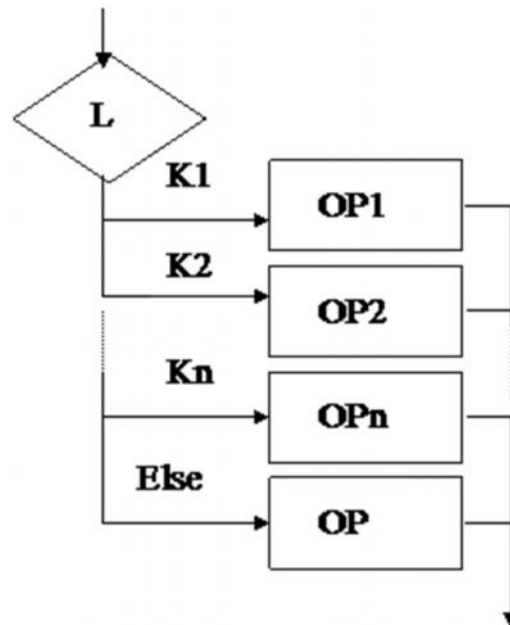




# Тема уроку



## Вкладені розгалуження в Pascal



# *План уроку*

- 1. Перевірка виконання домашнього завдання: варіанти 1-15.**
- 2. Вкладені розгалуження в Pascal.**
- 3. Приклади програм.**
- 4. Підсумки уроку та домашнє завдання.**
- 5. Практична робота (домашнє завдання).**
- 6. Підсумки уроку та домашнє завдання.**

# Вкладені розгалудження в Pascal

---

**Задача.** Фірма набирає співробітників від 25 до 40 років включно. Ввести вік людини і визначити, чи підходить вона фірмі (вивести відповідь “підходить” або “не підходить”).

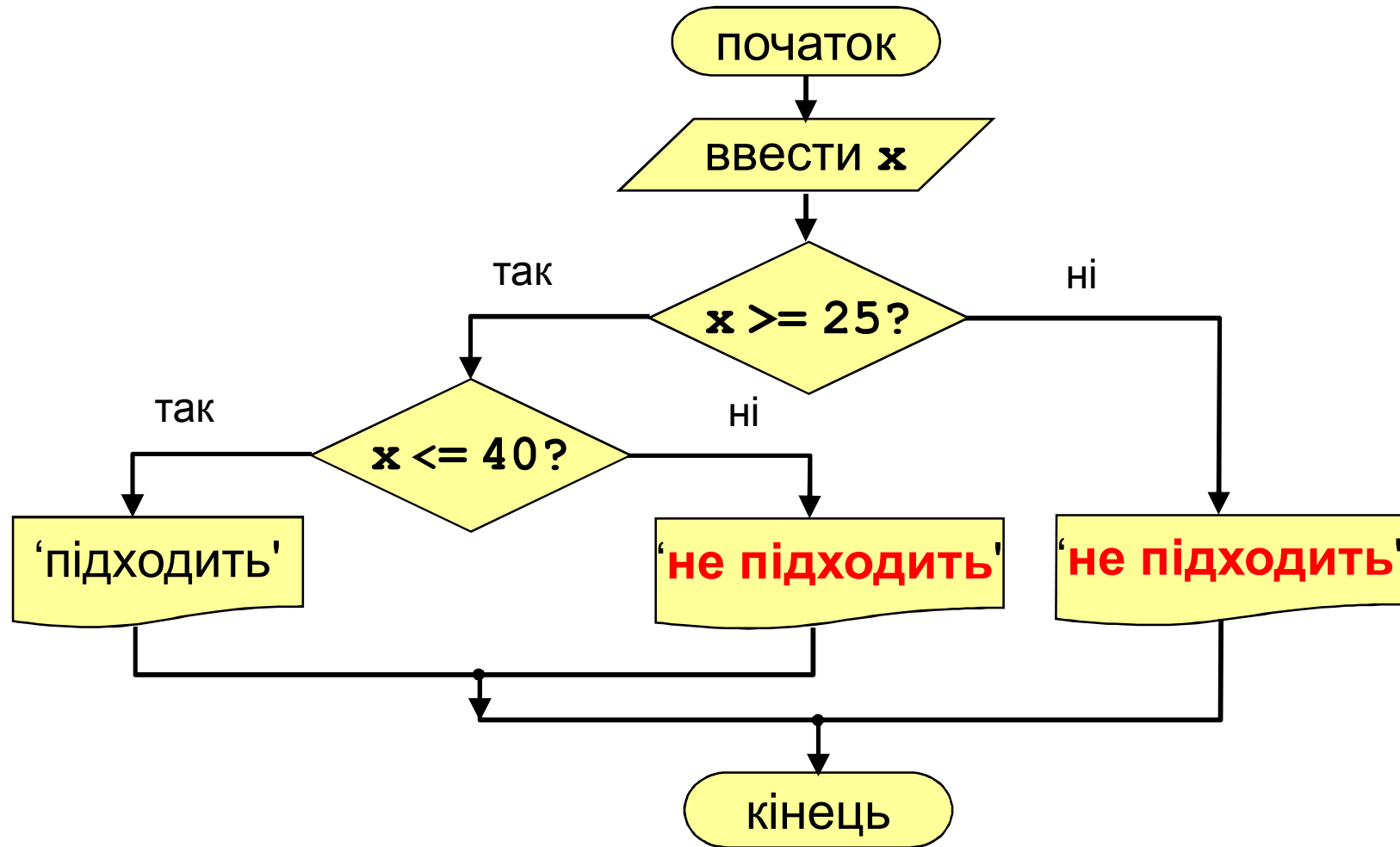
**Особливості:** потрібно перевірити, виконання двох умов одночасно.



