

ใบเนื้อหา

เรื่อง การแสดงผลข้อมูลและการรับค่าข้อมูล

เมื่อต้องการแสดงผลและรับข้อมูลจะมีกระบวนการคำสั่งที่ทำหน้าที่ในการแสดงผลข้อมูลออกและทำหน้าที่ในการรับข้อมูลเข้า ซึ่งการแสดงผลลัพธ์ออกของคอมพิวเตอร์และการอ่านข้อมูลเข้าไปในคอมพิวเตอร์ จัดว่าเป็นคำสั่งพื้นฐานของการเขียนโปรแกรม ประกอบด้วย คำสั่งในการแสดงผลลัพธ์ และคำสั่งการรับค่าข้อมูล

1. การแสดงผลข้อมูล

คำสั่งแสดงผล หมายถึง คำสั่งที่สั่งให้เครื่องทำการแสดงผลข้อมูลที่ต้องการออกมาทางจอภาพ ในการแสดงผลสามารถแสดงผลได้ทั้ง ตัวเลข และ ตัวอักษร คำสั่งที่ใช้สำหรับการแสดงผลในภาษาซี ที่จะกล่าวในที่นี้ มี 2 คำสั่ง ดังต่อไปนี้

1.1 คำสั่ง printf()

คำสั่ง printf เป็นคำสั่งที่ใช้สำหรับทำการแสดงผลของข้อความหรือตัวแปรออกทางจอภาพ อยู่ใน ไลบรารี stdio.h มีรูปแบบการใช้งานดังนี้

```
printf("control string",variable list);
```

control string หมายถึง คำสั่งที่ใช้ในการควบคุมการแสดงผลข้อมูล ต้องอยู่ในเครื่องหมายคำพูด “ ” (Quotation mark) แบ่งออกเป็น 2 ส่วน คือ ข้อความ และรหัสรูปแบบ

ตัวอย่างที่ 1 โปรแกรมตัวอย่างในการแสดงผลข้อมูล ในแบบของตัวเลขจำนวนเต็มโดยใช้สัญลักษณ์ %d และ จำนวนจริง ด้วยสัญลักษณ์ %f หลังจากแสดงผลในแต่ละบรรทัด ให้ขึ้นบรรทัดใหม่ด้วย สัญลักษณ์ \n

```
#include <stdio.h>
#include <conio.h>
main()
{
    int a;
    float b;
    a=50;
    b=10.583;
```

```

printf("a =%d\n",a);
printf("b =%f\n",b);
printf("a =%5d\n",a);
printf("b =%10.4f\n",b);
printf("b =%-10.4f\n",b);
getch();
}

```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

```

a =50
b =10.583000
a =00050
b = 10.5830
b =10.5830

```

ตัวอย่างที่ 2 โปรแกรมตัวอย่างการพิมพ์ด้วยสัญลักษณ์พิเศษที่ขึ้นต้นด้วยเครื่องหมาย back slash ได้แก่ \n เพื่อขึ้นบรรทัดใหม่ \t เพื่อเว้นระยะในการแสดงผล

```

#include <stdio.h>
#include <conio.h>
main()
{
printf("12345678901234567890\n");
printf("\tExample line 1\n");
printf("Example line2\n");
printf("\"Example line3\"\n");
getch();
}

```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

```

12345678901234567890
      Example line 1
Example line2
"Example line3"

```

ผลลัพธ์ของโปรแกรมตัวอย่าง มีการใช้ \n หมายถึงขึ้นบรรทัดใหม่ \t หมายถึง เว้นระยะการพิมพ์แน่นอน จะเห็นว่า ข้อความ Example line 1 เว้นระยะการพิมพ์มาจากทางด้านซ้าย \" หมายถึง การพิมพ์ เครื่องหมาย “

1.2 คำสั่ง putchar()

คำสั่ง putchar() เป็นคำสั่งที่อยู่ในไลบรารี stdio.h ใช้ในการแสดงผลข้อมูลทางจอภาพทีละตัวอักษรที่มีรูปแบบการใช้งานดังนี้

```
putchar(char);
```

char หมายถึงตัวแปรชนิดตัวอักษรที่เขียนไว้ในเครื่องหมาย ‘ ‘

ตัวอย่างที่ 3 โปรแกรมตัวอย่างการแสดงผลข้อมูลด้วยคำสั่ง putchar()

```
#include <stdio.h>
#include <conio.h>
main()
{
    char first='O';
    putchar(first);
    putchar('K');
    getch();
}
```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

