

บทที่ 2

เรื่อง หลักการพื้นฐานของภาษา C++

วัตถุประสงค์

- เพื่อให้นักศึกษาเข้าใจถึงองค์ประกอบพื้นฐานต่างๆ ของภาษา C++
- เพื่อให้ นักศึกษารู้จักและเข้าใจคำสั่งการแสดงผลและนำเข้าข้อมูล

พื้นฐานภาษา C++

➤ คำอธิบาย(Comment)

การใส่คำอธิบาย คือการใส่ข้อความเพิ่มเติม โดยไม่มีผลต่อโปรแกรม เพื่อให้ผู้ดูแลโปรแกรม หรือผู้พัฒนาอื่นสามารถเข้าใจได้ง่าย สำหรับภาษา C++ สามารถเขียนคำอธิบายได้ 2 แบบ คือ แบบแรกใช้เครื่องหมายทับ 2 อันติดกัน(double slash) หรือ // โดยวางหน้าคำอธิบาย และใช้ได้เพียงบรรทัดเดียว แบบที่สอง ในกรณีต้องการใส่คำอธิบายหลายๆบรรทัด ใช้เครื่องหมาย /* ใส่หน้าบรรทัดแรก และเครื่องหมาย */ ใส่หลังบรรทัดสุดท้ายที่ต้องการ

ตัวอย่าง

```
/* Program 2_1 : Display text
   Author : Somchai Cheingpongpan
   Date : 16/05/2007 */
#include <iostream> // preprocessor directive
using namespace std; // using directive

int main()
{
    // display statment
    cout << "This is structure of C++ language" << endl;
    return (0); // end program
}
```

➤ คำสั่ง Preprocessor

คำสั่งประเภทนี้ส่วนใหญ่จะใช้เครื่องหมาย # วางหน้า Preprocessor เป็นส่วนที่ตัวคอมไพล์จะต้องทำก่อน จึงจะคอมไพล์ตัวโปรแกรม สำหรับ Preprocessor ที่ใช้งานมากที่สุดจะเป็น include ใช้สำหรับอ้างไฟล์ไลบรารี ซึ่งมี 2 แบบ คือ

- ถ้าใช้ #include <> อยู่ภายในเครื่องหมาย <> จะใช้กับไฟล์ไลบรารีมาตรฐาน เช่น
#include <iostream>
- ถ้าใช้ #include " " อยู่ภายในเครื่องหมาย " " จะใช้กับไฟล์ไลบรารีที่สร้างเอง เช่น
#include "my_header.h"

ส่วนอีกคำสั่งที่นิยมใช้ #define สำหรับกำหนดตัวแปรที่เป็นค่าคงที่ไว้ใช้งานในโปรแกรม เช่น

```
#define PI 3.1415
```

➤ คำสั่ง Using

คำสั่งนี้จะใช้กับไฟล์ไลบรารีต่างๆ ที่สร้างภายในเนมสเปซเป็นหลัก เพื่อใช้สำหรับการเรียกใช้ฟังก์ชันหรืออื่นๆในไฟล์ไลบรารีนั้น สามารถเรียกได้โดยตรง ซึ่ง namespace ที่กำหนดมาให้เป็นหลักชื่อ std

ตัวอย่าง

```
#include <iostream>          // preprocessor directive
using std::cout;             // using directive

int main()
{
    cout << "This is structure of C++ language" << std::endl;
    return (0);
}
```

```
#include <iostream> // preprocessor directive
using namespace std; // using directive

int main()
{
    cout << "This is structure of C++ language" << endl;
    return (0);
}
```

➤ คำสั่ง (Statement)

คำสั่งคือ โค้ดที่ใช้ให้โปรแกรมทำงาน โดยแต่ละคำสั่งจะต้องปิดด้วยเครื่องหมาย ; ซึ่งในโปรแกรมหนึ่งจะมีหลายคำสั่ง เพื่อให้โปรแกรมทำงานได้ตามต้องการ ลักษณะคำสั่งประกอบด้วย

- คำสั่งเดียว (Simple Statement) คือคำสั่งที่ปิดท้ายด้วย ; ซึ่งจะมีชื่อเรียกต่างๆกันออกไปดังนี้

