

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра прикладної математики

КУРСОВА РОБОТА

з дисципліни «Програмування»

на тему: РОЗВ'ЯЗУВАННЯ СИСТЕМ ЛІНІЙНИХ РІВНЯНЬ МЕТОДОМ
РЕЛАКСАЦІЇ

Студента І курсу, групи КМ-53
напряму підготовки 6.040301 –
прикладна математика
ГАЛЕТИ М. С.

Керівник
ЛЮБАШЕНКО Н.Д.

Національна оцінка _____
Кількість балів: _____ Оцінка:
ECTS _____

ЗМІСТ

ВСТУП	3
1. Постановка задачі	4
2. Вибір методу розв'язання задачі	5
3. Алгоритм програми	7
4. Опис програми	8
5. Результати	10
ВИСНОВКИ	14
СПИСОК ЛІТЕРАТУРИ	15
Додаток А. Текст програми на мові С	16

ВСТУП

Обчислювальну техніку останніми роками широко застосовують у всіх сферах діяльності людини. Вона стала каталізатором науково-технічного прогресу. Бурхливий розвиток ЕОМ сприяв широкому процесу математизації науки, техніки і господарства в цілому. Саме розробка і застосування математичних методів розв'язування прикладних задач на базі ЕОМ є предметом сучасної математики.

Розвиток обчислювальної математики тісно пов'язаний з розвитком програмування, яке йде шляхом спрощення способів спілкування людини з комп'ютером. На сучасному етапі розвитку виникають мови програмування наближені до природних, розвиваються проблемно орієнтовані мови програмування, засоби візуального програмування, створюються пакети прикладних програм. Виникають і інтенсивно розвиваються структурне програмування і спеціалізовані мови для розробки структурованих програм.

$$\left\{ \begin{array}{l} R_1^{(0)} = c_1 - x_1^{(0)} + \sum_{j=2}^n b_{1j} x_j^{(0)} \\ R_2^{(0)} = c_2 - x_2^{(0)} + \sum_{\substack{j=1 \\ j \neq 2}}^n b_{2j} x_j^{(0)} \\ \dots\dots\dots \\ R_n^{(0)} = c_n - x_n^{(0)} + \sum_{j=1}^{n-1} b_{nj} x_j^{(0)} \end{array} \right.$$

Якщо одній із невідомих $x_s^{(0)}$ надати приросту $\delta x_s^{(0)}$, то відповідна нев'язка $R_s^{(0)}$ зменшується на величину $\delta x_s^{(0)}$, а всі інші нев'язки $R_i^{(0)} (i \neq s)$ збільшаться на величину $b_{is} \delta x_s^{(0)}$. Таким чином, щоб обернути чергову нев'язку $R_s^{(0)}$ в нуль, достатньо величині $x_s^{(0)}$ надати приросту $\delta x_s^{(0)} = R_s^{(0)}$. Тоді ми будемо мати $R_s^{(1)} = 0$ і $R_i^{(1)} = R_i^{(0)} + b_{is} \delta x_s^{(0)}$ при $i \neq s$.

Метод релаксації в його найпростішій формі полягає в тому, що на кожному кроці обертають в нуль максимальну за модулем нев'язку шляхом зміни значення відповідної компоненти наближення. Процес закінчується, коли всі нев'язки останньої перетвореної системи будуть дорівнювати нулю із заданою точністю.

3. Алгоритм програми

На рисунку 1 наведений алгоритм роботи програми у вигляді блок-схеми.

