7.1.1

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define n 10

int main()

{

int x[n], /\*массив из n элементов\*/ max; /\*максимальный элемент массива\*/

srand(time(NULL)); /\*функция рандома от времени\*/

printf("x = ");

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

{

x[i] = rand() % 20-9; /\*заполнение массива рандомными значениями\*/

printf("%2d ", x[i]); /\*вывод массива\*/

}

max = x[0]; /\*присвоение переменной max значения первого элемента массива\*/

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

{

if (max < x[i]) /\*сравнение max и элемента в массиве\*/

{

max = x[i]; /\*присвоение переменной max значения элемента бОльшего, чем max\*/

}

}

printf("\nMAX in array is = %2d ", max); /\*вывод максимального элемента в массиве\*/

getchar();

getchar();

return 0;

}

7.1.2(1)

#include <stdlib.h>  
#include <stdio.h>

#include <math.h>

#include <time.h>

#define n 10

int main()

{

int x[n], /\*массив из n элементов\*/

min,/\*минимальный элемент массива\*/

k,/\*разница между min и omin\*/

omin; /\*близкий к минимальному элемент массива\*/

srand(time(NULL)); /\*функция рандома от времени\*/

printf("x = ");

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

{

x[i] = rand() % 20 - 9; /\*заполнение массива рандомными значениями\*/

printf("%2d ", x[i]); /\*вывод массива\*/

}

min = x[0]; /\*присвоение переменной min значения первого элемента массива\*/

k = abs(abs(x[1]) - abs(min));

omin = x[1];

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

{

if (min > x[i]) /\*сравнение min и элемента в массиве\*/

{

min = x[i]; /\*присвоение переменной min значения элемента меньшего, чем min\*/

}

}

printf("\nMIN in array is = %2d ", min); /\*вывод минимального элемента в массиве\*/

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

{

if( (abs(min)!= abs(x[i])) && (k > abs(abs(x[i])-abs(min))))/\*нахождение ближайшего к минимальному\*/

{

k = abs(abs(x[i]) - abs(min)); /\*поиск разницы между элементом массива и минимальным\*/

omin = x[i]; /\*запись ближайшего к минимальному элемента массива\*/

}

}

printf("\nClouser with min in array is = %2d ", omin); /\*вывод ближайшего к минимальному элемента в массиве\*/

getchar();

getchar();

return 0;

}

7.1.2(2)

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

#include <time.h>

#define n 10

int main()

{

int x[n], /\*массив из n элементов\*/

min,/\*минимальный элемент массива\*/

num,/\*номер минимального элемента\*/

omin; /\*близкий к минимальному элемент массива\*/

srand(time(NULL)); /\*функция рандома от времени\*/

printf("x = ");

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

{

x[i] = rand() % 20 - 9; /\*заполнение массива рандомными значениями\*/

printf("%2d ", x[i]); /\*вывод массива\*/

}

min = x[0]; /\*присвоение переменной min значения первого элемента массива\*/

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

{

if (min > x[i]) /\*сравнение min и элемента в массиве\*/

{

min = x[i]; /\*присвоение переменной min значения элемента меньшего, чем min\*/

num = i;/\*запись номера минимального элемента\*/

}

}

printf("\nMIN in array is = %2d ", min); /\*вывод минимального элемента в массиве\*/

omin = x[0];

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

{

if ((i!=num)&&(x[i]<omin)&&(omin!=min))/\*нахождение ближайшего к минимальному\*/

{

omin = x[i]; /\*запись ближайшего к минимальному элемента массива\*/

}

}

printf("\nClouser with min in array is = %2d ", omin); /\*вывод ближайшего к минимальному элемента в массиве\*/

getchar();

getchar();

return 0;

}

7.1.4

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#define n 9 /\*количество элементов массива\*/

int main()

{

int x[n], /\*массив из n элементов\*/

i, j, /\*индексы массива\*/

c; /\*вспомогательная переменная\*/

srand(time(NULL));

printf("x(strt) = "); /\* указание на будущий вывод массива х\*/

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

{

x[i] = rand() % 20 - 9; /\*заполнение массива рандомными числами\*/

printf("%2d ", x[i]); /\*вывод массива\*/

}

printf("\nx(sort) = "); /\*перенос строки + указание на будущий вывод отсортированного массива х\*/

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

{

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

{

if (x[i] > x[j])/\*условие сортировки массива по возрастанию\*/

{

c = x[i];

x[i] = x[j];

x[j] = c;

}

}

printf("%2d ", x[i]); /\*вывод отсортированного массива\*/

}

int med = x[n/2]; /\*медиана массива\*/

printf("\nMediana in this array = %2d ", med); /\*вывод медианы массива\*/

getchar();

getchar();

return 0;

}

7.1.5

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#define n 9 /\*количество элементов массива\*/

int main()

{

int x[n], /\*массив из n элементов\*/

i, j, /\*индексы массива\*/

count=0, /\*количество различных элементов\*/

c; /\*вспомогательная переменная\*/

srand(time(NULL));

printf("x(strt) = "); /\* указание на будущий вывод массива х\*/

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

{

x[i] = rand() % 20 - 9; /\*заполнение массива рандомными числами\*/

printf("%2d ", x[i]); /\*вывод массива\*/

}

printf("\nx(sort) = "); /\*перенос строки + указание на будущий вывод отсортированного массива х\*/

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

{

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

{

if (x[i] > x[j])/\*условие сортировки массива по возрастанию\*/

{

c = x[i];

x[i] = x[j];

x[j] = c;

}

}

printf("%2d ", x[i]); /\*вывод отсортированного массива\*/

}

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

{

if (x[i]!=x[i-1])

{

count ++;

}

}

printf("\Difference el in this array = %2d ", count); /\*вывод медианы массива\*/

getchar();

getchar();

return 0;

}

7.2.1

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#define n 9 /\*количество элементов массива\*/

#define m 20 /\*максимальное число\*/

int main()

{

int x[n], /\*массив из n элементов\*/

c[m] = { 0 }, /\*вспомогательный массив из м элементов\*/

i, j = 0; /\*индексы массива\*/

srand(time(NULL));

printf("x(strt) = "); /\* указание на будущий вывод массива х\*/

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

{

x[i] = rand() % m; /\*заполнение массива рандомными числами\*/

printf("%2d ", x[i]); /\*вывод массива\*/

}

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

{

c[x[i]]++;

}

printf("\nx(sort) = "); /\*перенос строки + указание на будущий вывод отсортированного массива х\*/

int count = 0; /\*счетчик\*/

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

{

for (j = 0; j < c[i]; j++)

{

x[count] = i;/\*сортировка\*/

count++; /\*счетчик\*/

}

}

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

printf("%2d ", x[j]); /\*вывод отсортированного массива\*/

getchar();

getchar();

return 0;

}