**КПІ ім. Ігоря Сікорського**

**Факультет інформатики та обчислювальної техніки**

**Кафедра інформатики та програмної інженерії**

**Звіт до комп‘ютерного практикуму з курсу**

**“Основи програмування ”**

Прийняв Виконала

доцент кафедри ІПІ Студентка групи ІП-21

Лісовиченко О.І. Скрипець О.О.

“17” грудня 2022 р.

**Київ 2022**

**Комп’ютерний практикум №6**

***Тема****: Використання динамічних масивів*

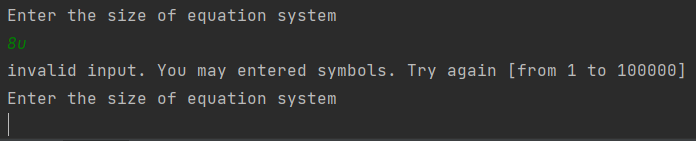
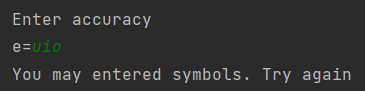
**6**

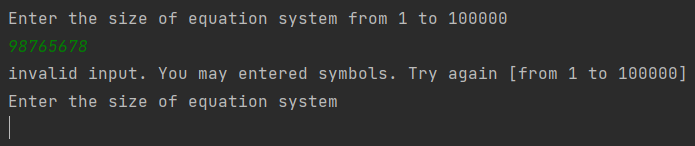
**Завдання:**

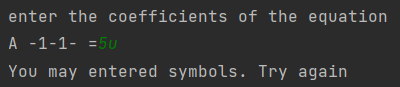
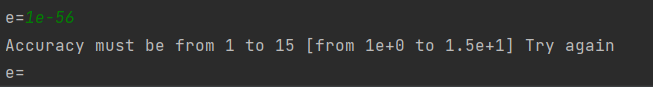
Написати програму розв’язання системи лінійних алгебраїчних рівнянь (СЛАР) методом простої ітерації з використанням динамічних масивів.

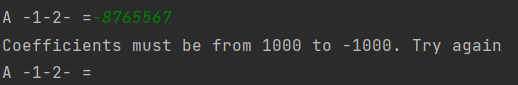
***Текст програми***

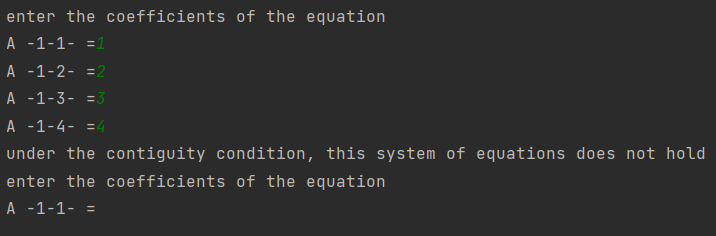
#include<stdio.h>  
#include<math.h>  
#include<stdlib.h>  
  
float accuracy\_check(float e);  
float coefficient\_check (float \*\*a, int i, int j);  
  
int main(){  
 int i, j, size;  
 float \*\*a, \*b, \*x, \*xp, e, delta, sum, zbiz;  
 char validation, exit;  
 do{  
  
 printf("Enter the size of equation system from 1 to 100000\n");  
  
 while (scanf("%d%c", &size, &validation) != 2 || validation != '\n' || size <=0 || size > 100000 ) {  
 printf("invalid input. You may entered symbols. Try again [from 1 to 100000]\nEnter the size of equation system\n");  
 }  
  
  
 a = (float\*\*)malloc(size\*sizeof(float\*));  
  
 for(i = 0; i < size; i++){  
 a[i] = (float\*)malloc(size\*sizeof(float));  
 }  
  
 b = (float\*)malloc(size\*sizeof(float));  
 x = (float\*)malloc(size\*sizeof(float));  
 xp = (float\*)malloc(size\*sizeof(float));  
  
 if(a==0 || b==0 || x==0 || xp==0 || a[i]==0){  
 printf("The memory was not allocated\n");  
 }  
  
 printf("Enter accuracy\ne=");  
 accuracy\_check(e);  
  
 if(e>1) {  
 e = 1.0 / pow(10, e);  
 }  
  
 again:  
 printf("\nenter the coefficients of the equation\n");  
 for(i=0; i<size; i++) {  
 for (j = 0; j < size; j++) {  
 coefficient\_check(a, i, j);  
 }  
  