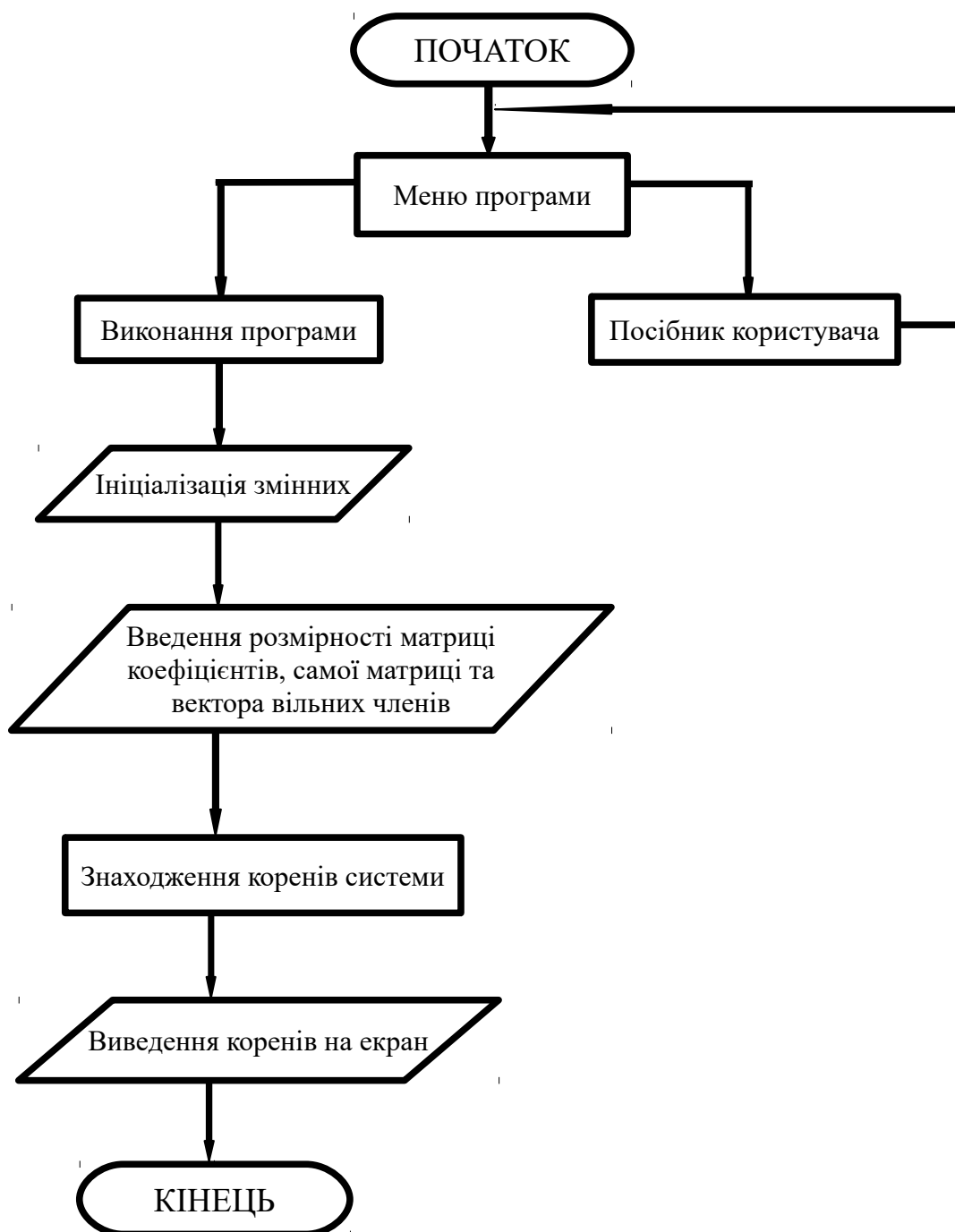


Рисунок 1

4. Опис програми

Таблиця 1 — Опис основних функцій програми

Назва функції	Опис формальних параметрів	Призначення (результат роботи)
stroka	- int j — затримка (мс) - string s — рядок, що виводиться на екран	Посимвольне виведення рядка на екран
NatBudKlav	- int l — координата по x - int p — координата по y	Виведення на екран рядка «Натисніть будь-яку клавішу»
menu	void	Меню програми
frame	void	Обкладинка програми
work	void	Виконання програми — пошук коренів системи
task	void	Посібник користувача

Таблиця 2 — Опис основних даних

Назва змінної	Опис	Призначення
y	Цілочисельна змінна	Перевірка на введення
n	Цілочисельна змінна	Містить в собі розмірність матриці коефіцієнтів
tmp	Цілочисельна змінна	Зберігає в собі індекс елемента масиву
**matrix	Вказівник на масив масиву даних з плаваючою точкою	Містить адресу матриці коефіцієнтів
*b	Вказівник на масив даних з плаваючою точкою	Містить адресу вектора вільних членів
*x	Вказівник на масив даних з плаваючою точкою	Містить адресу вектора невідомих змінних
*diag	Вказівник на масив даних з плаваючою точкою	Містить адресу масиву діагональних елементів матриці коефіцієнтів