Чи можна розв'язати відомими методами?

# Варіант 1. Алгоритм

---



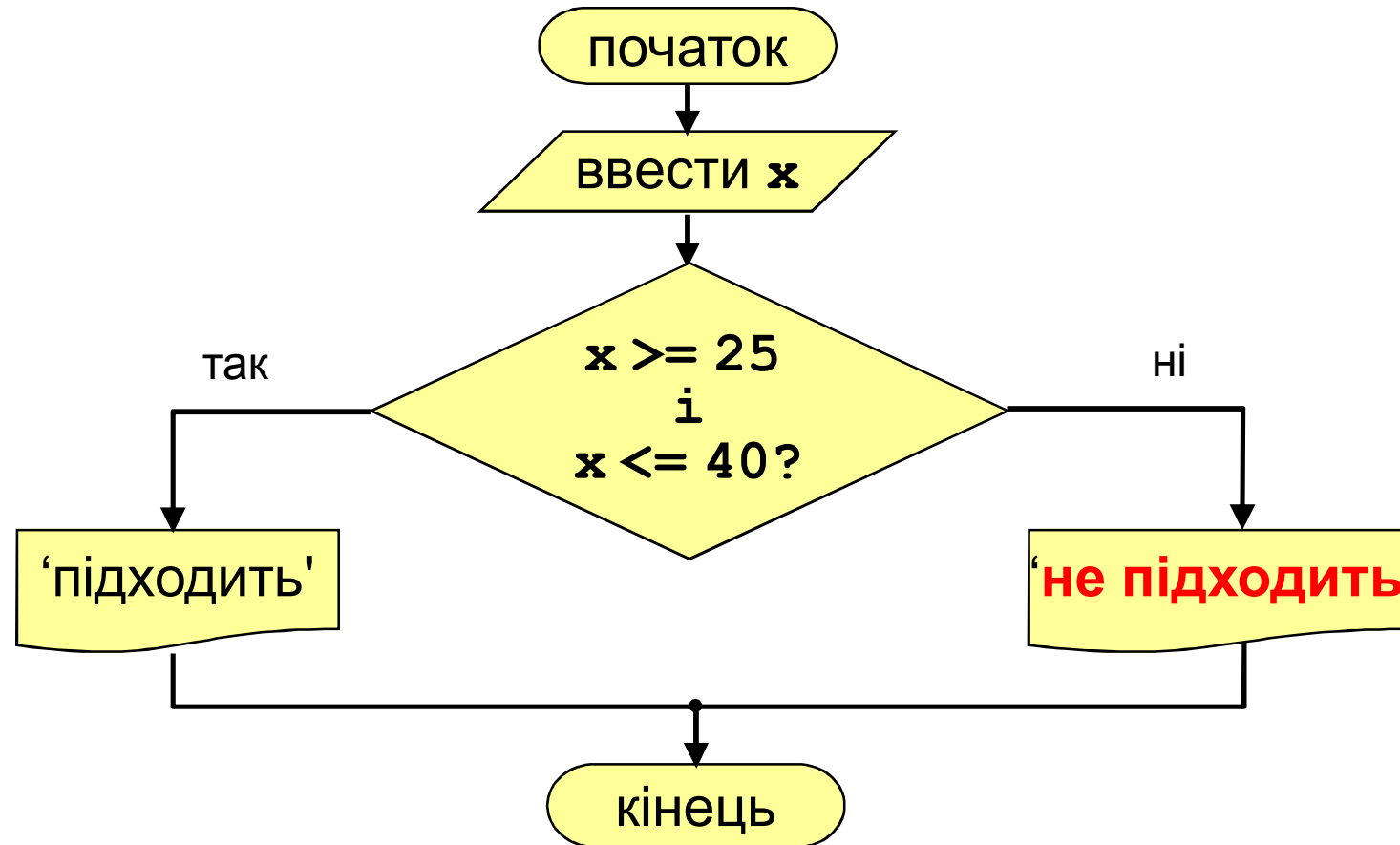
# Варіант 1. Програма

---

```
program qq;  
var x: integer;  
begin  
    writeln('Ввести вік');  
    read ( x );  
    if x >= 25 then  
        if x <= 40 then  
            writeln ('Підходить')  
        else writeln ('Не підходить')  
    else  
        writeln ('Не підходить');  
end.
```

# Варіант 2. Алгоритм

---



## Варіант 2. Програма

---

```
program qq;  
var x: integer;  
begin  
  writeln('Ввести вік');  
  read ( x );  
  if (x >= 25) and (x <= 40) then  
    writeln ('Підходить')  
  else writeln ('Не підходить')  
end.
```

складена  
умова

# Складена умова

---