  
 //умова збіжності  
 zbiz = 0;  
 for (int j = 0; j < size; j++) {  
 if (i != j) zbiz += a[i][j];  
 }  
  
 while(fabs(a[i][i])<=fabs(zbiz) ){  
 printf("under the contiguity condition, this system of equations does not hold\n");  
 goto again;  
 }  
  
  
  
 printf("B -%d- =",i+1);  
 while(scanf("%f%c", &b[i], &validation)!=2 || validation!='\n') {  
 fflush(stdin);  
 printf("You may entered symbols. Try again\n");  
 printf("B -%d- =",i+1);  
 }  
 printf("\n");  
 }  
  
  
 printf("Your entered this system\n");  
 for(i=0;i<size;i++){  
 for(j=0; j<size; j++){  
 printf("%f\*X%d ", a[i][j], j+1);  
 if(j!=size-1){  
 printf("+ ");  
 }  
 }  
 printf("= %f\n", b[i]);  
 }  
  
 for(i=0;i<size;i++){  
 xp[i]=b[i]/a[i][i];  
 }  
 do{  
 for(i=0; i<size;i++){  
 sum=0;  
 for(j=0;j<size;j++){  
  
 if(i!=j){  
 sum+=a[i][j]\*xp[j];  
 }}  
 xp[i]=(b[i]-sum)/a[i][i];  
 }  
  
  
 delta=0;  
 for(i=0; i<size; i++){  
 if(fabs(x[i]-xp[i])>delta){  
 delta=fabs(x[i]-xp[i]);  
 }x[i]=xp[i];  
 }  
  
  
  
 }while(delta>=e);  
  
 printf("\nresults:\n");  
 for(i = 0; i < size; i++){  
 printf("x%d = %f\n", i+1, x[i]);  
 }  
  
 printf("\nthe complete equation\n");  
 float res[size];  
 for(i=0; i<size; i++){  
 res[i]=0;  
 for(j=0; j<size; j++){  
 res[i]+=a[i][j]\* x[j];  
 }  
 }  
 for(i=0;i<size; i++){  
 for(j=0;j<size;j++){  
 printf("%f \* %f", a[i][j], x[j]);  
 if(j!=size-1){  
 printf(" + ");  
 }  
 }  
 printf("= %f\n", res[i]);  
  
 }  
  
 for(i=0; i<size; i++){  
 free(a[i]);  
 }  
 free(a);  
 free(b);  
 free(x);  
 free(xp);  
  
 exit = getchar();  
 fflush(stdin);  
 }while(exit != 'n' && exit != 'N');  
  
 return 0;  
}  
  
  
  
float accuracy\_check(float e){  
 char validation, invalid;  
 do{  
 if(scanf("%f%c", &e, &validation)!=2 || validation!='\n'){  
 printf("You may entered symbols. Try again\ne=");  
 fflush(stdin);  
 invalid=1;  
 }  
 else if( e > 1.5e+1 || e < 1e+0){  
 printf("Accuracy must be from 1 to 15 [from 1e+0 to 1.5e+1] Try again\ne=");  
 invalid=1;  
 }else{  
 invalid=0;  
 }  
 }while(invalid);  
}  
  
  
float coefficient\_check(float \*\*a, int i, int j){  
 char invalid, validation;  
 do {  
 printf("A -%d-%d- =", i + 1, j + 1);  
  
 if (scanf("%f%c", &a[i][j], &validation) != 2 || validation != '\n') {  
 printf("You may entered symbols. Try again\n");  
 fflush(stdin);  
 invalid = 1;  
 }  
 if (a[i][j] > 1000 || a[i][j] < -1000) {  
 printf("Coefficients must be from 1000 to -1000. Try again\n");  
 invalid = 1;  
 } else if (i == j && a[i][j] == 0) {  
 printf("Diagonal elements 0. Try again\n");  
 invalid = 1;  
 } else invalid = 0;  
 } while (invalid);  
}

