CSX3001/ITX3001
CS1201 COMPUTER PROGRAMMING 1

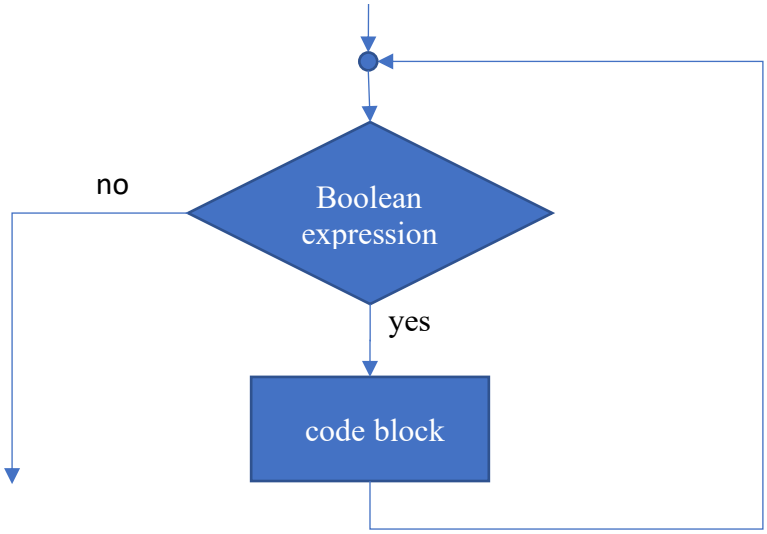
CLASS 04 REPETITION CONTROL STRUCTURE

LOOP CONTROL STRUCTURE, WHILE LOOP, DEFINITE VS. INDEFINITE LOOP,
USING WHILE AND WHILE...ELSE STATEMENTS

PYTHON

REPETITION CONTROL STRUCTURE

Repetition (aka. iteration or looping) control structure is a kind of control that will break the default control structure (sequential control structure) by repeating execution the same block of code (a statement or many statements) several times.



WHILE STATEMENT

```
while Boolean expression :
    code block
```

A ***while*** statement starts with the keyword ***while*** followed by a *condition*, which is always a **Boolean expression**. The computer checks the condition and executes the code block inside the body of ***while*** statement if the condition is *true* and loop back to check the condition again or exits from the loop if the condition is *false*. In another words, a while statement will keep executing the code block if the condition is evaluated to be *true* otherwise stop and continue with the other statement(s).

What happen if the condition is always evaluated to be true?

Let's try with the follow fragment of code.

```
count = 0

while count<10 :
    print("Keep printing")

```

What you have just seen when you run the above code is called infinite loop.

How to avoid the infinite loop?

……

……

How to modify the previous code and print out the string 10 times?

DEFINITE VS. INDEFINTE

In programming, there are two types of iteration which are definite and indefinite loop.

DEFINITE LOOP

Definite loop is a loop that the number of rounds the statement(s) in loop body executed is defined or specified. The number of times the loop body will repeat is known in advance. The previous program, printing the string 10 times is an example of definite loop.

Another Example:

The following fragment of code will print out the first five positive even number.

```
evenNumber = 0
nTime = 5

while nTime > 0 :
    print(evenNumber)
    evenNumber += 2
    nTime -= 1
```

INDEFINITE LOOP

Indefinite loop is a loop that the number of rounds the loop body will be executed is not explicitly specify. The number of times the loop body will repeat is unknown.
An example of indefinite loop

```
tester = int(input("Enter a positive integer (to terminate
enter a negative integer) : "))

while tester >= 0 :
   print("You enter number " + tester)
   tester = int(input("Enter a positive integer (to
   terminate enter a negative integer) : "))

print("Exit from a loop")
```

The above example will repeat printing the entered number if the input is not a negative integer.