**Складена умова** – це умова, яка складається з декількох простих умов (відношень), зв'язаних з допомогою логічних операцій:

- **not** – Ні (заперечення, інверсія)
- **and** – І (логічне множення, кон'юнкція, одночасне виконання умов)
- **or** – АБО (логічне додавання, диз'юнкція, виконання хоча б одної з умов)
- **xor** – виключаюче АБО (виконання тільки одної з двох умов, але не обох)

**Прості умови (відношення)**

<

<=

>

>=

дорівнює

=

не дорівнює

<>



# Складена умова

---

## Порядок виконання

- вирази в дужках
- not
- and
- or, xor
- <, <=, >, >=, =, <>

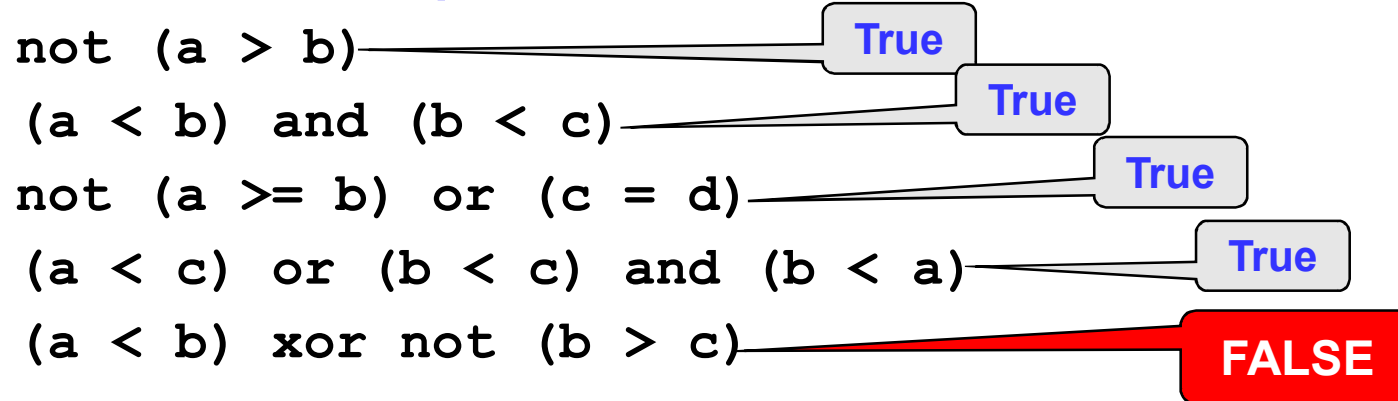
**Особливості** – кожна з простих умов обов'язково береться в дужки.

