|  |  |  |
| --- | --- | --- |
| Министерство образования Республики Беларусь  Учреждение образования  Белорусский Государственный Университет Информатики и Радиоэлектроники | | |
| Факультет компьютерных систем и сетей | | |
| Кафедра электронных вычислительных средств | | |
| **ОТЧЕТ**  по лабораторной работе №3  «**Операторы цикла**» | | |
| Выполнил  студ. Гр. 250504 (пг1)  Лагодич И.Р. |  | Проверил  ассист. каф. ЭВМ  Скиба И. Г. |
| Минск 2022 | | |

* **1 ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ**

Цель работы – научиться разрабатывать алгоритмы для работы с одномерными массивами и писать код на языке Си по составленному алгоритму. Лабораторная работа включает в себя 3 задачи для выполнения. Задачи решаются последовательно.

* **2 Результаты выполнения И ЗАДАЧИ лабораторной работы**

Задача 1

В одномерном массиве, состоящем из n вещественных элементов, вычислить:  
- номер максимального по модулю элемента массива;  
- сумму элементов массива, расположенных после первого положительного элемента.

Программа:

#include <stdio.h>

#include <stdlib.h>

#include <ctime>

int main()

{

int a[100], sum=0, n, choose, number, max;

printf("Enter the size of array: ");

scanf\_s("%d", &n);

srand(time(NULL));

while (n < 1 || n>100)

{

printf("Wrong input.\nEnter the size of array: ");

scanf\_s("%d", &n);

}

printf("1.User filling\n2.Random filling\n");

printf("Choose the type of array filling: ");

scanf\_s("%d", &choose);

while (choose != 1 && choose != 2)

{

printf("Wrong input.\nChoose the type of array filling: ");

scanf\_s("%d", &choose);

}

switch (choose)

{

case 1:

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

{

printf("Element number %d: ", (i + 1));

scanf\_s("%d", &a[i]);

}

break;

case 2:

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

{

a[i] = rand()%201-100;

printf("Element number %d: %d\n", (i + 1),a[i]);

}

break;

}

printf("Array: ");

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

printf("%d ", a[i]);

max = a[0];

number = 1;

for (int i=1;i<n; i++)

if (abs(a[i])>abs(max))

{

max = a[i];

number = i+1;

}

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

if (a[i] > 0)

{

while (i < n)

{

sum = sum + a[i];

i++;

}

break;

}

printf("\nNumber of the maximum modulo element of the array is: %d\nThe sum of the array elements after the first positive element: %d",number,sum);

return 0;

}

Результат:

Enter the size of array: 10

1.User filling

2.Random filling

Choose the type of array filling: 2

Element number 1: 65

Element number 2: -50

Element number 3: 50

Element number 4: 16

Element number 5: 42

Element number 6: -27

Element number 7: 81

Element number 8: -55

Element number 9: 91

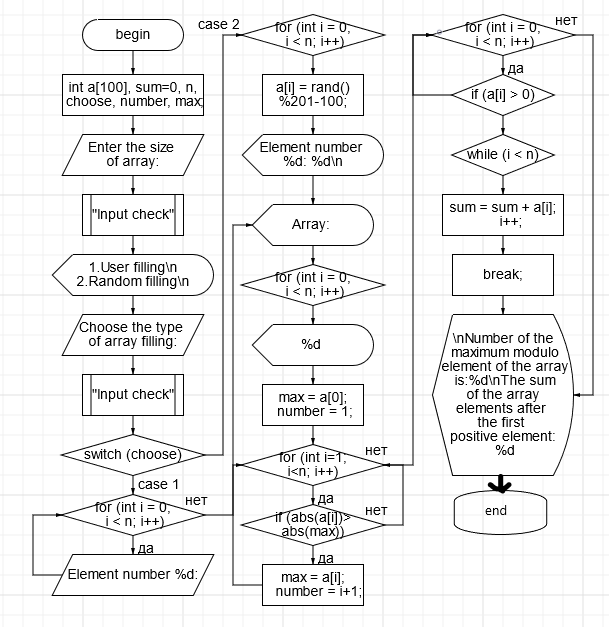
Element number 10: 78

Array: 65 -50 50 16 42 -27 81 -55 91 78

Number of the maximum modulo element of the array is: 9

The sum of the array elements after the first positive element: 291

Блок-схема:



