

### 1. Формування віконного інтерфейсу проекту

Для створення проекту “Розв’язування СЛАР” використаємо компоненту **TSringGrid (Additional)**, яка призначена для **введення і відображення** на формі текстової інформації у вигляді таблиці. Розмір таблиці визначається властивостями **ColCount** та **RowCount** (кількість стовпців і рядків відповідно). Розміри комірок у пікселях задають властивості **DefaultColWidth** та **DefaultRowHeight**.

Якщо треба задати розмір стовпця чи рядка окремо, скористайтеся властивостями **ColWidths** чи **RowHeights**.

Наприклад, ми задаємо для першого стовпця ширину у 40 пікселів так:

```
StringGrid1.ColWidths[0] := 40;
```

Нумерація комірок компоненти розпочинається з нуля.

Властивості **FixedCols** і **FixedRows** визначають скільки стовпців і рядків будуть використовуватися для назв стовпців і рядків. За замовчуванням ці дві властивості дорівнюють 1. Такі стовпці і рядки виділяються сірим кольором (це визначає властивість **FixedColor**) і при прокручуванні таблиці залишаються зафіксованими. Щоб прибрати ці фіксовані стовпець і рядок необхідно, на етапі формування інтерфейсу, при активізованій компоненті **TstringGrid**, в **Object Inspector** надати властивостям **FixedCols** і **FixedRows** значення 0.

Для доступу до деякої комірки таблиці, треба використовувати властивість **Cells**. Наприклад, щоб записати у другий рядок другого стовпця значення 12.5, використовуємо оператор:

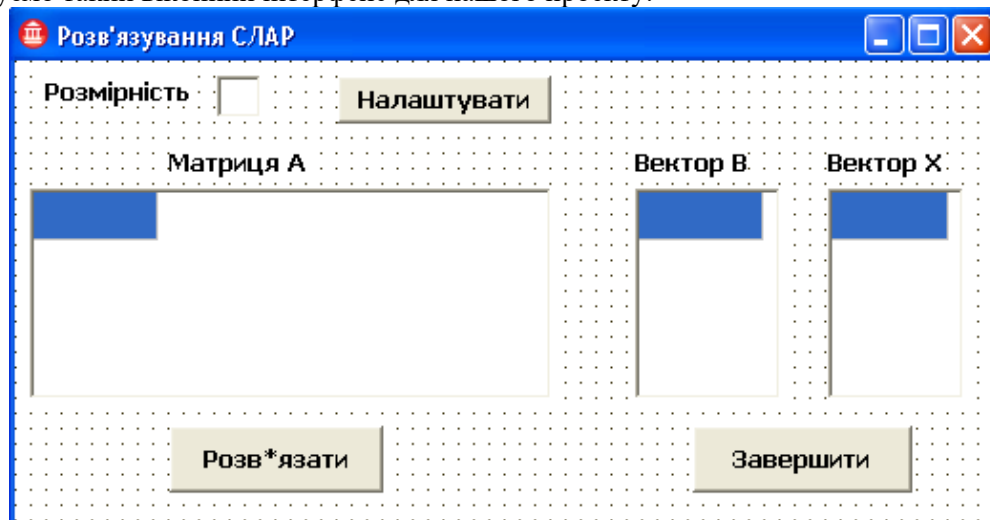
```
StringGrid1.Cells[1,1] := 12.5;
```

Важливим параметром компоненти **TSringGrid** є її властивість **Options**. Її підвластивість **goEditing** відповідає за **можливість редагування** комірок таблиці. Якщо ця властивість встановлена в **true**, то редагування є можливим.

Підвластивість **goRowSelect** регулює можливість виділення усього рядка комірок (якщо вона встановлена в **true**, то, при клацанні на комірці, виділяється весь ряд). Встановлення в **true** підвластивості **goTabs** дозволяє переміщатися між комірками за допомогою клавіші **TAB**.

Для дозволу змінювати розміри стовпців шляхом перетягування межів у заголовків, значення підвластивості **goColSizing** має бути **true**

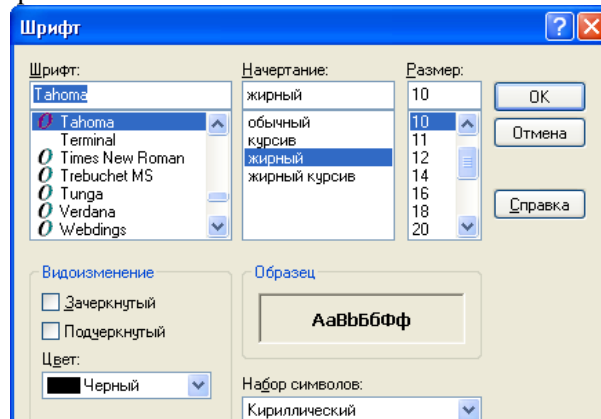
Сформуємо такий віконний інтерфейс для нашого проекту:



Для цього:

1. Запускаємо середовище Delphi (Lazarus) у віконному режимі роботи.
2. Встановлюємо значення властивості **Caption** нашої форми таким: **Ваше ПІБ Розв'язування СЛАР**.
3. Розміщуємо на полі форми такі візуальні компоненти
  - 4 компоненти **TLabel** для підписування інших компонент. Властивості **Caption** компоненти **Label1** надаємо значення **Розмірність**, відповідно:  
**Label2** – **Матриця А**,  
**Label3** – **Вектор В** і  
**Label4** – **Вектор X**.
  - 1 компоненту **TEdit** для введення числового значення розмірності матриці **А** і векторів **В** та **X**.

