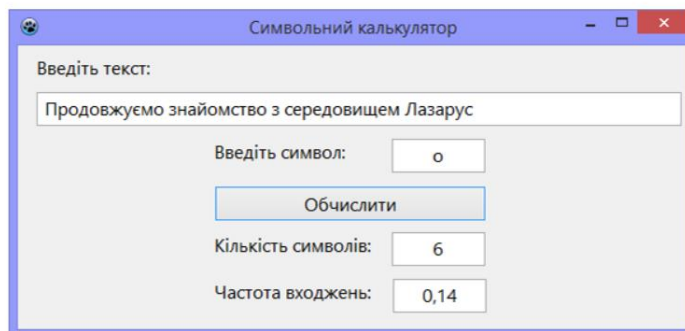


## Уроки 20 і 21. Символьні рядки

### Вивчення нового матеріалу

#### Слайд № 1

Сьогодні ми створимо проект символьного калькулятора.  
У ньому користувач вводитиме текст, а програма рахуватиме, скільки разів у цьому тексті трапляється та чи інша літера.



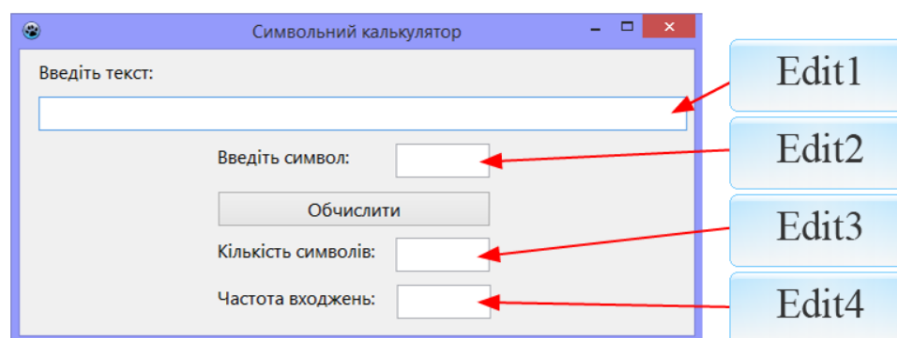
Так, у зображеному тексті літера “o” трапляється 6 разів.  
Обчислюватимемо також частоту – кількість входжень символу, поділену на загальну кількість символів ( $6/44 \approx 0,14$ )

### Вправа

#### Вправа № 1

### Вправа 1 у Lazarus

1. Створіть форму згідно зразка.
2. Встановіть необхідні властивості для елементів керування:
  - розмір шрифту на формі – 12 пт.;
  - для текстового поля **Edit2** задайте максимальну довжину тексту 1;
  - для текстових полів **Edit3** та **Edit4** – режим тільки для читання;
  - в **Edit2**, **Edit3** та **Edit4** встановіть вирівнювання тексту по центру.

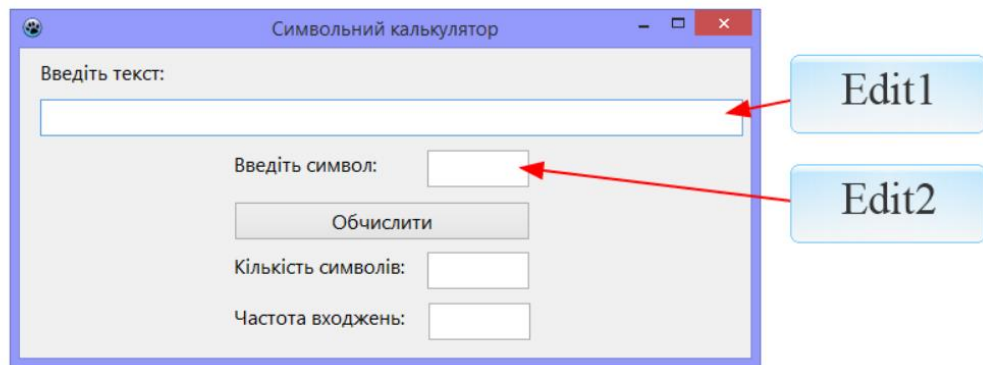


Слайд № 2



Тип string не підтримує символів кирилиці.

Для використання кирилиці призначено тип **Unicodestring**



Слайд № 3

Розглянемо принцип запису даних до змінних текстового типу

$a := \text{”моя програма”}$

Ось так, наприклад, у пам’яті комп’ютера буде записано рядок  
“моя програма”:

i	1	2	3	4	5	6	7	8	9	10	11	12
a	м	о	я		п	р	о	г	р	а	м	а

Тут **i** – порядковий номер символу рядка **a**.