Задача 2

В массиве из n элементов удалить каждый k-ый элемент

Программа:

#include <stdio.h>

#include <stdlib.h>

#include <ctime>

int main()

{

int a[100], n, k, choose, gr, zn;

printf("Enter the size of array: ");

scanf\_s("%d", &n);

while (n < 1 || n>100)

{

printf("Wrong input.\nEnter the size of array: ");

scanf\_s("%d", &n);

}

printf("Enter k: ");

scanf\_s("%d", &k);

while (k<1 || k>n)

{

printf("Wrong input. \nEnter k: ");

scanf\_s("%d", &k);

}

srand(time(NULL));

while (n < 1 || n>100)

{

printf("Wrong input.\nEnter the size of array: ");

scanf\_s("%d", &n);

}

printf("1.User filling\n2.Random filling\n");

printf("Choose the type of array filling: ");

scanf\_s("%d", &choose);

while (choose != 1 && choose != 2)

{

printf("Wrong input.\nChoose the type of array filling: ");

scanf\_s("%d", &choose);

}

switch (choose)

{

case 1:

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

{

printf("Element number %d: ", (i + 1));

scanf\_s("%d", &a[i]);

}

break;

case 2:

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

{

a[i] = rand() % 201 - 100;

printf("Element number %d: %d\n", (i + 1), a[i]);

}

break;

}

printf("Array: ");

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

printf("%d ", a[i]);

for (int i = k; i < n;)

{

zn = i;

while(i < n)

{

a[i-1] = a[i];

i++;

}

i = zn + k-1;

n--;

}

if (n % k == 0)

n--;

printf("Changed array: ");

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

printf("%d ", a[i]);

return 0;

}

Результат:

Enter the size of array: 5

Enter k: 2

1.User filling

2.Random filling

Choose the type of array filling: 2

Element number 1: 78

Element number 2: 86

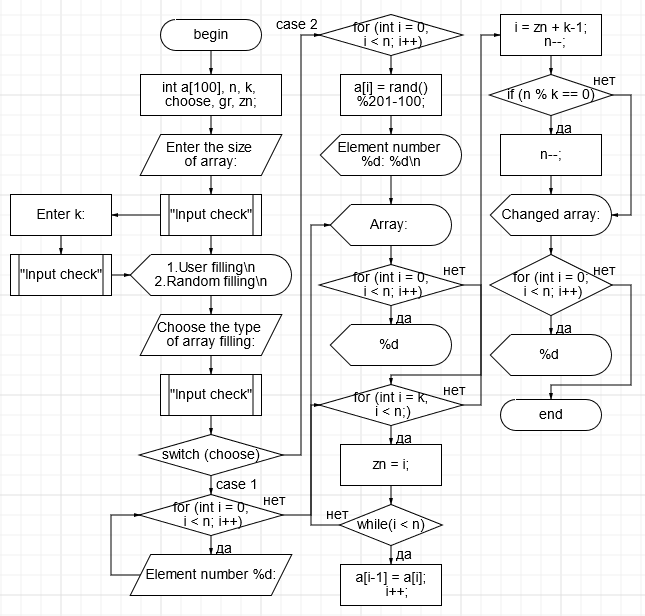
Element number 3: -21

Element number 4: -35

Element number 5: -26

Array: 78 86 -21 -35 -26 Changed array: 78 -21 -26

Блок-схема:



Задача 3:

Для арифметических операций с большими числами, которые не могут быть представлены в памяти компьютера, используется следующий прием. Каждая цифра таких чисел записывается в отдельный элемент массива, и необходимые операции проводятся с элементами массива цифр. Составить программу, выполняющую вычитание 30-значных чисел.

