**Лабораторна робота «УМОВНІ ОПЕРАТОРИ В СІ - ШАРП. ТЕРНАРНИЙ ОПЕРАТОР»**

**Теоретичні відомості**

Умовні оператори служать для розгалуження програми. Залежно від деякої умови виконується той чи інший набір команд.

В Сі-шарп є три умовних оператори: **«if-else», «switch» і «?:»** - тернарний оператор.

**Оператор «if-else»**

Даний оператор має наступну структуру:

if ([логічний вираз])

{

   Блок коду, який потрібно виконати при виконанні умови, [логічний вираз] = true (істина)

}

else

{

   Блок коду, який потрібно виконати при незадоволенні умови, [логічний вираз] = false (неправда)

}

Слово  else не є обов'язковим і може бути відсутнім. Тоді отримаємо неповну форму команди розгалуження.

**Приклад** використання оператора «if-else» в програмі, яка перевіряє введене число на парність:

static void Main (string [] args)

{

    int a;

    Console.WriteLine ("Введіть число:");

    a = Convert.ToInt32 (Console.ReadLine ()); // Вводимо дані з клавіатури

    if (a% 2 == 0) // перевіряємо число на парність шляхом знаходження остачі від ділення числа на 2

    {

      Console.WriteLine ("Число" + a + "- парне");

    }

    else

    {

      Console.WriteLine ("Число" + a + "- непарне");

    }

    Console.ReadKey ();

}

 Функція Console.ReadLine () дозволяє ввести дані з клавіатури. Дані вводяться як рядок, а так як потрібно число, ми перетворюємо її в числовий тип. Для перетворення ми використовуємо функцію Convert.ToInt32 ().

**!Якщо після if або else необхідно виконати лише одну команду, фігурні дужки можна опускати:**

if ([логічний вираз])

[команда1] // команда1 виконається лише якщо умовний вираз істинний

[команда2] // команда2 виконається в будь-якому випадку

Оператор if може мати декілька умов:

if ([логічний вираз1])

{Блок1}

else if ([логічний вираз2])

{блок2}

else

{блок3}

**Приклад програми**, яка визначає, яке з двох введених чисел більше:

 static void Main (string [] args)

{

    int a, b;

    Console.WriteLine ("Введіть перше число:");

    a = Convert.ToInt32 (Console.ReadLine ());

    Console.WriteLine ("Введіть друге число:");

    b = Convert.ToInt32 (Console.ReadLine ());

    if (a> b)

      Console.WriteLine ("Перше число більше другого"​​);

    else if (a <b)

      Console.WriteLine ("Друге число більше першого"​​);

    else

      Console.WriteLine ("Числа дорівнюють");

     Console.ReadKey ();

}

Логічне вираз може бути складним. Тоді використовуються логічні оператори «!», «||» і «&&».

**Приклад** програми яка дає пораду, що робити, в залежності від температури на дворі:

static void Main (string [] args)

{

    int t;

    Console.WriteLine ("Введіть температуру на дворі");

    t = Convert.ToInt32 (Console.ReadLine ());

    if (t <-20 || t> 40) // якщо температура менше -20 або більше 40

      Console.WriteLine ("Вам ​​краще посидіти вдома!");

    else

      Console.WriteLine ("Можете йти гуляти");

    Console.ReadKey ();

}

**Оператор switch**

В деяких випадках зручно використовувати умовний оператор «switch» замість «if-else». Він має наступну структуру:

switch (вираз)

{

    case значення1:

      Блок1;

      break;

    case значення2:

      блок2;

      break;

      ...

    case значеніеN:

      блокN;

      break;

    default:

      блокN + 1;

      break;

}

Вираз порівнюється послідовно зі значеннями. Якщо вираз дорівнює значенню - виконується відповідний блок коду і при досягненні ключового слова break оператор switch закінчує роботу. Якщо вираз не відповідатиме ні одного значення, тоді виконається блок після default.

**Приклад програми** з використанням switch, яка виводить на екран назву дня тижня відповідно до порядкового номер дня:

static void Main (string [] args)

{

    int a;

    Console.WriteLine ("Введіть порядковий номер дня тижня:");

    a = Convert.ToInt32 (Console.ReadLine ());

    switch (a)

    {

      case 1:

        Console.WriteLine ("Понеділок");

        break;

      case 2:

        Console.WriteLine ("Вівторок");

        break;

      case 3:

        Console.WriteLine ("Среда");

        break;

      case 4:

        Console.WriteLine ("Четвер");

        break;

      case 5:

        Console.WriteLine ("П'ятниця");

        break;

      case 6:

        Console.WriteLine ("Субота");

        break;

      case 7:

        Console.WriteLine ("Неділя");

        break;

      default:

        Console.WriteLine ("Помилка");

        break;

      }

      Console.ReadKey ();

}

**Тернарний оператор «?:»**

 Цей оператор використовується для скорочення обсягу коду. Ним можна замінювати прості по складності оператори if-else. Тернарний оператор має таку структуру:

 логічний вираз? вираз1: вираз2

 Спочатку обчислюється логічний вираз. Якщо він істинний, то обчислюється вираз1, в іншому випадку - обчислюється вираз2.

**Приклад** використання тернарного оператора «?:» В тій же програмі для перевірки числа на парність:

 static void Main (string [] args)

{

    int a;

    Console.WriteLine ("Введіть число:");

    a = Convert.ToInt32 (Console.ReadLine ());

    Console.WriteLine (a% 2 == 0? "Число парне": "Число непарне");

    Console.ReadKey ();

}

«?:» Також можна використовувати для присвоювання значень.

**Приклад програми**, яка знаходить більше число з двох вводяться:

 static void Main (string [] args)

{

    int a, b, max;

    Console.WriteLine ("Введіть перше число:");

    a = Convert.ToInt32 (Console.ReadLine ());

    Console.WriteLine ("Введіть друге число:");

    b = Convert.ToInt32 (Console.ReadLine ());

    max = a> b? a: b;

}

**Домашнє завдання**

1) З клавіатури вводяться два числа - кількість забитих голів господарями і гостями в футбольному матчі. Вивести на екран результат гри - перемогли господарі / гості / нічия.

2) Напишіть програму, яка перевірятиме число на кратність 3-м і 7-ми (це числа 21, 42, 63 ...). Вивести на екран відповідне повідомлення.