- คำสั่งการประกาศตัวแปร ใช้เพื่อประกาศตัวแปร
- คำสั่งการกำหนดค่า ใช้เพื่อกำหนดค่าให้กับค่า
- คำสั่งนิพจน์ ใช้เพื่อกำนวณนิพจน์ทางคณิตศาสตร์หรืออื่นๆ

- คำสั่งผสม(Compound Statement) เป็นการรวมกันของคำสั่งเดียว หลายๆคำสั่ง ประกอบกัน หรือคำสั่งการควบคุม คำสั่งผสมจะอยู่ภายในบล็อกของเครื่องหมายปีกกาเปิด { และเครื่องหมายปีกกาปิด }

ตัวอย่าง

```
{
    x = 10;
    y = x + 20;
    if (y < 30) x = 20;
}
```

- คำสั่งควบคุม(Control Statement)

➤ The C++ Character Set

เป็นลักษณะของกลุ่มตัวอักษรที่ใช้ในภาษา C++ ซึ่งประกอบด้วยตัวอักษร A-Z ทั้งตัวใหญ่และตัวเล็ก ตัวเลข 0-9 และตัวอักษรพิเศษ เพื่อมาใช้ในโปรแกรม (เช่น ค่าคงที่ ตัวแปร ตัวดำเนินการ นิพจน์ เป็นต้น) โดยมีอักขระพิเศษ ดังต่อไปนี้

| | | | | | |
|---|---|---|---|---|---------|
| ! | * | + | \ | “ | < |
| # | (| = | | { | > |
| % |) | ~ | ; | } | / |
| ^ | - | [| : | , | ? |
| & | _ |] | ' | . | (blank) |

และยังมีตัวอักขระแบบพิเศษอีกแบบ เรียกว่า Escape sequence หรือ Escape character ซึ่งจะต้องนำหน้าด้วยเครื่องหมาย \ (backslash)

ตารางแสดงความหมายของอักขระ Escape sequence ต่างๆ

| อักขระ | ความหมาย | ชื่อรหัส ASCII |
|--------|------------------------------------|----------------|
| \n | เลื่อน cursor ขึ้นบรรทัดใหม่ | NL(LF) |
| \t | เลื่อน cursor ไปแนวนอน 1 แท็บ | HT |
| \r | เลื่อน cursor ไปที่ต้นบรรทัด | CR |
| \b | เลื่อน cursor ไปทางซ้าย 1 ตัวอักษร | BS |
| \\ | พิมพ์ \ ออกทางหน้าจอ | \ |
| \? | พิมพ์ ? ออกทางหน้าจอ | ? |
| \' | พิมพ์ ' ออกทางหน้าจอ | ' |
| \" | พิมพ์ " ออกทางหน้าจอ | " |
| \000 | ใช้เลขฐาน 8 ของรหัส ASCII | 000 |
| \xhhh | ใช้เลขฐาน 16 ของรหัส ASCII | hhh |

การทดลองที่ 2_1 โปรแกรมแสดงข้อความด้วยการใช้งานอักขระพิเศษ

| | |
|----|--|
| 1 | /* Program 2_1 : Display with escape sequence */ |
| 2 | #include <iostream> |
| 3 | using namespace std; |
| 4 | |
| 5 | int main() |
| 6 | { |
| 7 | cout << "12345678901234678901234567890\n"; |
| 8 | cout << "ID : \t50-6626-120-1\n"; |
| 9 | cout << "Name : \tSomchai Cheingpongpan\n"; |
| 10 | cout << "\" Information Technology \""n"; |
| 11 | return (0); |
| 12 | } |

บันทึกผลการทดลอง

การทดลองที่ 2_2 โปรแกรมแสดงข้อความด้วยการใช้งานอักขระพิเศษ

```

1  /* Program 2_2 : Display with escape sequence */
2  #include <iostream>
3  using namespace std;
4
5  int main()
6  {
7      cout << "'A\' is Character.\n";
8      cout << "\"Microsoft Visual Studio 2010 Professional\" is String.\n";
9      cout << "A = \101 = \x41 \n";
10     return (0);
11 }