WHILE...ELSE STATEMENT

```
while Boolean expression :
   code block
else :
   code block
```

In Python, there is an option for ***while*** statement to put ***else*** after the body of ***while*** statement. The code block after this ***else*** will be executed whenever the condition of the ***while*** loop is evaluated to be ***false***.
An Example of while...else statement

```
tester = int(input("Enter a positive integer(to terminate  enter
a negative integer) : "))

while tester >= 0 :
   print("You enter number " + tester)
   tester = int(input("Enter a positive integer(to\
            terminate  enter a negative integer) : "))
else :
   print("Exit from a loop")

```

Note: You may notice that there is no difference between previous two examples, therefore when should we use while…else? The answer of this question will be answered in the later class.

PSEUDOCODE

Pseudocode is an informal high-level description of the operating principle of a computer program or other algorithm. It can be used to explain the sequence of a program as same as flowchart.

An example of using pseudocode,
Let's start from a programming problem:

> Write a program to read in a positive integer number and converse and print out the representation of this number in base 2 (binary number)

This programming problem can be solved by the sequence which is represented by the following pseudocode:

```
Get a positive integer as an input
Declare an empty string for an output
During input is greater than zero keep doing the following:
    - find the remainder of the input divided by 2
    - convert the remainder to string and concatenate with
      output string
    - update input by dividing it with 2 and take only the
      integer part
print out the output
```

which can be directly convert into Python program as follow:

```
1  myInput = int(input("Enter a positive integer: "))
2  binOutput = ""
3  while myInput > 0:
4      binDigit = myInput % 2
5      binOutput = str(binDigit) + binOutput
6      myInput = int(myInput / 2)
7  print(binOutput)
```

Can you draw a flowchart to represent the same sequence as this pseudocode?

LOOP EXERCISES

1. Write a program that keeps asking the user to type positive integers until negative number entered and report that how many numbers entered and the sum of these positive integers.

   Sample Run#1

   Enter a positive integer (to terminate enter a negative number): 3
   Enter a positive integer (to terminate enter a negative number): 5
   Enter a positive integer (to terminate enter a negative number): 10
   Enter a positive integer (to terminate enter a negative number): -4
   The total of positive number entered is 3
   The sum of all positive integer(s) is 18

   Sample Run#2

   Enter a positive integer (to terminate enter a negative number): -1
   The sum of all positive integer(s) is 0

---

2. Write a program that asks the user to enter number of numbers, n, then let the user enter these n numbers and then find out the average of these n numbers.

   Sample Run#1

   How many numbers? 4
   Number#1: 12.5
   Number#2: -12
   Number#3: 13.5
   Number#4: -13
   The average of these number is 0.25

   Sample Run#1

   How many numbers? 2
   Number#1: 11
   Number#2: 19
   The average of these number is 15.0

3. Write a program to find out the summation of all integer start from 1 up to N. Your input/output format must strictly follow the example in the sample run.

Sample Run#1
Enter n: 4
1+2+3+4 = 10

Sample Run#2
Enter n: 0
The entered value must be greater than ZERO!

Sample Run#3
Enter n: 7
1+2+3+4+5+6+7 = 28

---

4. Write a program to compute n factorial.

Sample Run#1
Enter n: 0
0! = 1

Sample Run#2
Enter n: 1
1! = 1

Sample Run#3
Enter n: 4
4! = 1x2x3x4 = 24

5. Write a program to get two integer numbers, firstInt and secondInt and print out all integer numbers start from firstInt to secondInt over the same line. *Note: If the first integer less than the second integer, list all numbers in ascending(increasing) order, otherwise in descending (decreasing) order.*

Sample Run#1

Enter the first integer: -3
Enter the second integer: 5
Output: -3 -2 -1 0 1 2 3 4 5

Sample Run#2

Enter the first integer: 3
Enter the second integer: -2
Output: 3 2 1 0 -1 -2

---

6. Write a program to get 3 integer numbers, numX, numY, and numZ, where numX ≥ numY ≥ numZ, and count how many number start from numX to numZ that can be divisible by numZ.