*r	Вказівник на масив даних з плаваючою точкою	Містить адресу масиву нев'язок
*sum	Вказівник на масив даних з плаваючою точкою	Містить адресу масиву сум $\sum_{j=1}^n a_{ij} x_j, \quad i = \overline{1..n}$
max	Змінна з плаваючою точкою	Зберігає в собі максимальний елемент масиву нев'язок
flag	Булева змінна	Перевірка на рівність нулю масиву нев'язок

5. Результати

Результати роботи програми показано на рисунках 1, 2, 3, 4, 5 та 6. В якості контрольного прикладу взято наступну систему рівнянь

$$\begin{cases} 4x_1 - 3x_2 = 1 \\ x_1 - 4x_2 = -3 \end{cases}.$$

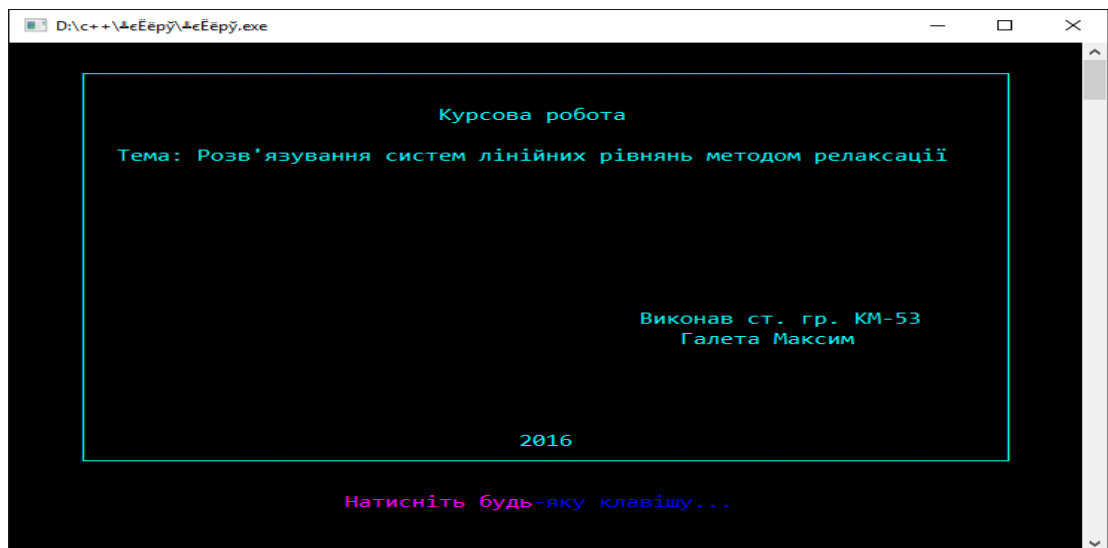


Рисунок 1 — Обкладинка програми

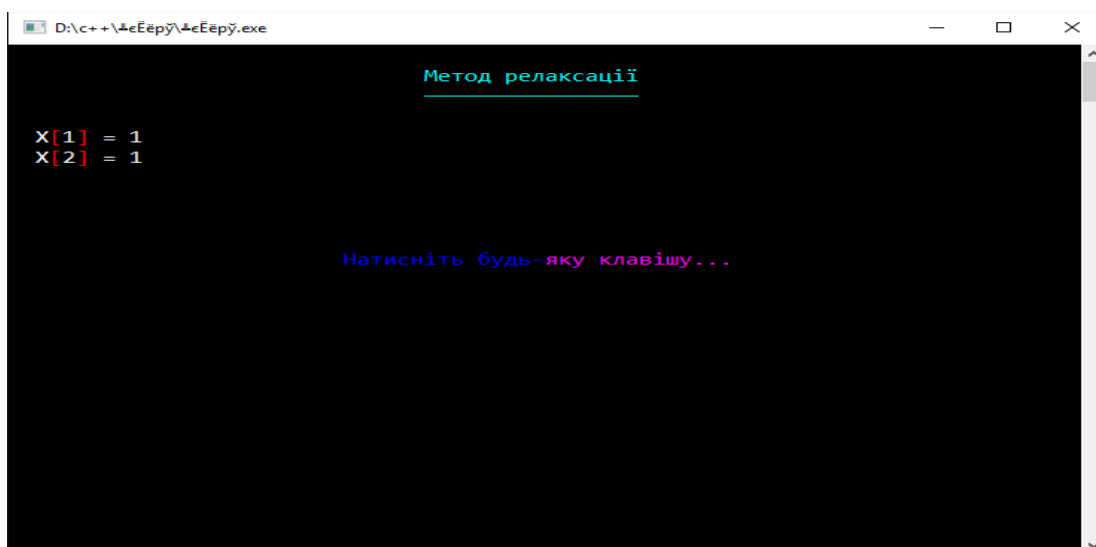


Рисунок 6 — Корені системи

ВИСНОВКИ

Під час виконання курсової роботи було вдосконалено навички опрацювання одновимірних і багатовимірних динамічних масивів, придбано навички з розробки зручного інтерфейсу програми, зокрема курсорного меню, поглиблено знання про методи розв'язання систем лінійних рівнянь.