```

บันทึกผลการทดลอง

➤ **Identifiers and Keywords**

เป็นการตั้งชื่อเพื่อนำไปใช้งานในโปรแกรมอาจเป็นชื่อตัวแปร ฟังก์ชัน และอื่นๆ โดยมีหลักการดังนี้

- จะต้องขึ้นต้นด้วยตัวอักษร a ถึง z หรือ A ถึง Z หรือขึ้นต้นด้วยขีดล่าง(under score) เท่านั้น
- ตัวเลขสามารถใช้ตั้งชื่อได้ แต่ไม่สามารถนำมาขึ้นต้นชื่อได้
- ห้ามมีการเว้นวรรค ห้ามตั้งชื่อตัวแปรซ้ำกัน
- ชื่อตัวแปรไม่ควรยาวเกินไป และสามารถสื่อความหมายได้ดี
- ไม่ใช่ชื่อตัวแปรที่ซ้ำกับคำสงวน หรือคีย์เวิร์ด

ตาราง Keyword ในภาษา C++

| | | | | |
|--------------|-----------|----------|------------------|-------------|
| asm | auto | bool | break | case |
| catch | char | class | const | const_cast |
| continue | default | delete | do | double |
| dynamic_cast | else | enum | explicit | export |
| extern | false | float | for | friend |
| goto | if | inline | int | long |
| mutable | namespace | new | operator | private |
| protected | public | register | reinterpret_cast | return |
| short | signed | sizeof | static | static_cast |
| struct | switch | template | this | throw |
| true | try | typedef | typeid | typename |
| union | unsigned | using | virtual | void |
| volatile | wchar_t | while | | |

ตัวอย่าง

```
x      y12    sum_1      _temp      Names      tax_rate
TABLE 4th    "x"      order-no      error flag
```

➤ ชนิดข้อมูล (Data Types)

สำหรับชนิดข้อมูลพื้นฐานต่างๆ ซึ่งส่วนใหญ่จะถูกกำหนดไว้แล้วในภาษา C++ (Built-in Type) หรืออาจเรียกว่า ชนิดข้อมูลแบบดั้งเดิม(Primitive Type) ซึ่งมีรายละเอียดของชนิดข้อมูลดังนี้

| ประเภทข้อมูล | คำอธิบาย | ช่วงข้อมูล | ขนาด(byte) |
|--------------|----------------------|----------------------------------|------------|
| bool | เก็บค่าทางตรรกศาสตร์ | true/false หรือ 1/0 | 1 |
| char | ตัวอักษร 1 ตัว | ค่าอักขระ 1 ตัว | 1 |
| short | ตัวเลขจำนวนเต็ม | -32,768 ถึง 32767 | 2 |
| int | ตัวเลขจำนวนเต็ม | -2,147,483,648 ถึง 2,147,483,647 | 4 |
| long | ตัวเลขจำนวนเต็ม | -2,147,483,648 ถึง 2,147,483,647 | 4 |
| float | ตัวเลขทศนิยม | -3.4e38 ถึง 3.4e38 | 4 |
| double | ตัวเลขทศนิยม | -1.7e308 ถึง 1.7e308 | 8 |
| long double | ตัวเลขทศนิยม | -1.7e308 ถึง 1.7e308 | 8 |

ถ้าหากต้องการเก็บข้อมูลอักขระเป็นข้อความจำนวนหลายๆ ตัว จะสามารถเก็บได้ 2 วิธี คือ

- ใช้ชนิดข้อมูลแบบ char แต่ต้องประกาศตัวแปรในลักษณะอาเรย์ ซึ่งจะได้อธิบายต่อไปในบทหลังๆ
ตัวอย่าง char Product_Name[20];
- ใช้คลาส string ที่ภาษา C++ ได้สร้างขึ้นมาเพิ่ม โดยทำงานในลักษณะวัตถุ(Object)
ตัวอย่าง string Product_Name;