Sample Run#1

Enter 3 integer numbers, where as numX ≥ numY ≥ numZ.
numX: 2
numY: 3
numZ: 12
There are 4 numbers in 2...12 that can be divisible by 3.

Sample Run#2

Enter 3 integer numbers, where as numX ≥ numY ≥ numZ.
numX: 10
numY: 5
numZ: 20
There are 3 numbers in 10...20 that can be divisible by 5.

7. Write a program to read in a positive integer number and print out the sum of each digit in the input number.

Sample Run#1
Input number: 1243
The sum of each digit in 1243 is 10.

Sample Run#2
Input number: 59124
The sum of each digit in 59124 is 21.

---

8. Write a program to print out n stars (“*”) on a line.

Sample Run#1
n: 2
**

Sample Run#2
n: 5
*****

---

9. Write a program to print out n stars (“*”) on n lines.

Sample Run#1
n: 2
**
**

Sample Run#2
n: 3
***
***
***

10. Write a program to print out the left-aligned triangle of star(s) depending on the input, n, by follow the pattern in the following samples.

Sample Run#1

```
n: 1
*
```

Sample Run#2

```
n: 2
*
**
```

Sample Run#3

```
n: 5
*
**
***
****
*****
```

---

11. Write a program to print out the right-aligned triangle of star(s) depending on the input, n, by follow the pattern in the following samples.

Sample Run#1

```
n: 1
*
```

Sample Run#3

```
n: 5
    *
   **
  ***
 ****
*****
```

12. Write a program to print out the triangle of star(s) depending on the input, n, by follow the pattern in the following samples.

Sample Run#2

```
n: 2
 *
***
```

Sample Run#3

```
n: 5
    *
   ***
  *****
 *******
*********
```

---

13. Write a program to get an odd positive integer and print out the diamond shape of star(s) depending on the input, n, by follow the pattern in the following samples.

Sample Run#2

```
n: 3
 *
***
 *
```

Sample Run#3

```
n: 5
  *
 ***
*****
 ***
  *
```

💎 ASSIGNMENTS

Complete the following exercises in Python IDLE. You must name the python file as, {your-id}_class0{number}_{course-code}_{section-number}_assignment{number}.py for example, for assignment 1 will be named,

6120001_class04_csx3001_541_assignment1.py

1. Write a program to ask the user to enter number of real numbers, n, then read in all n real numbers and find out the first highest and the second highest number.

   Sample Run#1

   Enter number of real numbers: 5
   Number#1: 45.23
   Number#2: 23.5
   Number#3: 12
   Number#4: 12.321
   Number#5: 105.5

   The first highest number is 105.5
   The second highest number is 45.23

   Sample Run#2

   Enter number of real numbers: 1
   Number#1: 100.275

   The first highest number is 105.5
   There is no second highest number

2. Write a program to read in an integer number and print out another integer number in reverse order of the input.

   Sample Run#1
   Input number: 1214
   Output number: 4121

   Sample Run#2
   Input number: 122259
   Output number: 952221

   Note: String feature are not allowed in solving this problem.

---

3. Write a Python code that computes the total balance in the account after depositing for n years with the first deposit of x Baht, and the interest rate is fixed at y% in the first year and is increased 1% every year. The interest is calculated at each completion of one year and the interest earned is deposited into the account. The output is shown as below as example.

```
Enter intial account balance, interest rate and years: 10000,10,10
Year   Int Rate  Int Earned    Balance
  1    10.00     1000.00       11000.00
  2    11.00     1210.00       12210.00
  3    12.00     1465.20       13675.20
  4    13.00     1777.78       15452.98
  5    14.00     2163.42       17616.39
  6    15.00     2642.46       20258.85
  7    16.00     3241.42       23500.27
  8    17.00     3995.05       27495.31
  9    18.00     4949.16       32444.47
 10    19.00     6164.45       38608.92
```