## Практические домашние задания для Урока 6 Модуля 2

1. Запросите ввод с клавиатуры количество элементов массива. Объявите целочисленный массив и выделите необходимый объем памяти для хранения элементов массива. Запросите ввод значений элементов массива с клавиатуры. Отсортируйте массив по возрастанию и выведите элементы отсортированного массива на экран через пробел.
2. Запросите ввод с клавиатуры количество элементов массива. Объявите целочисленный массив и выделите необходимый объем памяти для хранения элементов массива. Запросите ввод значений элементов массива с клавиатуры. Найдите среднее арифметическое чётных элементов массива и нечётных элементов массива. Если среднее арифметическое чётных элементов массива больше среднего арифметического нечётных элементов массива, то заменить значения всех нечётных элементов массива на ближайшее чётное число в большую сторону, иначе заменить все значения чётных элементов массива на ближайшее нечетное число в меньшую сторону. Если среднее арифметическое чётных элементов и нечётных элементов массива равны, заменить значения всех элементов массива на 0. Вывести элементы массив на экран в строку через пробел.
3. Объявите целочисленный массив, количество элементов заранее не известно. Организуйте ввод элементов массива, пока не будет введен 0. Создайте массив действительных чисел с таким же количеством элементов, заполните его значениями — разность значения соответствующего элемента первого массива и среднего арифметического всех элементов первого массива. Выведите на экран в строку элементы второго массива в формате 8 знакомест под всё число и 2 знакоместа под дробную часть.

## Решения

1.

#include <stdio.h>

#include <malloc.h>

int main()

{

int \*a;

int i, n,sum=0;

printf("Введите размер массива: ");

scanf("%d", &n);

a = (int\*)malloc(n \* sizeof(int));

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

{

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

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

}

int min, temp;

for (int i = 0; i < n - 1; i++)

{

min = i;

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

{

if (a[j] < a[min])

min = j;

}

temp = a[i];

a[i] = a[min];

a[min] = temp;

}

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

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

free(a);

return 0;

}

2.

#include <stdio.h>

#include <malloc.h>

int main()

{

int \*a;

int i, n,sum=0;

printf("Введите размер массива: ");

scanf("%d", &n);

a = (int\*)malloc(n \* sizeof(int));

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

{

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

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

}

int k\_ch=0,k\_nch=0;

float sum\_ch=0,sum\_nch=0,sa\_ch, sa\_nch;

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

if (a[i]%2==0)

{

k\_ch++;

sum\_ch+=a[i];

}

else

{

k\_nch++;

sum\_nch+=a[i];

}

sa\_ch=sum\_ch/k\_ch;

sa\_nch=sum\_nch/k\_nch;

if (sa\_ch>sa\_nch)

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

if (a[i]%2!=0)

a[i]=a[i]+1;

if (sa\_ch<sa\_nch)

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

if (a[i]%2==0)

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

if (sa\_nch==sa\_ch)

for (i = 0; i<n; i++) a[i]=0;

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

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

return 0;

}

3.

#include <stdio.h>

#include <stdlib.h>

int main()

{

int input;

int counter = 0;

int \* values = NULL;

int \* many\_numbers;

do {

printf("Введите целое значение (0 - выход): ");

scanf("%d",&input);

counter++;

many\_numbers = (int\*) realloc (values, counter \* sizeof(int));

if (many\_numbers != NULL)

{

values = many\_numbers;

values[counter - 1] = input;

}

else

{

free (values);

printf( "Ошибка перевыделения памяти!");

exit (1);

}

} while (input != 0);

int k=0;

float sum=0;

for (int ix = 0; ix < counter; ix++)

sum+=values[ix];

float sa;

sa=sum/(counter-1);

float\* m;

m = (float\*)malloc((counter-1) \* sizeof(float));

for (int i = 0; i < counter-1; i++)

m[i]=values[i]-sa;

for (int i = 0; i < counter-1; i++)

printf("%8.2f ",m[i]);

free (values);

free(m);

return 0;

}