OK

จากตัวอย่างที่ 3 จะใช้ putchar(first) แสดงค่าตัวแปร first และได้อักษรตัว O แล้วใช้ putchar('K') ให้แสดงตัวอักษร K จะได้ผลลัพธ์ออกมาเป็น OK

1.3 คำสั่ง puts()

คำสั่ง puts() เป็นคำสั่งที่อยู่ในไลบรารี stdio.h ใช้ในการแสดงข้อความออกทางจอภาพ มีรูปแบบการใช้งานดังนี้

```
puts(str);
```

str หมายถึง ตัวแปรที่เก็บข้อมูลชนิดข้อความ หรือข้อความที่เขียนอยู่ในเครื่องหมาย คำพูด “ “

ตัวอย่างที่ 4 โปรแกรมตัวอย่างการแสดงผลข้อมูลด้วยคำสั่ง puts()

```
#include <stdio.h>
#include <conio.h>
main()
{
    char message[]="Computer ";
    puts(message);
    puts("Education");
    getch();
}
```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

Computer
Education

จากตัวอย่างที่ 4 จะใช้คำสั่ง puts ตั้งแสดงค่าตัวแปร message และได้ข้อความ Computer แล้วใช้คำสั่ง puts ให้แสดงข้อความ Education

2. การรับค่าข้อมูล

คำสั่งรับข้อมูล หมายถึง คำสั่งที่สั่งให้เครื่องรับข้อมูลเข้ามาเพื่อดำเนินการอย่างใดอย่างหนึ่งต่อไป การเขียนโปรแกรมแต่ละครั้ง โปรแกรมโดยทั่วไปจะมีการรับข้อมูล ประมวลผล และการแสดงผล การรับข้อมูลจึงเป็นคำสั่งที่สำคัญอีกคำสั่งหนึ่ง ที่สามารถทำให้ผู้ใช้ ติดต่อกับเครื่องคอมพิวเตอร์ได้ การรับข้อมูล สามารถรับข้อมูลได้ทั้งแบบที่เป็น ตัวเลข และ ตัวอักษร คำสั่งที่ใช้รับข้อมูลในภาษาซีมีดังนี้

2.1 คำสั่ง scanf()

คำสั่ง scanf อยู่ในไลบรารี stdio.h เป็นฟังก์ชันมาตรฐานใช้สำหรับรับข้อมูล เมื่อคำสั่งนี้ถูกสั่งให้ทำงานเครื่องจะหยุดเพื่อรอรับการป้อนข้อมูลจากผู้ใช้ โดยข้อมูลที่ป้อนจะปรากฏบนจอภาพ เมื่อป้อนข้อมูลเสร็จกด Enter ข้อมูลจะถูกส่งไปเก็บที่ตัวแปร variable และ ข้อมูลที่ป้อนลงไปจะถูกกำหนดรูปแบบโดย format code มีรูปแบบดังนี้

```
scanf ("format code",&variable);
```

format code หมายถึง การใช้รหัสรูปแบบ กำหนดชนิดของข้อมูลที่จะรับเข้าโดยรูปแบบของฟังก์ชัน scanf() จะใช้ชุดเดียวกับรหัสรูปแบบของฟังก์ชัน printf() สิ่งสำคัญควรจะกำหนดรหัสรูปแบบให้ตรงกับชนิดของตัวแปรในส่วน &variable

variable หมายถึง ตัวแปรที่ใช้สำหรับเก็บค่าของข้อมูลที่รับเข้ามา โดยจะต้องเขียนเครื่องหมาย & นำหน้า ยกเว้นตัวแปรที่เก็บข้อมูลชนิดข้อความ เท่านั้นที่สามารถใส่ชื่อตัวแปรได้ โดยไม่ต้องมีเครื่องหมาย & นำหน้า

ตัวอย่างที่ 5 โปรแกรมตัวอย่างการใช้คำสั่ง scanf เพื่อรับข้อมูลประเภทตัวเลข ด้วยสัญลักษณ์ %d เพื่อรับข้อมูลตัวเลขจำนวนเต็ม %f เพื่อรับข้อมูลประเภทตัวเลขจำนวนจริง และมีการแสดงผลลัพธ์ ด้วยคำสั่ง printf

```
#include <stdio.h>
#include <conio.h>
main()
{
    int n;
    float x;
    printf("Enter Integer Number :");
    scanf("%d",&n);
    printf("Enter Float Number :");
    scanf("%f",&x);
    printf("\nInteger Number =%d\n",n);
    printf("Floating Number =%.4f\n",x);
    getch();
}
```

