



การเขียนโปรแกรมคอมพิวเตอร์

ทบทวนคำสั่งรับค่าแสดงผล นิพจน์ ตัวดำเนินการ

อาจารย์ โอภาส วงษ์ทวีทรัพย์ (อ.โอ๊ด)

ศูนย์ปฏิบัติการวิจัย และพัฒนาระบบสารสนเทศอันชาญฉลาด
ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

E-Mail :: oatcomster@gmail.com

ขอบเขตของเนื้อหา

- ทบทวนคำสั่งที่ใช้ในการรับข้อมูลเข้าผ่านทางคีย์บอร์ด (*Input Statement via Keyboard*)
- ทบทวนคำสั่งที่ใช้ในการแสดงผลลัพธ์ของข้อมูลออกสู่จอภาพ (*Output Statement via Monitor*)
- นิพจน์การคำนวณในภาษาโปรแกรม และลำดับความสำคัญของการคำนวณ (*Expression & Precedence of Operators*)
- ข้อกำหนดของการเก็บข้อมูลของเครื่องคอมพิวเตอร์

โครงสร้างของภาษาซี

- A C program contains
 - Preprocessor Directive Type Definitions
 - Function Prototypes Variables
 - Functions
- All program must contain single main() function.
- Return Default of Main Function is Integer.

โครงสร้างของภาษาซี(2)

```

/* Preprocessor directive */
#include <stdio.h>

main()           //Default return Integer
{               //Begin of main program
  ...           //Statement
  ...           //Statement
}               //End of main program

```

โครงสร้างของภาษาซี(3)

```

#include <stdio.h> //Preprocessor directive

void main()       //Function no return value
{               //Begin of main program
  ...           //Statement
  ...           //Statement
}               //End of main program

```

โครงสร้างของภาษาซี(4)

```

#include <stdio.h> //Preprocessor directive

int main()       //Function return value
{               //Begin of main program
  ...           //Statement
  return(0);    //Statement Return
}               //End of main program

```

ตัวอย่างของโปรแกรมในภาษาซี

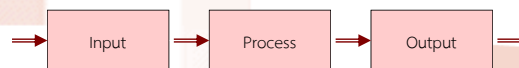
```

#include <stdio.h>
float PI = 3.1415927;
int main()
{
  float radias,area;
  printf("Input Radias : ");scanf("%f",&radias);
  area = PI * radias * radias;
  printf("Area of Circle : %f",area);
  return(0);
}

```

พฤติกรรมของโปรแกรมคอมพิวเตอร์

- A Normal Program :



is when a process has inputs

and produces complete outputs.

คำสั่งรับข้อมูลเข้า

- คำสั่งรับข้อมูลเข้านี้ เป็นคำสั่งที่ใช้ในการรับข้อมูลเข้าสู่เครื่องคอมพิวเตอร์
- โดยผ่านทางอุปกรณ์ที่มีชื่อเรียกว่า หน่วยรับข้อมูลเข้า (Input Devices) ชนิดคีย์บอร์ด (Keyboard)
- ในที่นี้ขอให้นักศึกษาทบทวนการใช้คำสั่ง scanf

คำสั่งแสดงผลของข้อมูล

- คำสั่งแสดงผลของข้อมูลนี้ เป็นคำสั่งที่ใช้ในการแสดงผล หลังการประมวลผลของคอมพิวเตอร์เสร็จเป็นที่เรียบร้อยแล้ว
- โดยผ่านทางอุปกรณ์ที่มีชื่อเรียกว่า หน่วยแสดงผลข้อมูล (Output Devices) ชนิดจอภาพ (Monitor)
- ในที่นี้ขอให้นักศึกษาทบทวนการใช้คำสั่ง printf

สรุปตัวควบคุมและลำดับหลัก

- Format Codes
 - %i or %d Signed Decimal Integer
 - %c or %s Display one Character or String
 - %e,E,f,g,G Real a Floating-point
- Control Character or Escape Sequence
 - \n Beginning a New line

โครงสร้างของนิพจน์

- Expression has two components ==> A + B , (A+4)/sqrt(8)
 - Operand may be Constant, Variable Name and Function Name(Return Value).
 - Example : A , B , 4 , 8 , sqrt()
 - Operator is Symbols that replaced process.
 - Example : + , - , * , / , ++ , -- , < , > etc.

นิพจน์ทางคณิตศาสตร์ และตรรกศาสตร์

- Mathematics Expression
 - contain mathematics symbols : + , - , * , / , ++ , -- etc.,
 - result return is value of calculation.
- Boolean Expression
 - contain relational symbols : < , > , <= , >= , == , !=
 - contain logical symbols : ! , && , || , & , |
 - Boolean value : zero(False) , nonzero (True).

ตัวดำเนินการเดี่ยว (Unary Operators)

Unary Operator is a statement contain one operand and one operand.