Результатом роботи є програма знаходження розв'язків системи лінійних рівнянь методом релаксації.

СПИСОК ЛІТЕРАТУРИ

Демидович Б.П., Марон И.А. Основы вычислительной математики.: М.,
Наука, 1970, - 660 с.

Додаток А. Текст програми на мові С

```
#include<windows.h>
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
#include<math.h>
#include<iostream>
using namespace std;

void menu();

void frame()
{
    system("cls");
    textcolor(LightCyan);
    gotoxy(5,1);
    putchar(218);
    for(int i=0; i<68; i++)
        putchar(196);
    putchar(191);
    for(int i=2; i<20; i++)
    {
        gotoxy(5,i);
        putchar(179);
        gotoxy(74,i);
        putchar(179);
    }
    gotoxy(5,20);
    putchar(192);
    for(int i=0; i<68; i++)
        putchar(196);
    putchar(217);
    ukr(1251);
    gotoxy(32,3);
        stroka(15,"Курсова робота");
    gotoxy(8,5);
    stroka(15,"Тема: Розв'язування систем лінійних рівнянь методом релаксації");
    gotoxy(47,13);
    stroka(15,"Виконав ст. гр. КМ-53");
    gotoxy(50,14);
    stroka(15,"Галета Максим");
    gotoxy(38,19);
    stroka(15,"2016");
    NatBudKlav(25,22);
    menu();
}

void task();

void work();
```



```

void menu()
{
    int item=0, ch;
    string MenuItem[3]={"Запустити програму на виконання","Посібник користувача",
    "Вийти з програми"};
    system("cls");
    while(true)
    {
        textcolor(LightCyan);
        gotoxy(37,1);
        printf("МЕНЮ");
        gotoxy(37,2);
        ukr(866);
        for(short i=0; i<4; i++)
            putchar(196);
        ukr(1251);
        for(int i=0; i<3; i++)
        {
            if (i==item)
            {
                textcolor(Black);
                textbackground(LightCyan);
                gotoxy(2,4+i);
                cout<<MenuItem[i];
                textbackground(Black);
            }
            else
            {
                textbackground(Black);
                textcolor(White);
                gotoxy(2,4+i);
                cout<<MenuItem[i];
            }
        }
        gotoxy(79,6);
        ch=getch();
        switch(ch)
        {
            case 80:
                item+=1;
                if(item>2)
                    item=0;
                break;
            case 72:
                item-=1;
                if(item<0)
                    item=2;
                break;
            case 13:
                if(item==0)
                    work();
                if(item==1)
                    task();
                if(item==2)
                    return;
        }
    }
}

```

```

                                break;
                            }
                        }
                    }

void work()
{
    int y, n, tmp;
    float **matrix, *b, *x, *diag, *r, *sum, max;
    bool flag;
    while(true)
    {
        system("cls");
        textcolor(LightCyan);
        gotoxy(31,1);
        printf("Метод релаксації");
        gotoxy(31,2);
        ukr(866);
        for(short i=0; i<16; i++)
            putchar(196);
        ukr(1251);
        textcolor(White);
        gotoxy(2,4);
        stroka(15,"У прямокутнику введіть розмірність матриці коефіцієнтів та
натисніть ");
        textcolor(LightGreen);
        stroka(15,"ENTER");
        ukr(866);
        textcolor(LightCyan);
        gotoxy(1,5);
        putchar(218);
        for(short i=0; i<76; i++)
            putchar(196);
        gotoxy(78,5);
        putchar(191);
        gotoxy(1,6);
        putchar(179);
        gotoxy(78,6);
        putchar(179);
        gotoxy(1,7);
        putchar(192);
        for(short i=0; i<76; i++)
            putchar(196);
        gotoxy(78,7);
        putchar(217);
        gotoxy(3,6);
        ukr(1251);
        textcolor(White);
        y=scanf("%d",&n);
        if((y==0)||(n<2))
        {
            gotoxy(2,9);
            stroka(15,"Ви ввели некоректні дані. Повторіть будь-ласка
введення");
            Sleep(1000);