การทดลองที่ 2_3 โปรแกรมแสดงข้อมูลการใช้หน่วยความจำของชนิดข้อมูลต่างๆ โดยใช้คำสั่ง sizeof

```
1  /* Program 2_3 : Check size of data type */
2  #include <iostream>
3  using namespace std;
4  int main()
5  {
6      cout << "bool \t= " << sizeof(bool) << " bytes." << endl;
7      cout << "char \t= " << sizeof(char) << " bytes." << endl;
8      cout << "short \t= " << sizeof(short) << " bytes." << endl;
9      cout << "int \t= " << sizeof(int) << " bytes." << endl;
10     cout << "long \t= " << sizeof(long) << " bytes." << endl;
11     cout << "float\t= " << sizeof(float) << " bytes." << endl;
12     cout << "double\t= " << sizeof(double) << " bytes." << endl;
13     cout << "long double\t= " << sizeof(long double) << " bytes." << endl;
14     return (0);
15 }
```

บันทึกผลการทดลอง[illegible]

สำหรับชนิดข้อมูลที่เกี่ยวกับค่าของตัวเลขในตารางนั้น จะเป็นการเก็บทั้งค่าบวกและค่าลบ ที่เรียกว่า signed คือแบบคิดเครื่องหมาย แต่ในบางกรณีของข้อมูลค่าตัวเลขอาจไม่มีค่าลบเลย ฉะนั้นหากต้องการให้เป็นชนิดข้อมูลตัวเลขที่เก็บมีแต่ค่าบวกก็สามารถทำได้ โดยใช้เครื่องหมาย unsigned เต็มไว้ข้างหน้าชนิดข้อมูลนั้นๆ จะทำให้สามารถเก็บข้อมูลได้มากขึ้น เช่น

| | |
|----------------|-----------------------|
| unsigned short | → 0 ถึง 65,535 |
| unsigned int | → 0 ถึง 4,294,967,295 |
| unsigned long | → 0 ถึง 4,294,967,295 |

 การเติม unsigned ไว้ข้างหน้าชนิดข้อมูลนั้นๆ จะทำให้ขนาดของข้อมูลไม่มีการเปลี่ยนแปลง

```
1  /* Program 2_4 : Check scope value of data type */
2  #include <iostream>
3  using namespace std;
4  int main()
5  {
6      cout << "Minimum value of char : " << CHAR_MIN << endl;
7      cout << "Maximum value of char : " << CHAR_MAX << endl;
8      cout << "Minimum value of short : " << SHRT_MIN << endl;
9      cout << "Maximum value of short : " << SHRT_MAX << endl;
10     cout << "Minimum value of int : " << INT_MIN << endl;
11     cout << "Maximum value of int : " << INT_MAX << endl;
12     cout << "Minimum value of long : " << LONG_MIN << endl;
13     cout << "Maximum value of long : " << LONG_MAX << endl;
14     cout << "Maximum value of unsigned char : " << UCHAR_MAX << endl;
15     cout << "Maximum value of unsigned short : " << USHRT_MAX << endl;
16     cout << "Maximum value of unsigned int : " << UINT_MAX << endl;
17     cout << "Maximum value of unsigned long : " << ULONG_MAX << endl;
18     return (0);
19 }
```

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

เป็นการกำหนดค่าคงที่ให้กับตัวแปร มีค่าคงที่ด้วยกัน 4 ประเภทดังนี้

- ## ตัวอย่าง

ฐานสิบ

23 45 101 55

23L 45l 101L 55L

ฐานแปด

023 077L 045 010 (19 63 37 8)

ฐานสิบหก

0x2a 0x45 0XffL 0xA1e (42 69 255 2590)

- ตัวอย่าง

2.34 3.1416 45.3F 0.21L

1.23E10 0.23E-4 45.e+23L

- ### ตัวอย่าง

'A' '4' '=' '\n' '\010' '\xff'

- ตัวอย่าง

“Hello World”

```
1  /* Program 2_5 : Output different base constants */
2  #include <iostream>
3  using namespace std;
4  int main()
5  {
6      cout << "Display integer constants\n" << endl;
7      cout << "Octal constant 023 is " << 023 << " decimal\n";
8      cout << "Decimal constant 23 is " << 23 << " decimal\n";
9      cout << "Hexa constant 0x23 is " << 0x23 << " decimal\n";
10     return (0);
11 }
```


