## switch...case Statements

Type the following program into SharpDevelop, and answer the questions

using System ;

class Coffee {

static void Main () {

int add\_up ;

double cost = 35.0;

Console .W riteLine ("You .can .add .into .your.coffee .with .either: ");

Console .W riteLine (".1-Milk ,.2-Sugar,.3-M ilk + Sugar,.4-Nothing ");

Console .W rite ("Your.choice .is: .");

add\_up = int.Parse (Console .ReadLine ());

switch (add\_up ) {

case 1 : cost += 9.00;

break;

case 2 : cost += 5.00;

break;

case 3 : cost += 12.00;

break;

default: cost += 0.0;

break;

}

Console .W riteLine ("Your.coffee cost.is.{0}.baht.",cost);

Console .W riteLine ("Thank.you.");

}

}

• What are the input values of add up to make the value of cost less than 45?

**1 and 2**

• If the user enters 1 to the program, what will be the resulting value of cost?

**44**

• If the user enters 4 to the program, what will be the results displayed?

**Your.coffee cost.is.35.baht.**

**Thank.you.**

• If the user enters -1 to the program, what will be the results displayed?

**Your.coffee cost.is.35.baht.**

**Thank.you.**

Modify the switch statement and the last two display statements so that the program

can handle invalid inputs from the user. If the user enters an invalid input, the program

should set cost to zero and display to the user that “Your coffee is canceled.”

static void Main(string[] args)

{

int add\_up;

double cost = 35.0;

Console.WriteLine("You .can .add .into .your.coffee .with .either: ");

Console.WriteLine(".1-Milk ,.2-Sugar,.3-Milk + Sugar,.4-Nothing ");

Console.Write("Your.choice .is: .");

add\_up = int.Parse(Console.ReadLine());

switch (add\_up)

{

case 1:

cost += 9.00;

break;

case 2:

cost += 5.00;

break;

case 3:

cost += 12.00;

break;

case 4:

cost += 0.0;

break;

default:

Console.WriteLine("Your coffee is canceled.");

cost = -1;

break;

}

if (cost>0)

{

Console.WriteLine("Your.coffee cost.is.{0}.baht.", cost);

Console.WriteLine("Thank.you.");

}

}

## Programming Exercises

### Task 2.1: Which number am I pressing?

Write a C# program to translate a letter to a number according to a given mapping table (from a cell-phone’s dial pad).

|  |  |
| --- | --- |
| **Letters** | **Number** |
| A B C | 2 |
| D E F | 3 |
| G H I | 4 |
| J K L | 5 |
| M N O | 6 |
| P Q R S | 7 |
| T U V | 8 |
| W X Y Z | 9 |

static void Main(string[] args)

{

Console.Write("Please input a letter: ");

var input = Console.ReadKey();

int num = 0;

switch (input.KeyChar)

{

case 'A': case 'a':

case 'B': case 'b':

case 'C': case 'c':

num = 2; break;

case 'D': case 'd':

case 'E': case 'e':

case 'F': case 'f':

num = 3; break;

case 'G': case 'g':

case 'H': case 'h':

case 'I': case 'i':

num = 4; break;

case 'J': case 'j':

case 'K': case 'k':

case 'L': case 'l':

num = 5; break;

case 'M': case 'm':

case 'N': case 'n':

case 'O': case 'o':

num = 6; break;

case 'P': case 'p':

case 'Q': case 'q':

case 'R': case 'r':

case 'S': case 's':

num = 7; break;

case 'T': case 't':

case 'U': case 'u':

case 'V': case 'v':

num = 8; break;

case 'W': case 'w':

case 'X': case 'x':

case 'Y': case 'y':

case 'Z': case 'z':

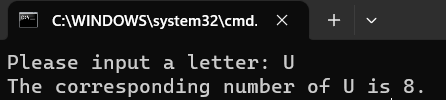
num = 9; break;

default: num = -1; break;

}

Console.WriteLine($"\r\nThe corresponding number of {input.KeyChar} is {num}.");