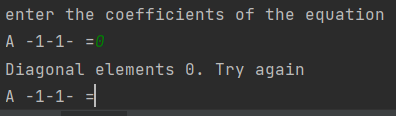
***Введені та одержані результати:***

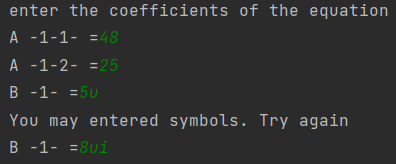


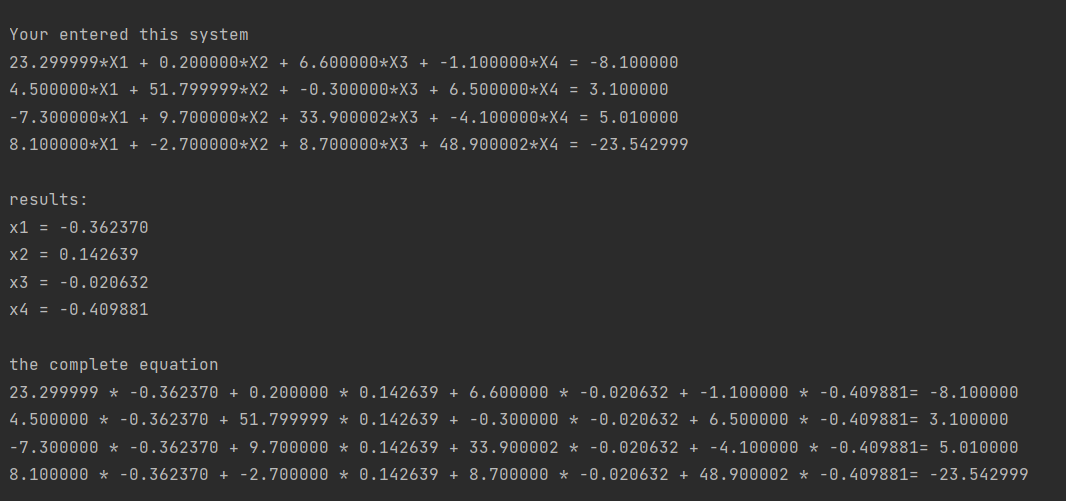


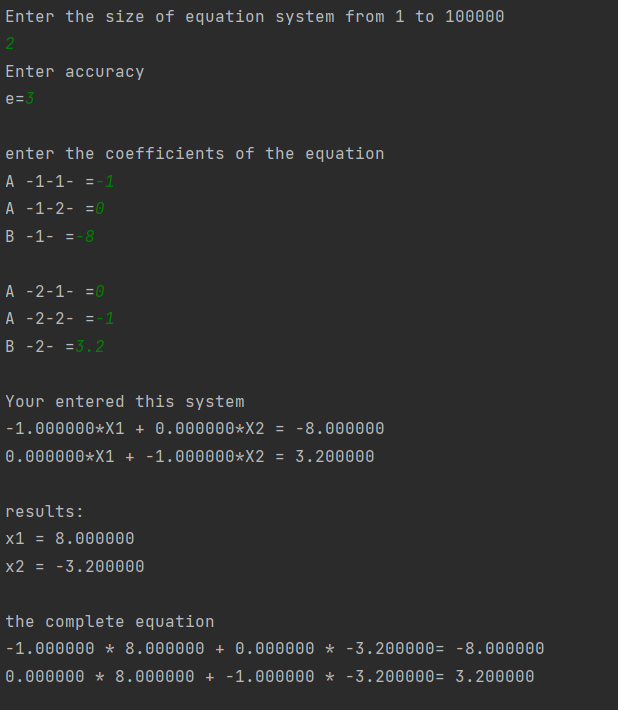


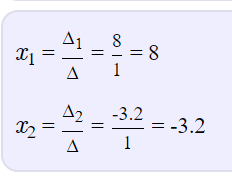


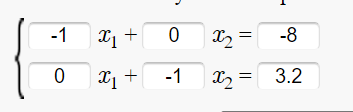












***Висновки:*** Теоретичні розрахунки відповідають отриманим. Програма працює

коректно та вирішує поставлені завдання.