Зверніть увагу, що пробіл враховується як окремий символ!

Присвоєння значення окремому символу в рядку

`a:= 'морозево';`

i	1	2	3	4	5	6	7	8
a	м	о	р	о	з	е	в	о

`a[6]:= 'и';`

i	1	2	3	4	5	6	7	8
a	м	о	р	о	з	и	в	о

Отже, сьогодні ми запрограмуємо символний калькулятор, який підраховуватиме кількість певних символів у рядку.

Розглянемо принцип його дії.

Слайд № 6

Сюди користувач вводитиме текст

А сюди – символ, кількість входжень якого в текст рахуватимемо

i	1	2	3	4	5	6	7	8	9	10	11	12
a												

Символ для підрахунку

Кількість входжень символу

Слайд № 7

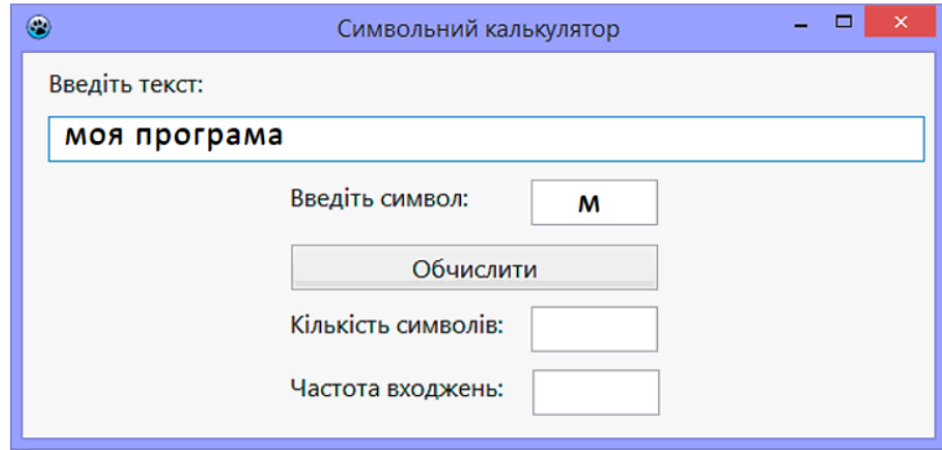
i	1	2	3	4	5	6	7	8	9	10	11	12
a	м	о	я		п	р	о	г	р	а	м	а

