

Практична робота №11

Тема: Обробка одновимірних масивів: введення та виведення даних, пошук.

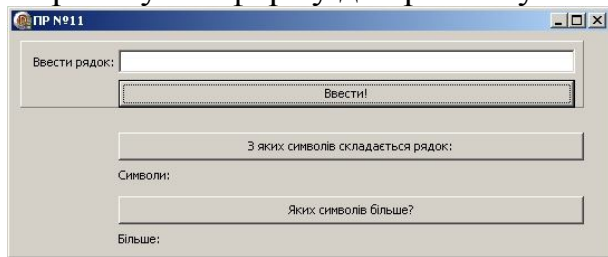
Мета: Навчитися програмній обробці масивів.

Хід роботи

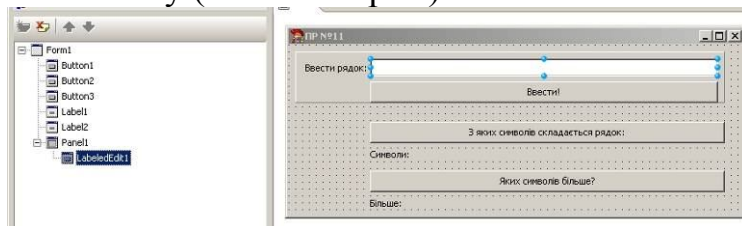
1. Створити у своїй робочій папці каталог PRACT11.
2. Запустити середовище Delphi та створити новий додаток.
3. Виконати File – Save All, вказавши пунктом призначення створений Вами каталог PRACT11, під іменами, які пропонує середовище Delphi.

Умова задачі. Потрібно ввести в програму рядок та визначити, з яких символів він складається, порівняти, яких символів більше – «а» та «о» чи «е» та «и».

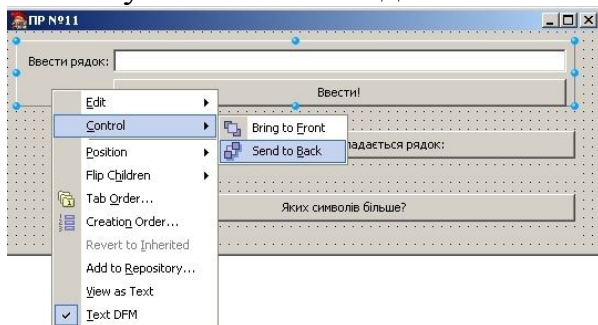
Спроекуємо форму для розв'язування цієї задачі:



Важливо! На формі розмістимо панель Panel1, на якій помістимо елемент для введення тексту (із коментарем) Labelededit1 та кнопку «Ввести!» (Button1).



Розташування панелі відносно Labelededit1 та Button1 – на задньому плані.



Окрім перерахованих елементів у нашій програмі будуть використовуватися ще дві кнопки – «З яких символів складається рядок:» та «Яких символів більше?», а також два поля для виведення тексту «Символи: » та «Більше: ».

4. В інтерфейсній частині проекту опишемо необхідні масиви:

```
var
  Form1: TForm1;
  s: string;
  a: array [char] of integer;

implementation
  {$R *.dfm}
```

5. Внесемо програмний код в обробник події OnClick першої кнопки «Ввести!»:

```
procedure TForm1.Button1Click(Sender: TObject);  
var i: integer;  
    c: char;  
begin  
    s:=labelededit1.text;  
    panel1.enabled:=false;  
    for i:=1 to length(s) do  
        begin  
            c:=s[i];  
            a[c]:=a[c]+1;  
        end;  
    end;  
end;
```

6. В обробник події OnClick другої кнопки «З яких символів складається рядок:» внесемо наступний програмний код:

```
procedure TForm1.Button2Click(Sender: TObject);  
var c: char;  
begin  
    label1.Caption := 'Символи: ';  
    for c:=#0 to #255 do  
        if a[c]>0 then  
            label1.Caption:=label1.Caption + c + ' ';  
        end;  
    end;
```

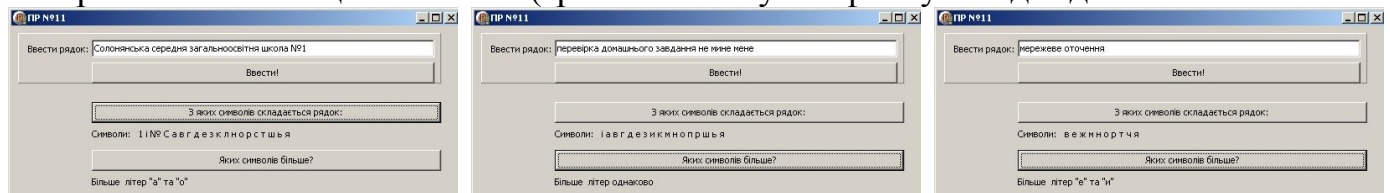
7. Внесемо програмний код в обробник події OnClick третьої кнопки «Яких символів більше?»:

```
procedure TForm1.Button3Click(Sender: TObject);  
var sum1, sum2: integer;  
begin  
    sum1:=a['a']+a['o'];  
    sum2:=a['e']+a['и'];  
    label2.Caption := 'Вільше: ';  
    if sum1>sum2 then  
        label2.caption:=label2.caption+' літер "а" та "о";  
    else  
        if sum2>sum1 then  
            label2.caption:=label2.caption+' літер "е" та "и";  
        else  
            label2.caption:=label2.caption+' літер однаково';  
        end;  
    end;
```

8. Збережемо отриманий додаток на диску та запустимо його на виконання.

9. Завершити роботу із середовищем Delphi.

10. Із операційної оболонки запустити файл із розширенням exe, який утворився у створеній Вами папці PRACT11 (зробити 3 запуски файлу. Вхідні дані:



11. Результати роботи програми продемонструвати викладачу.

12. Навести порядок на робочому місці.

Деякі зауваження. У цій задачі розглядається рядковий тип даних (string). Він являє собою одновимірний масив, кожний елемент якого є символом (char). Це досить часто використовуваний тип, так як величезний обсяг інформації людина отримує не у вигляді чисел, а у вигляді текстових даних. Для роботи із рядковим типом як із масивом необхідно знати кількість елементів у рядку (символів – довжину рядка). Це пов'язано із тим, що рядки є динамічним типом – їх фактична довжина визначається в операторі присвоєння. Кількість символів у рядку визначається за допомогою функції Length (рядок).

Розміщення поля для введення Labelededit1 та кнопки Button1 на панелі Panel1 дозволяє заборонити доступ (по завершенні введення) до всіх компонентів шляхом присвоєння Panel1.Enabled:=False

Заповнення масиву a. Цілочисельний масив описано у «верхньому» розділі var, тому він є глобальним, значить, всі його елементи за замовчуванням рівні 0. Індексом масиву виступають символи (їх 256 значень, - від 0 до 255). Таким чином, у циклі For ми послідовно отримуємо доступ до всіх символів введеного рядка і в залежності від того, який символ розглядається, збільшуємо відповідне значення масиву (лічильник) на одиницю.

Обробник події OnClick компоненту Button2 передбачає перебір всіх можливих символів і виведення в Label1 тих, які зустрілися хоча б один раз.

Обробник події OnClick компоненту Button3 визначає дві суми sum1 та sum2. У подальшому відбувається їх порівняння. Зверніть увагу на механізм порівняння значень двох змінних. Можливі три варіанти результату – перше значення більше, друге значення більше чи значення рівні. Запропоноване рішення із використанням вкладених операторів If ... Then ... Else, з точки зору оптимізації коду, є найкращим.