ผลลัพธ์ที่ได้จะออกทางจอภาพ โดยมีรูปแบบดังนี้

```
Enter Integer Number :45
Enter Float Number :30.25
```

```
Integer Number =45
Float Number =30.2500
```

2.2 คำสั่ง getch()

คำสั่ง getch() อยู่ในไลบรารี stdio.h จะใช้สำหรับรับข้อมูลชนิดตัวอักษรเข้ามาทางแป้นพิมพ์ได้ครั้งละหนึ่งตัวอักษรเท่านั้น มีรูปแบบดังนี้

```
variable = getch();
```

variable หมายถึง ชื่อของตัวแปรชนิดตัวอักษร(char) ที่จะใช้เก็บค่าของตัวอักษรที่รับเข้ามาทางแป้นพิมพ์

ตัวอย่างที่ 6 โปรแกรมตัวอย่างการใช้คำสั่ง getch() เพื่อรับข้อมูล 1 ตัวอักษรแล้วแสดงข้อมูลที่ป้อนเข้าไปออกมา

```
#include <stdio.h>
#include <conio.h>
main()
{
    char ch;
    printf("Enter one character :");
    ch=getchar();
    printf("The character you typed is =%c\n",ch);
    getch();
}
```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

```
Enter one character :c
The character you typed is =c
```

2.3 คำสั่ง getch()

คำสั่ง getch() เป็นคำสั่งที่อยู่ในไลบรารี conio.h ที่ใช้สำหรับรับข้อมูลชนิดตัวอักษรเข้าทางแป้นพิมพ์ โดยรับได้ครั้งละ 1 ตัวอักษรเหมือนกับคำสั่ง getchar() แตกต่างกันที่คำสั่ง getch() เมื่อป้อนข้อมูลแล้วโปรแกรมจะทำงานต่อโดยทันทีไม่ต้องกด enter และจะไม่มีการแสดงข้อมูลที่รับให้เห็นทางจอภาพ มีรูปแบบดังนี้

```
variable = getch();
```

variable หมายถึง ชื่อของตัวแปรชนิดตัวอักษร ที่จะใช้เก็บค่าของข้อมูลที่รับเข้ามาทางแป้นพิมพ์

ตัวอย่างที่ 7 โปรแกรมตัวอย่างการใช้ คำสั่ง getch() โปรแกรมนี้จะทำการรับ ข้อมูลเข้ามาเก็บที่ตัวแปร a แล้วสั่งให้แสดงข้อมูลภายหลัง

```
#include <stdio.h>
#include <conio.h>
main()
{
    char a;
    printf("Enter one character to a :");
    a=getch();
    printf("\nThe character you to a is =%c\n",a);
    getch();
}
```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

```
Enter one character to a :b
The character you to a is =b
```

2.4 คำสั่ง gets()

คำสั่ง gets() เป็นคำสั่งที่อยู่ในไลบรารี stdio.h ที่ใช้สำหรับรับข้อมูลชนิดข้อความเข้าทางแป้นพิมพ์ โดยรับได้ครั้งละหลายตัวอักษร เมื่อป้อนข้อมูลจะทำงานต่อต้องกด enter มีรูปแบบดังนี้

```
gets(str);
```

str หมายถึง ชื่อของตัวแปรชนิดตัวอักษรแบบข้อความ ที่จะใช้เก็บค่าของข้อมูลที่รับเข้ามาทางแป้นพิมพ์

ตัวอย่างที่ 8 โปรแกรมตัวอย่างการใช้ คำสั่ง gets() โปรแกรมนี้จะทำการรับ ข้อมูลเข้ามาเก็บที่ตัวแปร a ในรูปแบบของข้อความ

```
#include <stdio.h>
#include <conio.h>
main()
{
    char a[30];
    printf("Enter you message :");
    gets(a);
    printf("\nYou message is :%s\n",a);
    getch();
}
```

ผลลัพธ์ที่ได้จะออกทางจอภาพโดยมีรูปแบบดังนี้

```
Enter you message :computer
You message is :computer
```

3. บทสรุป

ในการแสดงผลข้อมูลและรับค่าข้อมูลจะมีกระบวนการคำสั่งที่ทำหน้าที่ในการ แสดงผลข้อมูลออกและทำหน้าที่ในการรับข้อมูลเข้า ซึ่งประกอบด้วยคำสั่ง printf() putchar() puts() scanf() getchar() getch() และ gets()