м

Символ для підрахунку

Кількість входжень символу

Слайд № 8



i	1	2	3	4	5	6	7	8	9	10	11	12
а	м	о	я		п	р	о	г	р	а	м	а

=

м

True

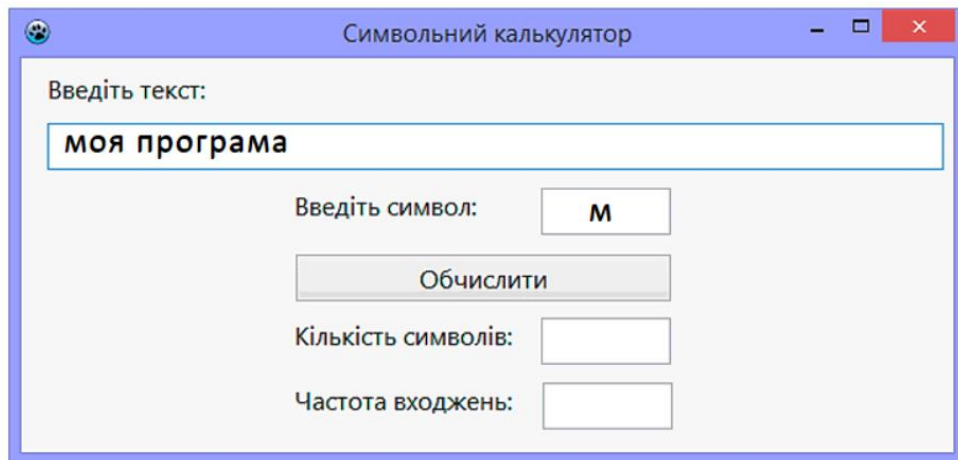
Символ для підрахунку

б м

Кількість входжень символу

п 0

Слайд № 9



i	1	2	3	4	5	6	7	8	9	10	11	12
а	м	о	я		п	р	о	г	р	а	м	а

=

м

False

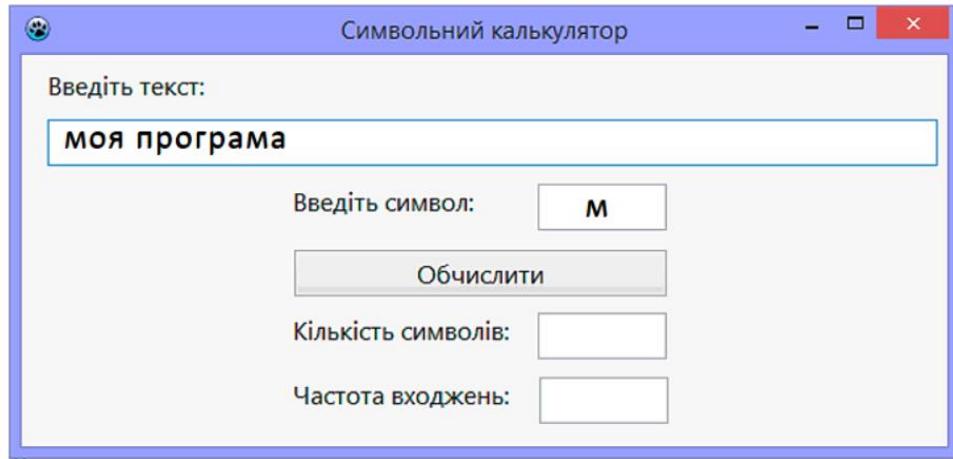
Символ для підрахунку

б м

Кількість входжень символу

п 1

Слайд № 10



i	1	2	3	4	5	6	7	8	9	10	11	12
а	м	о	я		п	р	о	г	р	а	м	а

=  
м  
**True**

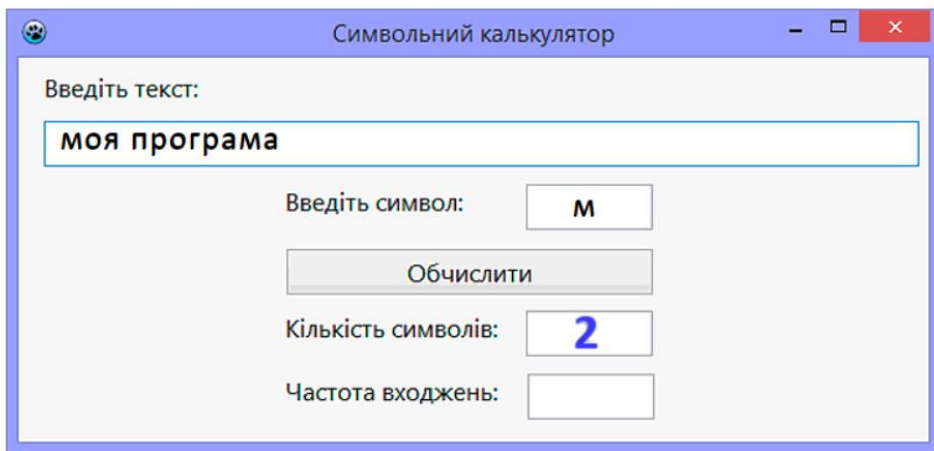
Символ для підрахунку 

б	м
---	---

Кількість входжень символу 

п	1
---	---

Слайд № 11



i	1	2	3	4	5	6	7	8	9	10	11	12
а	м	о	я		п	р	о	г	р	а	м	а

=  
м  
**False**

Символ для підрахунку 

б	м
---	---

Кількість входжень символу 

п	2
---	---

Дійсно, для підрахунку кількості входжень символу тіло циклу має бути виконано стільки разів, скільки символів містить рядок.

Отже, перед циклом потрібно обчислити довжину рядка, що визначається за допомогою функції `Length`.

Наприклад, команда  
`n:=Length(a);`  
визначає довжину рядка `a`

## Вправа

### Вправа № 2

#### Вправа 2 у Lazarus

Виконайте оголошення змінних для проекту

“Символьний калькулятор”:

`a` – текстовий рядок;

`b` – символ, кількість якого буде підраховуватися;

`n` – кількість символів рядка;

`i` – порядковий номер символу рядка;

`x` – кількість входжень символу `b` в рядок `a`.

### Підказка до вправи № 2

#### Підказка до вправи 2

Оголошення змінних починається зі службового слова `Var`



Вправа № 3

Вправа 3 у Lazarus

1. Запрограмуйте обробник події натискання кнопки **Обчислити** для підрахунку кількості входжень заданого символу в рядок.
2. Запустіть проект та перевірте правильність його виконання.

Вправа № 4

Вправа 4 у Lazarus

1. Доповніть код, щоб можна було обчислювати кількість входжень символу скільки завгодно разів.
2. Запустіть проект та перевірте правильність його виконання.