- 3 компоненти **TStringGrid**: **TStringGrid1** для елементів матриці **A**, **TStringGrid2** – для введення елементів вектору правої частини СЛАР, **TStringGrid3** – для виведення значень вектору розв'язку **X**.
4. Налаштовуємо всі три компоненти **TStringGrid** так:
- Встановлюємо значення **0** для властивостей **FixedCols** і **FixedRows**.
  - Змінюємо з **False** на **True** значення під властивості **goEditing** для властивості **Options**.
  - Встановлюємо значення **1** для властивостей **ColCount** та **RowCount**.
  - Маркерами встановлюємо видимі розміри компонент так, як на рисунку.
5. Розміщуємо на полі форми 3 компоненти **TButton** і їхнім властивостям **Caption** надаємо значення: **TButton1** – **Налаштувати**, **TButton2** – **Розв'язати** і **TButton3** – **Завершити**.
6. Обираємо шрифт значення **Caption** для кожної компоненти такий, як на рис. Для цього активізуємо компоненту і в **Object Inspector** обираємо властивість **Font**. Праворуч натискаємо кнопку з трикрапкою: **Font** (TFont) ... Одержимо вікно **Шрифт**.



7. Програмуємо кнопку **Завершити**. Для цього двічі клацнемо на ній правою кнопкою мишки. Одержимо заготовку процедури обробки цієї події. Вставляємо туди виклик процедури **Close**.

```
Procedure TForm1.Button3Click(Sender: TObject);
Begin Close
end;
```

8. Програмуємо кнопку **Налаштувати** так:

```
procedure TForm1.Button1Click(Sender: TObject);
Var N : Integer;
Begin
N := StrToInt(Edit1.Text);
StringGrid1.ColCount := N;
StringGrid1.RowCount := N;
StringGrid2.RowCount := N;
StringGrid3.RowCount := N;
end;
```

9. Програмуємо кнопку **Розв'язати** так:

```
procedure TForm1.Button2Click(Sender: TObject);
Var A : Matr;
b, x : Vec;
i, j, N: Integer;
begin
N := StrToInt(Edit1.Text);
For i:=1 to N do
For j:=1 to N do
Begin
a[i,j] := StrToFloat(StringGrid1.Cells[j - 1,i - 1]);
b[j] := StrToFloat(StringGrid2.Cells[0,j - 1]);
End;
// <тут поставите виклик процедури розв'язання СЛАР>
For i:=1 to N do
Begin
StringGrid3.Cells[0,i-1]:=FloatToStr(x[i]);
End;
end;
```

10. Вставляємо в текст модуля unit **Unit1** описи власних типів даних **Matr** і **Vec**:

```

Type Matr = Array[1..5,1..5] of Real;
Vec = Array[1..5] of Real;
TForm1 = class(TForm)

```

### Програмування методу Крамера для розв'язання СЛАР

Використаємо для створення нашого проекту, відомий з математики, метод Крамера для розв'язування СЛАР третього порядку ( $N = 3$ ). Записуємо цю систему:

$$\begin{cases} a_{1,1}x + a_{1,2}y + a_{1,3}z = b_1 \\ a_{2,1}x + a_{2,2}y + a_{2,3}z = b_2 \\ a_{3,1}x + a_{3,2}y + a_{3,3}z = b_3 \end{cases} \quad (1)$$

Введемо позначення визначника системи:

$$\Delta = \begin{vmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{vmatrix}. \quad (2)$$

Визначник матриці третього порядку можна обчислити так:

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \left( \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} + \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} - \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \right) =$$

$$\Delta = a_{1,1} \times a_{2,2} \times a_{3,3} + a_{1,2} \times a_{2,3} \times a_{3,1} + a_{1,3} \times a_{2,1} \times a_{3,2} - a_{3,1} \times a_{2,2} \times a_{1,3} - a_{3,2} \times a_{2,3} \times a_{1,1} - a_{2,1} \times a_{1,2} \times a_{3,3};$$

Якщо  $\Delta \neq 0$ , тоді розв'язок системи (1) однозначно обчислюється за формулами Крамера:

$$x = \frac{\Delta_x}{\Delta}, \quad y = \frac{\Delta_y}{\Delta}, \quad z = \frac{\Delta_z}{\Delta} \quad (\Delta \neq 0), \text{ де:}$$

$$\Delta_x = \begin{vmatrix} b_1 & a_{1,2} & a_{1,3} \\ b_2 & a_{2,2} & a_{2,3} \\ b_3 & a_{3,2} & a_{3,3} \end{vmatrix}, \quad \Delta_y = \begin{vmatrix} a_{1,1} & b_1 & a_{1,3} \\ a_{2,1} & b_2 & a_{2,3} \\ a_{3,1} & b_3 & a_{3,3} \end{vmatrix}, \quad \Delta_z = \begin{vmatrix} a_{1,1} & a_{1,2} & b_1 \\ a_{2,1} & a_{2,2} & b_2 \\ a_{3,1} & a_{3,2} & b_3 \end{vmatrix}.$$

Тестовий приклад

#### Завдання:

Створити інтерфейс проекту згідно з методикою, яку описано вище. написати процедуру методу Крамера, у якій обчислення значення кожного з визначників здійснюється функцією, яку необхідно написати окремо. Під'єднати процедуру до проекту. Протестувати роботу програми на системах рівнянь порядку (3\*3).

#### Додаткове завдання:

Написати консольний варіант методу Крамера, із використанням підпрограм.