

## Урок №25

**Тема уроку:** Оператори циклів. Складання та програмна реалізація алгоритмів із циклами.

**Мета:**

**Навчальна:** сформувати поняття: цикл; оператори повторення;

**Розвивальна:** формувати вміння: працювати в середовищі програмування; створювати проект та налагоджувати його; використовувати цикли та оператори повторення; редагувати код обробників подій; використовувати середовище програмування для створення проектів; створювати проект для власного використання; чітко і лаконічно висловлювати думки; робити висновки;

**Виховна:** виховувати: уважність, дисциплінованість під час роботи на ПК.

**Тип пари:** лекція.

**Базові поняття й терміни:** цикл; повторення.

**Структура пари:**

I. Організаційний етап

II. Перевірка домашнього завдання

III. Актуалізація опорних знань

IV. Мотивація навчальної діяльності

V. Сприйняття та усвідомлення нового матеріалу

1. Цикл **FOR**

2. Цикл **WHILE**

3. Цикл **REPEAT**

VI. Первинне закріплення знань

VII. Підбиття підсумків уроку

VIII. Домашнє завдання

**Хід уроку**

**I. ОРГАНІЗАЦІЙНИЙ ЕТАП**

Привітання. Підготовка класу до занять. Оголошення теми, мети і завдань уроку.

## **II. ПЕРЕВІРКА ДОМАШНЬОГО ЗАВДАННЯ**

## **III. АКТУАЛІЗАЦІЯ ОПОРНИХ ЗНАНЬ**

Пошукова робота

Використовуючи Інтернет, знайдіть інформацію про «Базові структури програмування».

1. Які структури вам відомі?
2. Яке їх призначення?

## **IV. МОТИВАЦІЯ НАВЧАЛЬНОЇ ДІЯЛЬНОСТІ**

Розглянемо задачу. Обчислити дохід з депозиту, якщо відомо відсоткову ставку, початкову суму та термін вкладу (відсотки нараховуються щомісячно та додаються до суми депозиту).

На перший погляд, нескладна задача. Але якщо термін складає 10 років, 20 років, скільки однотипних дій потрібно виконати? На допомогу приходять цикли.

## **V. СПРИЙНЯТТЯ ТА УСВІДОМЛЕННЯ НОВОГО МАТЕРІАЛУ**

Циклом називають послідовність дій (операторів), яку записують у тексті програми один раз, але вона може виконуватися понад один раз. У циклічних алгоритмах **тіло циклу** виконується ніби по колу.

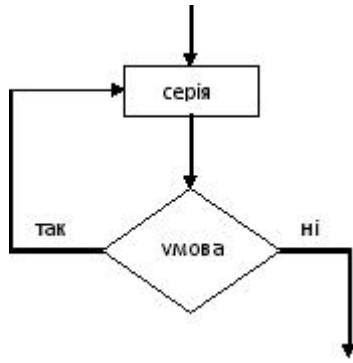
Повторення відбувається доти, доки не буде виконано заздалегідь визначену кількість повторів або доки не справдиться (чи порушиться) умова, записана у циклічному операторі. Цю умову можна записувати на початку або в кінці оператора повторення.

Тому цикли поділили на цикли:

- 1) **з передумовою** (умову записують та перевіряють на початку циклу)
- 2) **з післяумовою** (умову записують та перевіряють у кінці циклу)
- 3) **цикли із заздалегідь відомою кількістю повторів** (коли відомо, скільки разів ти маєш виконувати однакові дії).

**Цикл WHILE (з передумовою).**

Привести блок-схему циклу з передумовою.



В цьому циклічному операторі спочатку перевіряється умова  $i$ , якщо вона істинна, то вказана дія виконується черговий раз, якщо ж ні – то виконання дії припиняється.

Тому цей оператор називають циклом з передумовою. Перед кожним виконанням тіла циклу обчислюється значення умови його виконання. Формат оператора циклу з передумовою має вигляд:

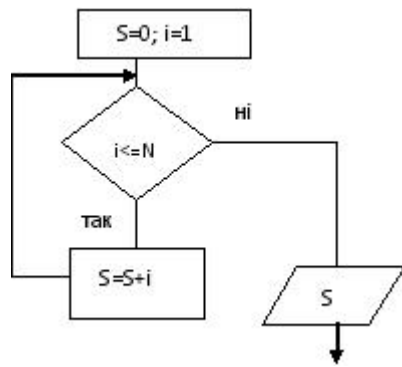
**While <умова> do begin <оператор> end;**

Оператор працює так: перевіряється умова, якщо вона вірна, то виконується оператор, який стоїть після do і знову перевіряється умова; так повторюється до тих пір, поки умова не стане невірною - тоді виконується наступний оператор програми.

Можливі ситуації, коли «цикл поки» не виконується жодного разу. Це відбувається в тому випадку, коли на першому кроці умова є хибною. Якщо при повторенні циклу умова залишається завжди істинною, то цикл може повторюватись нескінченно.

**Приклад1.** Алгоритм підрахунку суми  $N$  перших натуральних чисел.

Суму позначимо через  $S$ , через  $i$  – черговий доданок. Спочатку  $S=0$ , оскільки ще суми не знаходили,  $i=1$  (перше натуральне число). Щоб знайти суму, то потрібно до попередньої суми додати наступний доданок:  $S=S+i$ . Для отримання наступного числа потрібно попереднє збільшити на одиницю:  $i=i+1$ . Виконання циклу продовжується до тих пір, поки  $i \leq N$ .



### Цикл REPEAT (з післяумовою).

У циклічному операторі повторення з післяумовою спочатку виконується серія команд, а після цього перевіряється умова і визначається, чи є потреба виконувати її знову.