```

```

        fflush(stdin);
    }
    else
        break;
}
matrix=(float**)calloc(n,sizeof(float*));
if(matrix==NULL)
    exit(-1);
for(int i=0; i<n; i++)
{
    matrix[i]=(float*)calloc(n,sizeof(float));
    if(matrix[i]==NULL)
        exit(-1);
}
b=(float*)calloc(n,sizeof(float));
if(b==NULL)
    exit(-1);
x=(float*)calloc(n,sizeof(float));
if(x==NULL)
    exit(-1);
r=(float*)calloc(n,sizeof(float));
if(r==NULL)
    exit(-1);
diag=(float*)calloc(n,sizeof(float));
if(diag==NULL)
    exit(-1);
sum=(float*)calloc(n,sizeof(float));
if(sum==NULL)
    exit(-1);
for(int i=0; i<n; i++)
    for(int j=0; j<n; j++)
        while(true)
        {
            system("cls");
            textcolor(LightCyan);
            gotoxy(31,1);
            printf("Метод релаксації");
            gotoxy(31,2);
            ukr(866);
            for(short k=0; k<16; k++)
                putchar(196);
            ukr(1251);
            textcolor(White);
            gotoxy(2,4);
            stroka(15,"У прямокутнику введіть коефіцієнт A");
            textcolor(LightRed);
            stroka(15,"[");
            textcolor(White);
            printf("%d",i+1);
            Sleep(15);
            textcolor(LightRed);
            stroka(15,"]");
            textcolor(White);
            printf("%d",j+1);
            Sleep(15);
        }
    }
}

```

```

        textcolor(LightRed);
        stroka(15, "I");
        textcolor(White);
        stroka(15, " та натисніть ");
        textcolor(LightGreen);
        stroka(15, "ENTER");
        ukr(866);
        textcolor(LightCyan);
        gotoxy(1,5);
        putchar(218);
        for(short k=0; k<76; k++)
            putchar(196);
        gotoxy(78,5);
        putchar(191);
        gotoxy(1,6);
        putchar(179);
        gotoxy(78,6);
        putchar(179);
        gotoxy(1,7);
        putchar(192);
        for(short k=0; k<76; k++)
            putchar(196);
        gotoxy(78,7);
        putchar(217);
        gotoxy(3,6);
        ukr(1251);
        textcolor(White);
        y=scanf("%f",&matrix[i][j]);
        if(y==0)
        {
            gotoxy(2,9);
            stroka(15, "Ви ввели некоректні дані. Повторіть будь-
ласка введення");

            Sleep(1000);
            fflush(stdin);
        }
        else
            break;
    }
    for(int i=0; i<n; i++)
        while(true)
        {
            system("cls");
            textcolor(LightCyan);
            gotoxy(31,1);
            printf("Метод релаксації");
            gotoxy(31,2);
            ukr(866);
            for(short k=0; k<16; k++)
                putchar(196);
            ukr(1251);
            textcolor(White);
            gotoxy(2,4);
            stroka(15, "У прямокутнику введіть вільний член B");
            textcolor(LightRed);

```

```

        stroka(15,"[");
        textcolor(White);
        printf("%d",i+1);
        Sleep(15);
        textcolor(LightRed);
        stroka(15,"]");
        textcolor(White);
        stroka(15," та натисніть ");
        textcolor(LightGreen);
        stroka(15,"ENTER");
        ukr(866);
        textcolor(LightCyan);
        gotoxy(1,5);
        putchar(218);
        for(short k=0; k<76; k++)
            putchar(196);
        gotoxy(78,5);
        putchar(191);
        gotoxy(1,6);
        putchar(179);
        gotoxy(78,6);
        putchar(179);
        gotoxy(1,7);
        putchar(192);
        for(short k=0; k<76; k++)
            putchar(196);
        gotoxy(78,7);
        putchar(217);
        gotoxy(3,6);
        ukr(1251);
        textcolor(White);
        y=scanf("%f",&b[i]);
        if(y==0)
        {
            gotoxy(2,9);
            stroka(15,"Ви ввели некоректні дані. Повторіть будь-ласка
введення");

            Sleep(1000);
            fflush(stdin);
        }
        else
            break;
    }
    for(int i=0; i<n; i++)
        for(int j=0; j<n; j++)
            if(i==j)
                diag[i]=matrix[i][j];
    for(int i=0; i<n; i++)
    {
        b[i]=b[i]/diag[i];
        for(int j=0; j<n; j++)
            matrix[i][j]=matrix[i][j]/diag[i];
    }
    for(int i=0; i<n; i++)
        x[i]=0;