Вправа № 5

Вправа 5 у Lazarus

1. Допишіть до обробника натискання кнопки код обчислення частоти входжень символу.
2. Запустіть проект та перевірте правильність його виконання.







Слайд № 16

### Зітремо напис у полі Memo

Властивості

DragMode	dmManual
Enabled	<input checked="" type="checkbox"/> (True)
Font	(TFont)
Height	308
HelpContext	0
HelpKeyword	
HelpType	htContext
HideSelection	<input checked="" type="checkbox"/> (True)
Hint	
Left	20
Lines	(TStrings)
MaxLength	0

Оберіть властивість **Lines**

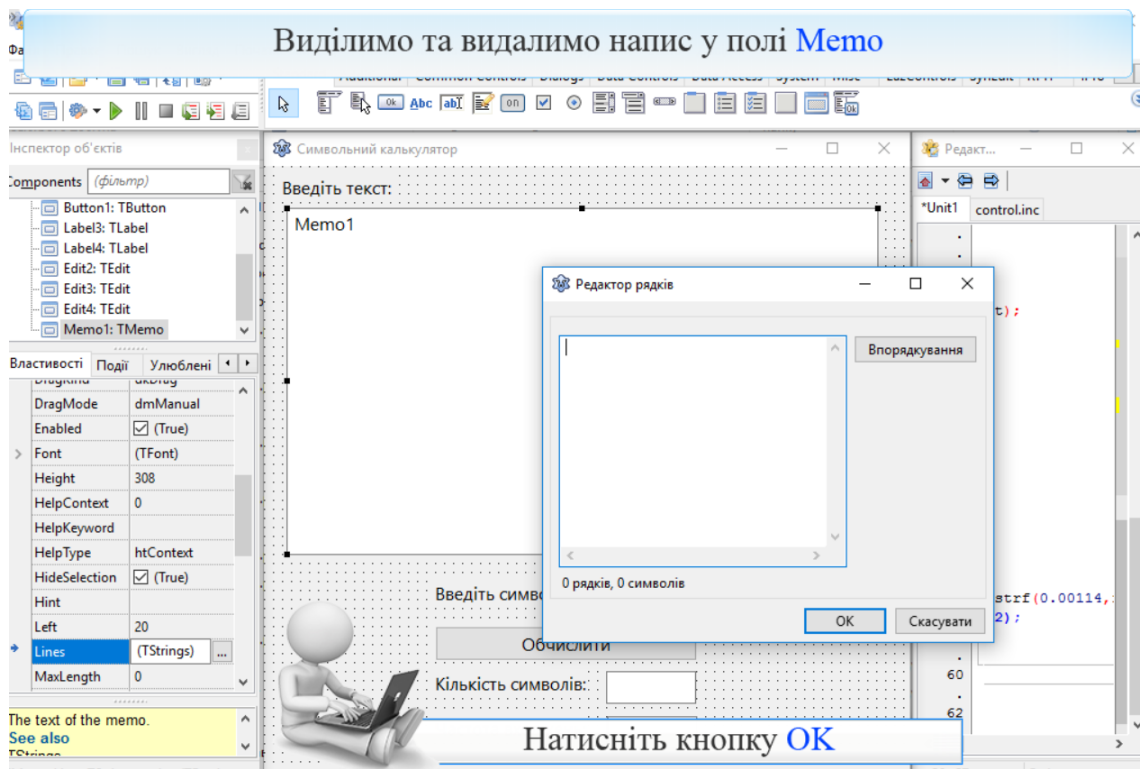
Слайд № 17

Властивості

DragMode	dmManual
Enabled	<input checked="" type="checkbox"/> (True)
Font	(TFont)
Height	308
HelpContext	0
HelpKeyword	
HelpType	htContext
HideSelection	<input checked="" type="checkbox"/> (True)
Hint	
Left	20
Lines	(TStrings) ...
MaxLength	0

Клацніть кнопку (...) біля властивості **Lines**

## Слайд № 18



Напис у полі **Мемо** видалено.

## Вправа

### Вправа № 7

## Вправа 7 у Lazarus

1. Видаліть з форми елемент **Edit1** і замість нього додайте елемент **Memo1**.
2. Видаліть текст з **Мемо1**.

Отже, текст із поля **Memo** міститься у властивості **Lines**.

Щоб змінній **a** присвоїти весь текст поля **Memo1**,  
використовують команду

```
a:=Memo1.Lines.Text;
```

або просто

```
a:=Memo1.Text;
```

### Вправа 8 у Lazarus

1. Змініть код присвоєння змінній **a** введеного тексту (змінна **a** має одержувати значення тексту поля **Memo1**).
2. Запустіть проект та перевірте правильність його виконання.