บันทึกผลการทดลอง

การทดลองที่ 2_6 โปรแกรมแสดงข้อมูลของค่าคงที่ที่ทศนิยมในรูปแบบต่างๆ

```
1  /* Program 2_6 : Illustrate different forms of floating-point constants
2     that have the same value */
3  #include <iostream>
4  using namespace std;
5  int main()
6  {
7      cout << 230.E+3 << endl;
8      cout << 230E3 << endl;
9      cout << 230000.0 << endl;
10     cout << 2.3e5 << endl;
11     cout << 0.23E6 << endl;
12     cout << .23e+6 << endl;
13     return (0);
14 }
```

บันทึกผลการทดลอง

➤ ตัวแปร (Variables)

ตัวแปรทำหน้าที่เก็บข้อมูลในหน่วยความจำของเครื่องคอมพิวเตอร์ เพื่อให้สามารถบันทึก ดึงข้อมูลมาและจัดการข้อมูลได้ เมื่อต้องการสร้างตัวแปรหนึ่งตัวขึ้นมา จะต้องรู้ขนาดข้อมูลที่ต้องการใช้เก็บ และกลุ่มของข้อมูล จึงจะทำให้เครื่องคอมพิวเตอร์จัดสรรหน่วยความจำได้อย่างถูกต้อง เช่น

```
int x; // ตัวแปร x เก็บชนิดข้อมูลจำนวนเต็ม(integer)
```

ตัวแปรสามารถทำงานได้ 2 หน้าที่ คือ

- หน้าที่ในการให้ค่าของข้อมูลที่เก็บอยู่ในหน่วยความจำหนึ่ง เรียกการทำงานนี้ว่า rvalue ซึ่งเป็นได้ทั้งค่าคงที่ ตัวแปร หรือนิพจน์ และจะต้องอยู่ทางด้านขวาของเครื่องหมาย = เสมอ
- หน้าที่ในการรับค่ามาเก็บตามตำแหน่งที่อยู่(address)ในหน่วยความจำ ซึ่งจะเก็บข้อมูลและสามารถอ้างอิงได้ เรียกการทำงานนี้ว่า lvalue ซึ่งจะเป็นได้เฉพาะตัวแปร

เช่น

```
int x = 10;    // 10 เป็นค่าคงที่ และเป็น rvalue
int y = x;     // x เป็นตัวแปร และเป็น rvalue
char ch = 'a'; // 'a' เป็น rvalue
x = x + 10     // x เป็นตัวแปร และเป็นทั้ง rvalue และ lvalue
```


➤ การประกาศ (Declarations)


เป็นการประกาศตัวแปรต่างๆ ที่ใช้งานในโปรแกรม สำหรับภาษา C++ การจะใช้ตัวแปรใด จะต้องมีการประกาศตัวแปรนั้นมาก่อนเสมอ มิฉะนั้นจะเกิดเป็นความผิดพลาด(Syntax Error) โดยมีรูปแบบการกำหนด คือ

```
ชนิดข้อมูล      ชื่อตัวแปร;
ชนิดข้อมูล      ชื่อตัวแปร=ค่าเริ่มต้น;
ชนิดข้อมูล      ชื่อตัวแปร1,ชื่อตัวแปร2, ...;
```

ตัวอย่าง

```
int    math, eng;
char   grade;
int    total = 0;
float  average = 0.0 ;
double salary;
int    WordCount, Radius, Height;
float  FlightTime, Speed;
```

 การประกาศตัวแปรจะกำหนดค่าเริ่มต้นให้กับตัวแปรเลยก็ได้ หรืออาจจะไปกำหนดในคำสั่งถัดไปได้ แต่ตัวแปรนั้นจะต้องมีค่าเริ่มต้นก่อนนำไปใช้งานเสมอ

 สามารถประกาศตัวแปรกับชนิดข้อมูลแบบ Primitive ในลักษณะเชิงวัตถุได้
เช่น int x(5);