```

```

flag=false;
while(flag==false)
{
    for(int i=0; i<n; i++)
    {
        sum[i]=0;
        for(int j=0; j<n; j++)
            sum[i]=sum[i]+(matrix[i][j]*x[j]);
    }
    for(int i=0; i<n; i++)
        r[i]=b[i]-sum[i];
    max=r[0];
    tmp=0;
    for(int i=1; i<n; i++)
        if(max<r[i])
        {
            max=r[i];
            tmp=i;
        }
    for(int i=0; i<n; i++)
        if(i==tmp)
            x[i]=x[i]+max;
    for(int i=0; i<n; i++)
    {
        if((r[i]==0)|| (r[i]<0.000001))
            flag=true;
        else
        {
            flag=false;
            break;
        }
    }
}
system("cls");
textcolor(LightCyan);
gotoxy(31,1);
printf("Метод релаксації");
gotoxy(31,2);
ukr(866);
for(short k=0; k<16; k++)
    putchar(196);
ukr(1251);
textcolor(White);
for(int i=0; i<n; i++)
{
    gotoxy(2,4+i);
    printf("X");
    textcolor(LightRed);
    printf("[");
    textcolor(White);
    printf("%d",i+1);
    textcolor(LightRed);
    printf("]");
    textcolor(White);
    printf(" = ");
}

```

```

        cout<<x[i];
    }
    NatBudKlav(25,8+n);
    for(int i=0; i<n; i++)
        free(matrix[i]);
    free(matrix);
    free(b);
    free(x);
    free(r);
    free(diag);
    free(sum);
    return;
}

void task()
{
    system("cls");
    textcolor(LightCyan);
    gotoxy(30,1);
    printf("Завдання програми");
    gotoxy(30,2);
    ukr(866);
    for(short k=0; k<17; k++)
        putchar(196);
    ukr(1251);
    textcolor(White);
    gotoxy(3,4);
    stroka(15,"Метод релаксації передбачає пошук раціональних коренів систем
лінійних рів-");
    gotoxy(1,5);
    stroka(15,"нянь. Коефіцієнти, що стоять при невідомих, утворюють квадратну
матрицю, а");
    gotoxy(1,6);
    stroka(15,"вільні елементи - вектор вільних членів. Для того, щоб метод спрацював,
необ-");
    gotoxy(1,7);
    stroka(15,"хідно, щоб діагональні елементи цієї матриці мали діагональну перевагу,
тобто");
    gotoxy(1,8);
    stroka(15,"мінімальний діагональний елемент по модулю має бути більшим за всі
інші недіа-");
    gotoxy(1,9);
    stroka(15,"гональні елементи теж взятими по модулю, включаючи елементи вектора
вільних");
    gotoxy(1,10);
    stroka(15,"членів. В іншому випадку таку систему неможливо розв'язати методом
релаксації,");
    gotoxy(1,11);
    stroka(15,"і розв'зки системи будуть невірними.");
    NatBudKlav(25,14);
    system("cls");
    textcolor(LightCyan);
    gotoxy(30,1);
    printf("Завдання програми");
    gotoxy(30,2);

```

```

        ukr(866);
        for(short k=0; k<17; k++)
            putchar(196);
        ukr(1251);
        textcolor(White);
        gotoxy(3,4);
        stroka(15,"Користувач має ввести з клавіатури необхідні для розв'язання дані, а
саме:");
        gotoxy(6,5);
        stroka(15,"- розмірність квадратної матриці коефіцієнтів при невідомих
(відповідно)");
        gotoxy(8,6);
        stroka(15,"такою ж буде розмірність вектора вільних членів");
        gotoxy(6,7);
        stroka(15,"- коефіцієнти матриці (поелементне введення)");
        gotoxy(6,8);
        stroka(15,"- елементи вектора вільних членів (поелементне введення).");
        gotoxy(3,9);
        stroka(15,"Після цього на екран буде виведено вектор невідомих змінних - розв'язок
си-");
        gotoxy(1,10);
        stroka(15,"стеми");
        NatBudKlav(25,13);
    }

    int main()
    {
        frame();
    }

```