## Практические 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. Найдите сумму элементов массива и выведите на экран.
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 значение элементов массива и выведите на экран в формате 8 знакомест под число и 2 знакоместа под дробную часть.
3. 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 максимальный и минимальный элемент массива и выведите на экран в отдельных строках.
4. 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 убыванию и выведите элементы отсортированного массива на экран через пробел.
5. Объявите целочисленный массив, количество элементов заранее не известно. Организуйте ввод элементов массива, пока не будет введен 0. Подсчитайте и выведите на экран количество элементов массива больших чем среднее арифметическое значение всех элементов массива.
6. Объявите целочисленный массив, количество элементов заранее не известно. Организуйте ввод элементов массива, после каждого введенного элемента необходимо спрашивать “добавить элемент (y/n)?”, пока пользователь не введет “n” продолжать ввод элементов массива, вывести количество элементов массива и их сумму в отдельных строках.

## 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]);

}

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

sum+=a[i];

free(a);

printf("%d ", sum);

return 0;

}

2.

#include <stdio.h>

#include <malloc.h>

int main()

{

int \*a;

int i, n;

float 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]);

}

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

sum+=a[i];

free(a);

printf("%8.2f ", sum/n);

return 0;

}

3.

#include <stdio.h>

#include <malloc.h>

int main()

{

int \*a;

int i, n, min, max;

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]);

}

max=a[0];

min=a[0];

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

{

if (a[i]>max) max=a[i];

if (a[i]<min) min=a[i];

}

printf("%d\n",max);

printf("%d\n",min);

return 0;

}

4.

#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;

}

5.

#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,sum=0;

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

sum+=values[ix];

float sa;

sa=sum/counter;

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

if (values[ix]>sa) k++;

printf("%d\n",k);

free (values);

return 0;

}

6.

#include <stdio.h>

#include <malloc.h>

int main()

{

int \*a = NULL, i = 0, elem, sum=0;

char c;

do {

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

scanf("%d", &elem);

a = (int\*)realloc(a, (i + 1) \* sizeof(int));

a[i] = elem;

i++;

getchar();

printf("добавить элемент (y/n)?");

c = getchar();

} while (c != 'n');

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

sum+=a[ix];

printf("%d\n",i);

printf("%d\n",sum);

free (a);

return 0;

}