การทดลองที่ 2_7 โปรแกรมแสดงค่าของตัวแปรที่ยังไม่ได้กำหนดค่าเริ่มต้น

```
1  /*  Program 2_7 : Output the values of uninitialized objects  */
2  #include <iostream>
3  using namespace std;
4  int main()
5  {
6      float f;
7      int i;
8      char c;
```

```
9      double d;  
10     cout << "f value is " << f << endl;  
11     cout << "i value is " << i << endl;  
12     cout << "c value is " << c << endl;  
13     cout << "d value is " << d << endl;  
14     return (0);  
15 }
```

บันทึกผลการทดลอง

ให้นักศึกษาแก้ไขบรรทัดของการประกาศตัวแปร โดยให้กำหนดค่าเริ่มต้นให้กับตัวแปร ดังนี้

```
float f = 2.8;  
int i = 120;  
char c = 'A';  
double d = 2e5;
```

บันทึกผลการทดลอง

การทดลองที่ 2_8 โปรแกรมแสดงค่าของตัวแปรที่กำหนดค่าเริ่มต้นแล้ว

```
1  /* Program 2_8 : Output the values of initialized objects */  
2  #include <iostream>  
3  #include <string>  
4  using namespace std;  
5  int main()  
6  {  
7      string ID = "50-6626-309-1";  
8      string Name = "Somchai Cheingpongpan";  
9      float Gpa = 2.89;  
10     short Age = 20;  
11     string Department = "Information Technology";  
12     string Room = "1RB";  
13     cout << "Student Code : " << ID << endl;  
14     cout << "Student Name : " << Name << endl;  
15     cout << "Department : " << Department << endl;
```

```

16     cout << "Room : " << Room << endl;
17     cout << "Age : " << Age << endl;
18     cout << "Gpa : " << Gpa << endl;
19     return (0);
20 }

```

บันทึกผลการทดลอง

คำสั่งการแสดงผลและนำเข้าข้อมูล

➤ คำสั่งการแสดงผลข้อมูล

สำหรับคำสั่งการแสดงผลข้อมูลในภาษา C++ จะใช้คำสั่ง cout ย่อมาจาก console output ทำหน้าที่แสดงผลข้อมูลออกบนคอนโซล(Console)หรือจอภาพ ในการใช้งานคำสั่ง cout จะต้องเรียกใช้ไฟล์ไลบรารี iostream และ namespace std ในโปรแกรมทุกครั้ง ซึ่งคำสั่งนี้จะต้องใช้ร่วมกับเครื่องหมาย << ต่อการแสดงผลหนึ่งข้อความ ซึ่งอาจเป็นค่าคงที่ ตัวแปร นิพจน์ก็ได้ ดังตัวอย่าง

```

cout << "Hello World";
cout << "C++" << endl;
cout << "C++" << " is a " << "breeze" << endl;
cout << "18 + 3 = " << (18 + 3) << endl;
cout << Hours << " hours is " << (Hours * 60) << " minutes" << endl;

```

➤ คำสั่งการนำเข้าข้อมูล

สำหรับคำสั่งการนำเข้าข้อมูลในภาษา C++ จะใช้คำสั่ง cin ย่อมาจาก console input เป็นคำสั่งทำหน้าที่รับค่าข้อมูลจากคีย์บอร์ด ในการใช้งานคำสั่ง cin จะต้องเรียกใช้ไฟล์ไลบรารี iostream และ namespace std ในโปรแกรมทุกครั้ง ซึ่งคำสั่งนี้จะต้องใช้ร่วมกับเครื่องหมาย >> ต่อการรับค่าหนึ่งค่า ซึ่งจะต้องเป็นตัวแปรเท่านั้น ดังตัวอย่าง

```

cin >> name;
cin >> Age;
cin >> Value1 >> Value2;
cin >> FValue;

```

การทดลองที่ 2_9 โปรแกรมแสดงการรับค่าข้อความ

```
1  /* Program 2_9 : Input value string by keyboard */
2  #include <iostream>
3  #include <string>
4  using namespace std;
5  int main()
6  {
7      string Name;
8      int Age;
9      cout << "Enter name : ";
10     cin >> Name;
11     cout << "Enter age : ";
12     cin >> Age;
13     cout << "\nHello, " << Name << "." << endl;
14     cout << "You have " << Age << " year old." << endl;
15     cout << "You are beginner programmer." << endl;
16     return (0);
17 }
```