- + Unary plus Ex z = +x;
- Unary minus Ex z = -x;
- ++ Increment Ex ++x; x++;
- Decrement Ex --x; x--;

ตัวดำเนินการคู่ (Binary Operators)

Binary Operator is a statement contain one operator and two operands ex Z = X + Y;

- * Multiply / Divide
- % Remainder (modulus)
- + Binary plus - Binary minus
- Ex Z = X * Y; A = B/2 - 6; F = (D + 7) - C;

Relational & Equality Operators

- < Less than <= Less than or equal to
- > Greater than >= Greater than or equal to
- = Equal to != Not equal to

Ex
if (score < 50) printf("You must get grade F");

Logical Operators

- Result of Return Value of Logical Operators is
- Boolean Value :: Zero(False) and Non-Zero (True)

- && Logical Operator AND
- || Logical Operator OR
- ^ Logical Operators XOR

Ex if(score>=0 && score<=100) printf("Score Verify");

Shift and Bit-wise Operators

- << Shift left Ex X = 10010110;
- >> Shift right Y = 01001110;
- & Bitwise AND Z = X << 1; Z = 00101100
- | Bitwise OR Z = Y >> 2; Z = 00010011
- ^ Bitwise XOR Z = X ^ Y; Z = 11011000

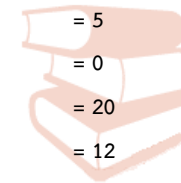
Assignment Operators

= *= /=
 %= += -=
 &= ^= |=
 <<= >>=



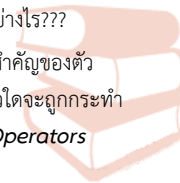
Assignment Operators(2)

X = 5; //Assigned X equal to 5
 X *= 5; //Multiple X by 5 = 25
 X /= 5; //Divide X by 5 = 5
 X %= 5; //Mod X by 5 = 0
 X += 20; //Add X by 20 = 20
 X -= 8; //Sub X by 8 = 12



Precedences of Operators

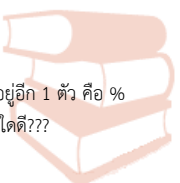
- จะเกิดอะไรขึ้นกับการเขียนนิพจน์ในรูปแบบนี้
 $result = a + b * c * d / 5 - 7 + f * g;$
- คุณคิดว่าคอมพิวเตอร์จะมีวิธีการคำนวณอย่างไร???
- ดังนั้นเราจึงจำเป็นต้องกำหนดลำดับความสำคัญของตัวดำเนินการ ว่าตัวดำเนินการแต่ละตัวนั้น ตัวใดจะถูกกระทำก่อนและหลัง เรียกว่า *Precedence of Operators*



Precedences of Operators

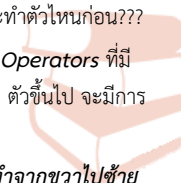
- โดยมีลำดับความสำคัญที่เราใช้กันบ่อยๆ คือ
 1. ถ้ามีวงเล็บต้องกระทำภายในวงเล็บก่อน
 2. ตัวดำเนินการ * และ /
 3. ตัวดำเนินการ + และ -

คำถาม ยังเหลือตัวดำเนินการที่เราใช้งานบ่อยๆ อยู่อีก 1 ตัว คือ % (Mod) หากแบบเอาเศษที่เหลือ) จะอยู่ในระดับใดดี???



Associativity

- แล้วถ้ามันมีแบบนี้ล่ะ $result = A * B + C / D;$
- เราจะเริ่มทำจากไหนนะ เห็นบอกว่า ต้องทำ * และ / ก่อน แล้ว * กับ / มีลำดับความสำคัญเท่ากัน จะทำตัวไหนก่อน???
- ในภาษาโปรแกรมจะมีการกำหนดว่า ถ้ามี *Operators* ที่มีลำดับความสำคัญเท่ากันจำนวนมากกว่า 1 ตัวขึ้นไป จะมีการกำหนดทิศทางของการดำเนินการ
- แบ่งเป็น 2 แบบ คือ ทำจากซ้ายไปขวา ทำจากขวาไปซ้าย

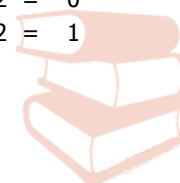


ตัวอย่างนิพจน์ที่น่าสนใจ



10%5 = 0 10%2 = 0
 15%5 = 0 11%2 = 1
 20%5 = 0 12%2 = 0
 25%5 = 0 13%2 = 1

Get???



ตัวอย่างนิพจน์ที่น่าสนใจ(2)

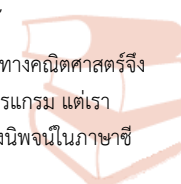
Number = 4725

4 = Number/1000
 7 = Number%1000/100
 2 = Number%1000%100/10 = Number%100/10
 5 = Number%1000%100%10
 = Number%10



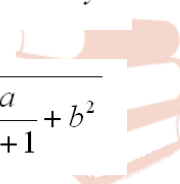
การเขียนนิพจน์ในภาษาซี

- อย่าลืมว่าการเขียนโปรแกรมนั้น จะต้องเขียนให้อยู่ภายในระนาบบรรทัดเดียวกันทั้งหมด
- ไม่สามารถเขียนตัวห้อย หรือตัวยกกำลังได้
- ดังนั้น นิพจน์ที่เราใช้โดยทั่วไปในสมการทางคณิตศาสตร์จึงไม่สามารถเขียนได้โดยตรง เมื่อเราเขียนโปรแกรม แต่เราจะต้องแปลงนิพจน์เหล่านั้นให้อยู่ในรูปของนิพจน์ในภาษาซีเสียก่อน



การเขียนนิพจน์ในภาษาซี(2)

1. $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$ 3. $a^2 + \frac{b}{x^2 + y^2}$
 2. $\frac{(x^2 + y^2)^2}{z^2 + 1}$ 4. $\sqrt{\frac{a}{x+1}} + b^2$



การเขียนนิพจน์ในภาษาซี(3)

1. $(-b + \sqrt{b*b - 4*a*c})/(2*a)$
2. $(x*x + y*y)*(x*x + y*y)/(z*z + 1)$
3. $(a*a) + b / (x*x + y*y)$
4. $\sqrt{a/(x + 1) + b*b}$

