EEE102 C++ Programming and Software Engineering II

Lab Practice 2

Fundamental of C++ language

Notice:

- The aim of this lab is for you to become familiar with the basics of C++ as a programming language.
- Practice with the exercises. These parts are not for submission.
- The work for submission is in the separated file "Assessment 1".

1. Variable and constant

When declaring a variable, take a moment to consider by what name it will be known. A valid name is made up of letters (uppercase and lowercase), digits and underscore character (_). The only limitation is that a variable name can't begin with a digit. Of course, the name must be unique. Neither you nor the compiler would be able to distinguish between variables with the same name. For some simple cases, using single letters like `i, j, k` is enough, but as programs grow larger this becomes less informative. Like the name for a product or service, a variable name should do two things:

1. Be unique
2. Express the purpose

Exercise 1.1				
Judge the following variable names are legal or not.				
`3d_movie`	**`Int`**	**`_PI`**	**`default`**	**`My name`**

It is encouraged to use meaningful names for variables, such as **`dateOfBirth`**, **`my_age`**, etc.

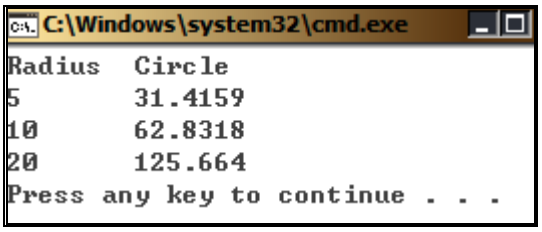
Exercise 1.2				
Guess the possible datatype and content stored in the following variables. Eg.: **`iNumOfStudent`** – int type, storing student number.				
`iMax`	**`fSum`**	**`chTemp`**	**`bIsEmpty`**	**`arrMark`**

With the **`const`** prefix you can declare constants with a specific type in the same way as you would do with a variable as shown below. The name of constant is usually all in capital to be distinguished from normal variable names.

```
const double PI = 3.14159;
const char TAB = '\t';
```

Exercise 1.3

Using the two constants defined above to write a program, calculating the circumference of a circle with different radius of 5, 10 and 20. Print the result on screen as shown below.



2. Datatypes

Data are used to express particular values within the source code of a program. Similar to variables, literal data are considered to have a specific datatype. By default, integer literals are of type **`int`**. However, we can force them to either be **`unsigned`** by appending the **`u`** character to it, or **`long`** by appending **`l`**, such as **`75u`** (**`unsigned int`**), **`75l`** (**`long`**), and **`75ul`** (**`unsigned long`**).

Exercise 2.1

Determine the datatype of the following data.

Eg.: **`3 – int`**

`3.0f`	**`'$'`**	**`6.02e23`**	**`1.6E-19`**	**`"ABCDE"`**

When programming, we store the variables in our computer's memory, but the computer has to know what kind of data we want to store in them, since it is not going to occupy the same amount of memory to store a simple number than to store a single letter or a large number, and they are not going to be interpreted the same way. The **`sizeof`** operator returns the size of a specific datatype or a variable. For example, **`sizeof(int)`** returns the size of **`int`** type, which is 4 for Windows XP, Vista and Win7 systems. The **`sizeof`** operator can be applied to a type name or to a variable name, or directly to a data.

Exercise 2.2

Write a programme which can tell you (using **`cout<<`** to display information on the screen) the length in bytes of all the datatypes in lecture note slide 7.

Exercise 2.3

In the following example, what are the results for the three statements in Lines 3-5? Explain the reason for it.

Line	Code
1	`double da = 5.5;`
2	`double db = 5.5;`
3	`int result1 = (int)da+(int)db;`
4	`int result2 = (int)da+db;`
5	`int result3 = (int)(da+db);`

3. Operators and expressions

Operators in C++ are mostly made of signs that are available in all keyboards. Recalling what you've learned and try the following exercises.

Exercise 3.1
In the following statements, what are the datatype and value of **z1-z5**?
Explain the reasons.

```
1    double z1=7/-2;
2    double z2=7.0/-2;
3    int z3 = 222 / 300;
4    double z4 = 222.0 / 300.0;
5    int z5 = 222 / 300.00;
```

Exercise 3.2
Translate the following algebraic expressions into C++.

Eg: $y = x + \frac{3}{4} - 2$	`y = x + 3/4 - 2;`
$y = x^2 + 2x + 1$	
$y = \frac{x}{1 - x^2}$	
$y = x^4 + e^x + 1$	

Exercise 3.3
Assuming **m** and **n** are two integer variables and **m=5**, **n=11**.
Work out the results of the following expressions:

```
1    m - n
2    n % m
3    n == m
4    n >= m
5    m == 5 && n == 11
```

Exercise 3.4
Assuming **a** and **b** are two integer variables and **a=5**, **b=8**.
Calculate the value of **a**, **b** and **c** after executing the statements.
(*** These statements are serially performed. The initial state of the second expression is no longer **a=5** and **b=8** ***).

```
1    c = a+++a++;
2    c = --b+b-10;
3    c = 50-++a+b++
```