ฟังก์ชัน printf()

เราได้เคยใช้งานฟังก์ชัน printf() ในตัวอย่างการเขียนโปรแกรมภาษาซีที่ผ่านมาบ้างแล้ว ในส่วนนี้จะกล่าวถึงการส่งค่าตัวแปรเพื่อแสดงผลผ่าน printf() โดยอธิบายจากการทำตัวอย่างต่อไปนี้

1. ให้ path อยู่ในตำแหน่ง /home/pi/c_codes
2. สร้างไฟล์ printf_exam.c ด้วยโปรแกรม nano
3. เขียนคำสั่งต่อไปนี้

```
#include <stdio.h>

int midterm;
int final;
int total;

int main(){
        midterm = 53;
        final = 28;
        total = midterm + final;

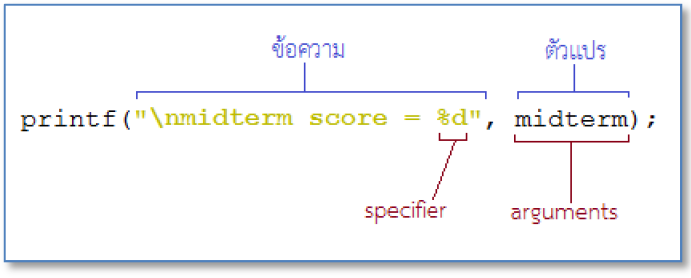
        printf("\nmidterm score = %d", midterm);
        printf("\nfinal score = %d",  final);
        printf("\n======================");
        printf("\ntotal score = %d", total);
        printf("\n======================\n");
}
```

4. บันทึก คอมไพล์ และรัน

```
pi@raspberrypi:~/c_codes $ gcc -o score printf_exam.c
pi@raspberrypi:~/c_codes $ ./score

midterm score = 53
final score = 28
======================
total score = 81
======================
pi@raspberrypi:~/c_codes $
```

จะเห็นว่า printf() เพิ่มส่วนที่เป็นตัวแปรเข้ามา โดยส่วนของข้อความและตัวแปรจะถูกคั่นด้วยเครื่องหมายจุลภาค (,) แบ่งส่วนกันอย่างชัดเจน ในส่วนของข้อความ หมายถึงข้อความที่ต้องการแสดงออกไปทางหน้าจอ แต่จะพบตัวอักขระพิเศษในข้อความเป็นเครื่องหมายเปอร์เซ็น (%) ตามด้วยตัวอักขระ d เราเรียกว่า specifier เมื่อข้อความ “midterm score = %d” ถูกนำไปแสดงผลหน้าจอไม่แสดง specifier แต่จะ



นำค่าจาก arguments ที่ได้รับจากตัวแปรไปใส่แทนที่ จะเห็นได้จากผลรันที่ผ่านมา specifier ที่นิยมใช้ร่วมกับคำสั่ง printf ได้แก่

specifier	ผลที่ได้	ตัวอย่าง
d	แสดงข้อมูลในรูปจำนวนเต็ม	392
f	แสดงข้อมูลในรูปจำนวนจริง	392.65
c	แสดงข้อมูลในรูปตัวอักขระ	a
s	แสดงข้อมูลในรูปข้อความ	sample

ในหนึ่งคำสั่ง printf สามารถมีได้หลาย specifier และแต่ละ specifier จะทำงานร่วมกับ argument กล่าวคือหากในข้อความมี specifier ปรากฏอยู่ 3 ตำแหน่ง แสดงว่าต้องมี argument จำนวน 3 ตัว เพราะ argument ตัวแรกจะส่งค่าไปยัง specifier ตัวแรก argument ตัวที่สองก็ส่งค่าไปให้ specifier ตัวที่ 2 เป็นอย่างนี้ไปตามลำดับ เพื่อให้เข้าใจมากขึ้นให้ทำการแก้ไข printf_exam.c ตามภาพต่อไปนี้

```c
#include <stdio.h>

int midterm;
int final;

int main(){
        midterm = 53;
        final = 28;
        printf("\nmidterm[%d] + final[%d] = total[%d]\n",midterm, final, midterm + final);
}
```

คอมไพล์ใหม่แล้วรันจะได้ผลดังนี้

```
pi@raspberrypi:~/c_codes $ gcc -o score printf_exam.c
pi@raspberrypi:~/c_codes $ ./score

midterm[53] + final[28] = total[81]
pi@raspberrypi:~/c_codes $
```

อธิบายด้วยภาพดังนี้