}



## Book shipping

Write a C# program to compute the cost of book shipping. The shipping cost is computed according to the shipping type and the package weight. The shipping rate is show in the following table

|  |  |  |
| --- | --- | --- |
| **Shipping types** | **Weight** | **Rate (bahts/gram)** |
| Regular | First 2000 | 0.25 |
| Regular | After 2000 | 0.35 |
| Express | Use the same rate as regular + 50 bahts fee | |

static void Main(string[] args)

{

Console.Write("Enter a shipping type (R-Regular, X-Express): ");

var type = Console.ReadKey().KeyChar;

var cost = 0.0;

if (type == 'R' || type == 'X')

{

Console.Write("\r\nEnter the weight of your book package (kilograms): ");

var weight = Convert.ToDouble(Console.ReadLine()) \* 1000;

if (weight <= 2000)

{

cost = weight \* 0.25;

}

else

{

const double fixedRate = 2000 \* 0.25;

cost = fixedRate + (weight - 2000) \* 0.35;

}

if (type == 'X')

{

cost += 50;

}

Console.WriteLine($"\r\nYour shipping cost is {cost} bahts with {type}-type shipping of {weight/1000}-kilogram books");

}

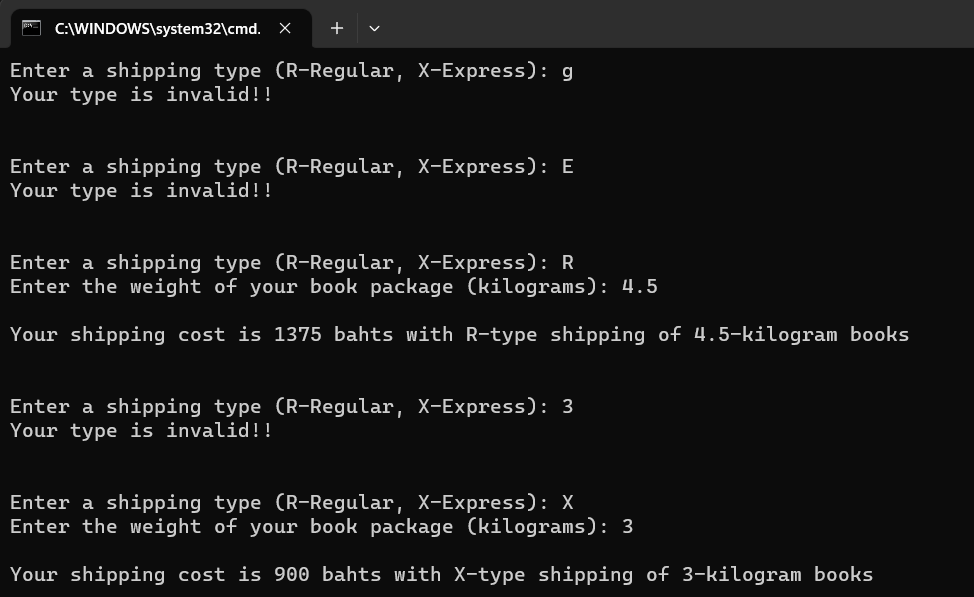
else

{

Console.WriteLine("\r\nYour type is invalid!!");

}

}



## while Loops

**Exercise 1.1: Modify the program in Example 1.1 so that it will display the numbers**

**from** *N* **to 1 instead. Determine what should be put in the blanks marked (a)–**

**(c).**

using System ;

class While2 {

static void Main () {

int N;

int i;

Console . Write(" Please. input.N:.");

N = int . Parse( Console . ReadLine ());

\_\_\_ (a) \_\_\_;

while ( \_\_\_ (b) \_\_\_) {

Console . WriteLine(i);

\_\_\_\_\_\_\_ ( c) \_\_\_\_\_\_ ;

}

}

}

|  |  |
| --- | --- |
|  | **Expression/Statement** |
| (a) | **i=N** |
| (b) | **i>0** |
| (c) | **i--** |