## Приклад

```
      4      1      6      2      5      3  
if not (a > b) or (c <> d) and (b <> a)  
then begin  
    ...  
end
```

# Складена умова

Істинне чи хибне при  $a := 2; b := 3; c := 4;$



Для яких значень  $x$  істинні умови:

- $(x < 6) \text{ and } (x < 10)$
- $(x < 6) \text{ and } (x > 10)$
- $(x > 6) \text{ and } (x < 10)$
- $(x > 6) \text{ and } (x > 10)$
- $(x < 6) \text{ or } (x < 10)$
- $(x < 6) \text{ or } (x > 10)$
- $(x > 6) \text{ or } (x < 10)$
- $(x > 6) \text{ or } (x > 10)$

$(-\infty, 6)$	$x < 6$
$\emptyset$	
$(6, 10)$	
$(10, \infty)$	$x > 10$
$(-\infty, 10)$	$x < 10$
$(-\infty, 6) \cup (10, \infty)$	
$(-\infty, \infty)$	
$(6, \infty)$	$x > 6$

# Команда вибору

Коли багато різних варіантів значень величини та відповідно дій, треба використовувати команду вибору:

**CASE** величина **OF**

значення 1 : команда 1;

значення 2 : команда 2;

.....

Значення n : команда n;

**ELSE** команда;

**END;**

- Величина – це параметр за значенням якого і виконується та чи інша команда.

# Команда вибору

Команда вибору цікава тим, що **можуть виконатися декілька команд**, якщо величина приймає значення, що зустрічаються декілька разів.

Якщо **жодне із значень величини не виконується**, то ПК виконує команди, що містяться після **ELSE**.

Тип величини, по якій ведеться вибір, повинен бути тільки **Integer**.

Якщо ми хочемо вказати **діапазон значень** величини, то вказуємо його через дві крапки (наприклад: 1..10 або 100..200).

# Задача визначення категорії оцінювання за введеним числом

Користувач вводить цифру, а ПК визначає до якої категорії оцінювання відноситься така оцінка.

Якщо від 1 до 3 – погано, від 4 до 6 – посередньо, від 7 до 9 – добре, від 10 до 12 – відмінно. Врахувати введення чисел, які не можуть бути оцінкою.



# Програма

```
Program OCINKA;  
  Var a:integer; -----(величина для вибору цілого типу)  
Begin  
  Writeln('Введіть Вашу оцінку');  
  Readln(a); -----(величина-оцінка, по якій буде вибір дій)  
  Case a of  
    1..3 : Writeln('Погано');  
    4..6 : Writeln('Посередньо');  
    7..9 : Writeln('Добре');  
    10..12 : Writeln('Відмінно');  
  Else Writeln('Такої оцінки немає'); ----(якщо жодне із  
                                           значень не підходить)  
  End;  
ReadKey;  
End.
```

## Задача: Представити дійсне число в грошовому форматі

```
program tm1;
  Uses crt;
  Var a,b:word; r:string; x:real;
begin
  clrscr;
  write ('Введіть число - ');
  readln (x);
  a:=trunc (x);
  if a div 10 mod 10 <>1 then
    case a mod 10 of
      0,5..9:r:='гривень';
      1:r:='гривня';
      2..4:r:='гривні';
    end
  else r:='гривень';
  write (a,' ',r);
  x:=frac (x);
  b:=trunc (x*100);
  if b div 10 mod 10 <>1 then
    case b mod 10 of
      0,5..9: r:='копійок';
      1: r:='копійка';
      2..4: r:='копійки';
    end
  else r:='копійок';
  write (' ',b,' ',r);
  readkey; end.
```

# Завдання

---

**«1"»: Ввести номер місяця і вивести назву пори року.**

**Приклад:**

Ввести номер місяця:

**4**

весна

**«2"»: Ввести вік людини (від 1 до 150 років) и вивести його разом з наступним слово "рік", "роки" або "років".**

**Приклад:**

Ввести вік:

**24**

Вам 24 роки

Ввести вік:

**57**

Вам 57 років



## Домашнє завдання:

- Вчити конспект
- Виконати №1, 2
- Підготуватися до тестової перевірки
- Підготуватися до контрольної роботи

**Дякую за роботу на уроці!  
До нових зустрічей!!!**