Программа:

#include <stdio.h>

#include <stdlib.h>

#include <ctime>

int main()

{

int a[100], b[100], c[100], choose, n = 3, zn=0;

srand(time(NULL));

printf("1.User filling\n2.Random filling\n");

printf("Choose the type of array filling: ");

scanf\_s("%d", &choose);

while (choose != 1 && choose != 2)

{

printf("Wrong input.\nChoose the type of array filling: ");

scanf\_s("%d", &choose);

}

switch (choose)

{

case 1:

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

{

printf("1 array element number %d: ", (i + 1));

scanf\_s("%d", &a[i]);

while (a[i] < 0 || a[i]>9)

{

printf("Wrong input.\n1 array element number %d: ", (i + 1));

scanf\_s("%d", &a[i]);

}

}

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

{

printf("2 array element number %d: ", (i + 1));

scanf\_s("%d", &b[i]);

while (b[i] < 0 || b[i]>9)

{

printf("Wrong input.\n2 array element number %d: ", (i + 1));

scanf\_s("%d", &b[i]);

}

}

break;

case 2:

a[0] = rand() % 9 + 1;

b[0] = rand() % 9 + 1;

printf("1 array element number 1: %d 2 array element number 1: %d\n", a[0], b[0]);

for (int i = 1; i < n; i++)

{

a[i] = rand() % 10;

b[i] = rand() % 10;

printf("1 array element number %d: %d 2 array element number %d: %d\n", (i + 1), a[i], (i+1), b[i]);

}

break;

}

printf("Number 1: ");

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

printf("%d", a[i]);

printf("\nNumber 2: ");

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

printf("%d", b[i]);

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

c[i] = 0;

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

{

if (a[i] < b[i] && a[i]!=b[i])

{

zn = 1;

break;

}

}

if (zn == 0)

{

for (int i = n - 1; i > 0; i--)

{

c[i] = c[i] + a[i] - b[i];

if (c[i] < 0)

{

c[i] = c[i] + 10;

c[i - 1]--;

}

}

c[0] = c[0] + a[0] - b[0];

}

else

{

for (int i = n - 1; i > 0; i--)

{

c[i] = c[i] + b[i] - a[i];

if (c[i] < 0)

{

c[i] = c[i] + 10;

c[i - 1]--;

}

}

c[0] = (c[0] + b[0] - a[0])\*(-1);

}

printf("\nEqual: ");

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

printf("%d", c[i]);

}

Результат:

1.User filling

2.Random filling

Choose the type of array filling: 2

1 array element number 1: 1 2 array element number 1: 3

1 array element number 2: 1 2 array element number 2: 4

1 array element number 3: 9 2 array element number 3: 4

1 array element number 4: 7 2 array element number 4: 6

1 array element number 5: 4 2 array element number 5: 9

1 array element number 6: 2 2 array element number 6: 6

1 array element number 7: 1 2 array element number 7: 6

1 array element number 8: 4 2 array element number 8: 1

1 array element number 9: 7 2 array element number 9: 3

1 array element number 10: 8 2 array element number 10: 9

1 array element number 11: 8 2 array element number 11: 8

1 array element number 12: 8 2 array element number 12: 5

1 array element number 13: 0 2 array element number 13: 9

1 array element number 14: 3 2 array element number 14: 9

1 array element number 15: 5 2 array element number 15: 8

1 array element number 16: 7 2 array element number 16: 5

1 array element number 17: 5 2 array element number 17: 5

1 array element number 18: 3 2 array element number 18: 6

1 array element number 19: 1 2 array element number 19: 6

1 array element number 20: 0 2 array element number 20: 3

1 array element number 21: 9 2 array element number 21: 8

1 array element number 22: 7 2 array element number 22: 2

1 array element number 23: 3 2 array element number 23: 3

1 array element number 24: 4 2 array element number 24: 2

1 array element number 25: 3 2 array element number 25: 0

1 array element number 26: 4 2 array element number 26: 8

1 array element number 27: 9 2 array element number 27: 3

1 array element number 28: 9 2 array element number 28: 6

1 array element number 29: 3 2 array element number 29: 7

1 array element number 30: 1 2 array element number 30: 9

Number 1: 119742147888035753109734349931

Number 2: 344696613985998556638232083679

Equal: -224954466097962803528497733748

Блок-схема:

