ЛАБ№7

Задача 1. Функції користувача.

Утворити і вивести масив y з елементами yk=fi+1(k), де k=1,12, і- номер варіанта. Для розрахунку yk створити функцію користувача. Скласти ще одну функцію користувача для пошуку даних у цьому масиві. Критерії пошуку взяти відповідно свого варіанту. Шукані дані вивести в головній функції.



Скільки було від’ємних значень? Визначити максимальне значення.

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

double \*A(int n) {

return (double \*)malloc(n\*sizeof(double));

}

void am(int \*a, int n) {

int i;

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

\*(a + i) = rand()%30\*1.12;}}

void ap(int \*a, int n) {

int i;

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

printf("%d ", \*(a + i));}}

void m(int \*a, int n) {

int i, j,ch,max=-999;

int k=0;

for (i = 0; i < n; i++){ch=\*(a + i);

if(ch<0){

k++;}

if(ch>max){max=ch;}

}

printf("\nkilkist =%d \n max=%d",k,max);}

int main(){

int \*a;

int n;

printf("vvedit n");

scanf("%d", &n);

a = A(n);

am(a, n);

ap(a, n);

m(a, n);

return 0;}

Задача 2. . Функції користувача.

Утворити і вивести масив А, елементи якого описані формулою , m,n=1…4, i – номер варіанта. Скласти функцію користувача для пошуку даних у цьому масиві. Критерії пошуку взяти відповідно свого варіанту, якщо критерії пошуку не відповідають вашим даним вибрати свої значення для пошуку. Шукані дані вивести в головній функції.



Обчислити суму квадратів елементів, значення яких більші, ніж 1 та кількість таких елементів.

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

double \*A(int n) {

return (double \*)malloc(n\*sizeof(double));

}

void am(int \*a, int n) {

int i;

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

\*(a + i) = rand()%30\*1.12;}}

void ap(int \*a, int n) {

int i;

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

printf("%d ", \*(a + i));}}

void m(int \*a, int n) {

int i, j,ch;

int d=0,k=0;

for (i = 0; i < n; i++){ch=\*(a + i);

if(ch>1){

k++;d=d+ch\*ch;}}

printf("\nkilkist =%d \n sum=%d",k,d);}

int main(){

int \*a;

int n;

printf("vvedit n");

scanf("%d", &n);

a = A(n);

am(a, n);

ap(a, n);

m(a, n);

return 0;}