บันทึกผลการทดลอง

การทดลองที่ 2_10 โปรแกรมแสดงการบวกค่าตัวเลข 2 ค่า

```
1  /* Program 2_10 : Add 2 value number */
2  #include <iostream>
3  using namespace std;
4  int main()
5  {
6      int Number1, Number2;
7      cout << "Program Addition 2 values." << endl ;
8      // input two number
9      cout << "Enter first number : ";
10     cin >> Number1;
11     cout << "Enter second number : ";
12     cin >> Number2;
13     // Display addition two value
14     cout << endl;
15     cout << "Sum " << Number1 << " + " << Number2;
16     cout << " = " << Number1 + Number2 << endl;
17     return (0);
18 }
```

บันทึกผลการทดลอง

แบบฝึกหัดท้ายบท

1. จงหาค่าเลขฐานสิบจากตัวเลขต่อไปนี้

- | | |
|------------------------|--------------------|
| 1.1 01001_2 <u>9</u> | 1.2 0374_8 _____ |
| 1.3 $A32E_{16}$ _____ | 1.4 4033_5 _____ |

2. ให้ตรวจสอบว่าการตั้งชื่อในข้อใดถูกต้องหรือไม่ถูกต้องตามหลักการตั้งชื่อของภาษา C++

- | | |
|-----------------------------------|---------------------------|
| 2.1 GPA <u>Yes</u> | 2.2 3CPO <u>No CPO3</u> |
| 2.3 Grade.pnt <u>No Grade_pnt</u> | 2.4 _dog <u>No dog</u> |
| 2.5 X-ray <u>No X_ray</u> | 2.6 Cost\$ <u>No Cost</u> |
| 2.7 ReturnV <u>Yes</u> | 2.8 main <u>No mainv</u> |

3. จากโปรแกรมการทดลองที่ 2_10 ถ้าหากใส่ข้อมูลเป็นตัวเลขทศนิยม โปรแกรมจะเป็นอย่างไร

Eorr

4. ตอบคำถามต่อไปนี้

- | | |
|---|-----------------------|
| 4.1 ชนิดข้อมูลใด ที่มีใช้หน่วยความจำเป็นจำนวน 16 บิต | <u>short</u> |
| 4.2 ชนิดข้อมูลใด ที่มีใช้หน่วยความจำเป็นจำนวน 32 บิต | <u>int,long,float</u> |
| 4.3 สามารถใช้ unsigned ประกาศร่วมกับชนิดข้อมูล float หรือ double ได้หรือไม่ | <u>Not</u> |

5. ให้นักศึกษาเขียนโปรแกรมเพื่อคำนวณหาพื้นที่ของรูปสี่เหลี่ยมผืนผ้า โดยมีการรับค่าความยาวและความกว้างเป็นตัวเลขจำนวนเต็ม มีรูปแบบการทำงานตามตัวอย่างด้านล่าง

```
Program Calculate Area Rectangle.
Enter Length : 20
Enter Widht : 20
Area of Rectagle = 400
```

6. ให้นักศึกษาเขียนโปรแกรมเพื่อคำนวณหาพื้นที่ของรูปวงกลมและความยาวของเส้นรอบรูปวงกลม โดยโปรแกรมมีการรับค่ารัศมีของวงกลมเป็นตัวเลขทศนิยม โดยใช้ค่า $\pi = 3.1415$ มีรูปแบบการทำงานตามตัวอย่างด้านล่าง

```
Program Calculate Area Circle.
Circle radius (real number) ? 5.1
Area of circle with radius 5.1 is 81.7104
Circumference is 32.0433
```

7. ให้นักศึกษาเขียนโปรแกรมเพื่อคำนวณราคาสินค้าที่รวมภาษีมูลค่าเพิ่มแล้ว โดยโปรแกรมมีการรับค่าราคาสินค้าและอัตราภาษีมูลค่าเพิ่ม(ร้อยละ) มีรูปแบบการทำงานตามตัวอย่างด้านล่าง

```
Enter price : 500
Enter vat (%) : 10
Net Price of product = 550
```

