);

}

1. Вывод элементов одномерного массива на С.

for (i=0; i<10; i++)

{

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

1. Алгоритм нахождения минимального элемента одномерного массива.

— Инициализация массива, переменной, хранящей минимальное значение.

— Заполнение массива случайными числами при помощи цикла и функции, возвращающей случайные числа.

— Вывод массива.

— Сравнение каждого элемента массива: Если элемент меньше переменной с минимальным значением, то значение записывается в переменную.

— Вывод переменной с минимальным элементом.

1. Нахождение максимального элемента одномерного массива на C.

int main()

{

int m[5]={1, -1, 0, 4, 2};

int max = m[0];

int i;

for(i=0; i<5; i++0)

{

if(m[i]>max)

{

max=m[i];

}

}

printf(“max=%d\n”,max);

return 0;}

### Выводы по работе

Разработал алгоритмы, составил, отладил и выполнил программы обработки одномерного массива.