

Практична робота №8

Тема: Використання циклів.

Мета: Навчитись використовувати цикли при розробці додатків.

Хід роботи

1. Створити у своїй робочій папці каталог PRACT08.
2. Запустити середовище Delphi та створити новий додаток.
3. Виконати File – Save All, вказавши пунктом призначення створений Вами каталог PRACT08, під іменами, які пропонує середовище Delphi.

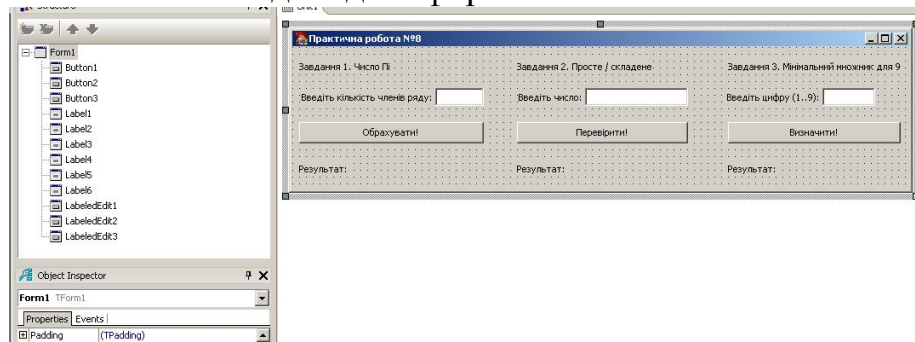
Умова задачі. Створити форму для виконання наступних завдань:

Завдання 1. За допомогою ряду Тейлора визначити число π .

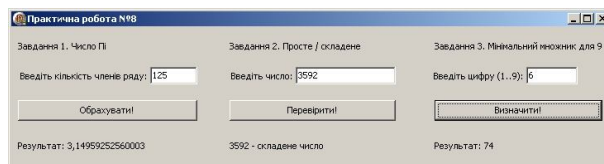
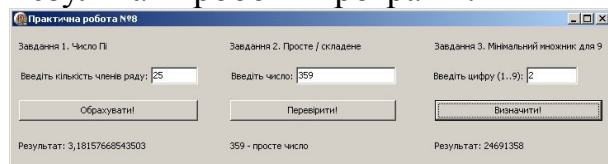
Завдання 2. Ввести ціле число. Визначити, чи воно є простим, чи складеним.

Завдання 3. Визначити, на який мінімальний множник потрібно помножити число 9, щоб запис результату складався лише із однакових цифр (від 1 до 9), які визначаються користувачем.

Зовнішній вигляд вхідної форми:



Результати роботи програми:



4. На заготовку форми, яка з'явилася у вікні, потрібно розмістити 3 кнопки LabeledEdit (для введення чисел), а також 6 кнопок Label (для виведення назви та результатів виконання завдань) та 3 кнопки Button («Обрахувати!», «Перевірити!», «Визначити!»).

5. В даній задачі програмується тільки обробник події OnClick кнопок Button.

Наведемо деякі пояснення щодо розв'язування окремих завдань.

Завдання 1. На початку XVIII століття англійський математик Брук Тейлор запропонував ряд формул для розкладання функцій в степеневі ряди. Використання цього відкриття зіграло важливу роль в реалізації на обчислювальних машинах різних функцій. Замість зберігання величезних таблиць значень функцій при різних аргументах в бібліотеці мови програмування може бути записаний лише алгоритм визначення результуючого значення шляхом обрахунку нескладної формули – суми обмеженої кількості елементів ряду (числової послідовності).

Найбільш відомою формулою (вона отримала назву «ряд Тейлора») є формула для визначення функції $\arctg(x)$:

$$\arctg(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \frac{x^9}{9} - \dots$$

Якщо в формулу поставити $x=1$, то

$$\arctg(1) = \frac{\pi}{4}$$

Тоді формула набуває наступного вигляду:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots$$

або

$$\pi = 4 \cdot \left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots\right)$$

Багато поколінь програмістів користувалися цим рядом для визначення числа π із заданою точністю.

```
procedure TForm1.Button1Click(Sender: TObject);
var i, k, zn: integer;
    s: real;
begin
  k:=strtoint (labelededit1.text);
  s:=0;
  zn:=1;
  for i:=1 to k do
  begin
    s:=s+zn*(1/(2*i-1));
    zn:=zn*(-1);
  end;
  label2.caption:='Результат: '+floattostr(4*s);
end;
```

Наведемо деякі пояснення до даного фрагменту програми. Зверніть увагу на типи даних: кількість членів ряду, параметр циклу та знак доданку – цілочисельні змінні, сума ряду, в якій накопичуються значення простих дробів – дійсна величина. Знак чергового доданку визначається окремою змінною, яка дає для парних елементів значення «-1», а для непарних – значення «+1». Оскільки в циклі після оператора For виконуються відразу два оператори, то вони поміщені в операторні дужки begin ... end;

Завдання 2. Ще одним популярним алгоритмічним додатком циклів є перевірка на подільність: досліджуване ціле число послідовно ділять на ряд натуральних чисел (як правило, від 2 до $n-1$), після чого перевіряють залишок від ділення – якщо ніде не з'явився залишок, рівний 0, то значить, що число просте.

```
procedure TForm1.Button2Click(Sender: TObject);
var i, k: integer;
    s: ansistring;
begin
  k:=strtoint(labelededit2.Text);
  s:=' - просте число';
  for I := 2 to k - 1 do
  if k mod i = 0 then
    s:=' - складене число';
  label4.Caption:=inttostr(k)+s;
end;
```

Завдання 3. Розв'язувати задачу доцільно від зворотного: припустимо, що ми відшукали число, що складається із однакових цифр, яке ділиться на 9. Тоді із усіх можливих таких чисел нам потрібно вибрати мінімальне. Реалізуємо це програмним шляхом. Виникає одне запитання – як зібрати числа, що складаються із однакових цифр, наприклад, із самих лише двійок: 2, 22, 222, 2222 тощо? Робиться це досить просто, проте потрібно слідкувати за межами цілочисельних типів даних – числа

можуть набувати величезних значень. Тому скористаємось типом даних `longint` (або `cardinal`).

```
procedure TForm1.Button3Click(Sender: TObject);  
var chislo, chislo1, stepen: longint;  
begin  
  chislo:=strtoint(labelededit3.Text);  
  chislo1:=chislo;  
  stepen:=10;  
  while chislo mod 9 <>0 do  
  begin  
    chislo:=chislo+chislo1*stepen;  
    stepen:=stepen*10;  
  end;  
  label6.caption:='Результат: '+inttostr(chislo div 9);  
end;
```

6. Збережемо отриманий додаток на диску та запустимо його на виконання.
7. Завершити роботу із середовищем Delphi.
8. Із операційної оболонки запустити файл із розширенням `exe`, який утворився у створеній Вами папці `PRACT08`. Результат роботи програми продемонструвати викладачу.
9. Навести порядок на робочому місці.