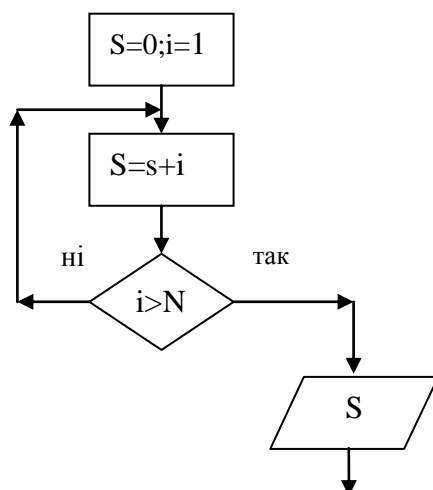
Умова, яка перевіряється, є умовою закінчення повторень (тобто умовою виходу із циклу). Формат оператора Repeat такий:

Repeat <оператори> until <умова>;

Оператор працює так: виконуються оператори, що стоять після службового слова Repeat до службового слова Until, потім перевіряється умова; якщо вона невірна - оператори виконуються знову; так повторюється до тих пір, поки умова не стане вірною - тоді виконується наступний оператор програми.

### Приклад 2. Алгоритм підрахунку суми N перших натуральних чисел.

Для підрахунку суми N перших натуральних чисел використаємо оператор repeat. Позначення змінних та алгоритм підрахунку аналогічні прикладу 1. Різниця в тому, що умовою виходу з циклу є  $i > N$ .



## **Цикл FOR**

Оператор циклу з параметром використовують у тих випадках, коли деякі дії треба повторювати і точно відома кількість повторень. Формат оператора For такий:

```
For <параметр> := <початкове значення> to <кінцеве значення>  
do begin <оператор> end;
```

АБО

```
For <параметр> := <початкове значення> downto <кінцеве  
значення>do begin <оператор> end;
```

Оператор працює так: змінній-параметру присвоюється початкове значення, виконується оператор, що стоїть після службового слова do, потім значення параметру збільшується (при використанні to), або зменшується (при використанні downto) на одиницю.

Від початкового і кінцевого значення параметру залежить, скільки разів буде виконуватися цикл.

Змінна-параметр може бути тільки будь-якого цілочисельного типу.

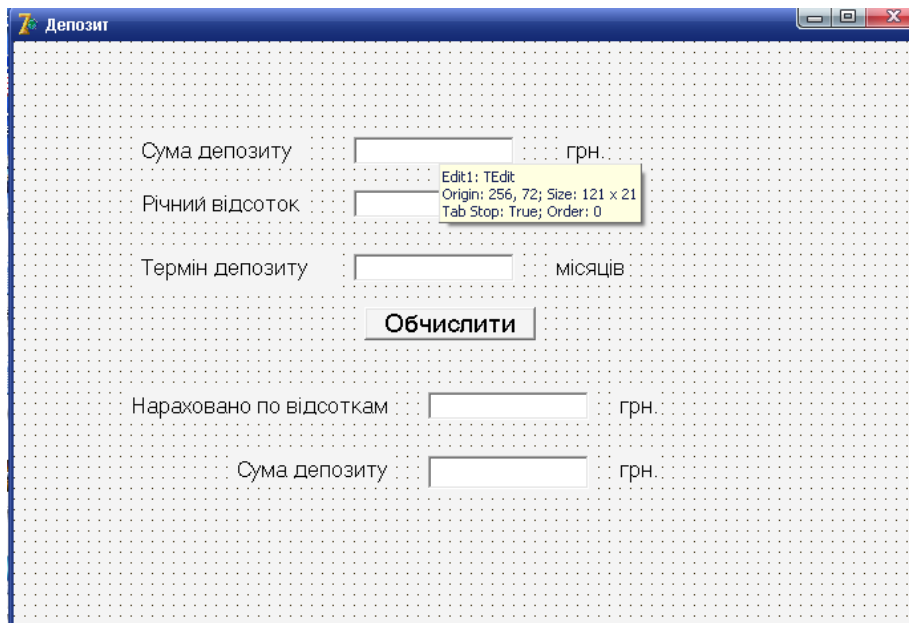
## **VI. ПЕРВИННЕ ЗАКРІПЛЕННЯ ЗНАНЬ**

### **Практичне завдання.**

Створимо проект, про який ми говорили на початку уроку. Обчисліть дохід з депозиту, якщо відомо відсоткову ставку, початкову суму та термін вкладу, враховуючи, що відсотки нараховуються щомісячно та додаються до суми депозиту.

Інструктивна картка.

1. Запустити середовище програмування Delphi 7.
2. Створити проект із назвою Project-Depoz.
3. Додати необхідні об'єкти на форму та встановити їх властивості.
4. Написати код обробників подій.
5. Запустити проект на виконання та перевірити його працездатність.
6. Зберегти проект.



### Код проекту

```

depoz.pas
depoz
var sp,t,k:integer;suma,v,sv:real;
begin
  sp:=strtoint(edit1.Text);
  v:=strtoint(edit2.Text);
  t:= strtoint(edit3.Text); k:=1;
  suma:=sp; v:=v/1200;
  while (k<=t) do begin
    suma:=suma+suma*v;
    k:=k+1;end;
    sv:=suma-sp;
    Edit4.Text:=FloatToStrF(sv,ffGeneral,10,4);
    Edit5.Text:=FloatToStrF(suma,ffGeneral,10,4);
  end;
end.

```

**VII. ПІДБИТТЯ ПІДСУМКІВ УРОКІВ.** Викладач оцінює роботу студентів на парі.

**VIII. ДОМАШНЄ ЗАВДАННЯ.** 1.Опрацювати конспект та відповідний розділ підручника.