**Цикл з післяумовою**

Оператори **while** і **repeat** використовуються в тому випадку, коли деяку послідовність дій треба виконати кілька разів, причому необхідне число повторень під час розробки програми невідомо і може бути визначено тільки під час роботи програми.

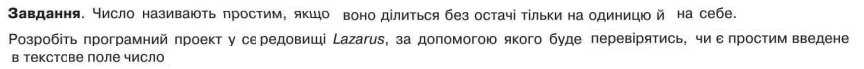
**Оператор циклу із післяумовою *repeat ... until***  
Загальний вигляд оператора:  
**repeat**  
    <Тіло циклу>  
**until** ***умова***

                                де ***умова*** - вираз логічного типу, визначальна умова*завершення* циклу.

Оператори тіла циклу виконуються доти, поки умова не стане істинною.  
  
Зверніть увагу на відмінності:

* в операторі while ... do вихід з циклу відбувається тоді, коли умова стане хибною,
* а в операторі repeat ... until - коли умова стане істинна.

**Завдання 1**. **[Просте число.](https://cloud.mail.ru/public/5UZN/uuzf9m2Yx)**



|  |  |  |
| --- | --- | --- |
| **Алгоритм виконання.**  **http://inform-school.ucoz.ua/zikl_lazarus/cikl2.png** | **Варіант інтерфейсу проекту**  **http://inform-school.ucoz.ua/zikl_lazarus/cikl4.png** | **Приклад виконання**  http://inform-school.ucoz.ua/zikl_lazarus/cikl3.png |

**Варіант програмного коду**

procedure**TForm1.Button1Click**(Sender: TObject);  
 var  i,n, k : integer;  
**begin**  
  n:= StrToInt(edit1.Text);  
   i:=1;

**repeat**  //начало цикла  
   i:=i+1;    //формирование следующего делителя  
   k:= n mod i;    // запоминание очередного остатка  от деления  
   listbox1.Items.Add('i = '+ inttostr(i) + '    ' + inttostr(n)+ ' mod '+ inttostr(i)+ ' =  '+ inttostr(k));  
              //вывод текущих значений переменных : i, k  
  **until** **n mod i = 0**;            //условие выхода из цикла

   if  n = i  then label2.Caption:= ' введенное число простое'  
   else label2.Caption:=' введенное число не простое';  
**end;**

**Вправа 1.** Вивести парні числа від  2 до 12.

|  |  |
| --- | --- |
| **Алгоритм проекта**  http://inform-school.ucoz.ua/zikl_lazarus/until1_interf.png | **Программный код проекта**  procedure TForm1.Button1Click(Sender: TObject); var n: Integer; begin                 n := 0;                 Repeat                 n := n + 2;                 listbox1.Items.add(' n = ' + inttostr(n));                 Until  n => 12; end;  procedure TForm1.Button2Click(Sender: TObject); begin    listbox1.Clear; end; |

**Вправа 2**. Знайти суму цифр натурального числа n, кількість цифр у якому невідома.

|  |  |
| --- | --- |
| **Алгоритм проекта**  http://inform-school.ucoz.ua/zikl_lazarus/until3_interf.png | **Вариант программного кода**  procedure **TForm1.Button1Click**(Sender: TObject);  var n, d, s: integer;  **begin**   n := StrToint(inputbox('введите число','',''));    label1.caption :=  inttostr(n);   s:= 0;   d:= 0;    Memo1.Lines.Add('n ='+#9 +inttostr(n) +#9 +'d = ' + inttostr(d) +#9+'s = ' + inttostr(s));  ***Repeat  //***начало цикла  d := n mod 10; // выделяется последняя цифра числа  s := s+d; // значение выделенной цифры  добавляется к сумме цифр числа  n := n div 10; // отбрасывается последняя цифра числа n  Memo1.Lines.Add('n ='+#9 +inttostr(n) +#9 +'d = ' + inttostr(d) +#9+'s = ' + inttostr(s));       Application.ProcessMessages; // обработка сообщений     Sleep(2000); // пауза на 2 с     ***Until***n=0;  // если  n = 0, выполняется  выход из цикла **end;**  #9 – табуляция |