

ปฏิบัติการครั้งที่ 3

โครงสร้างควบคุมแบบมีเงื่อนไขและการทำซ้ำ

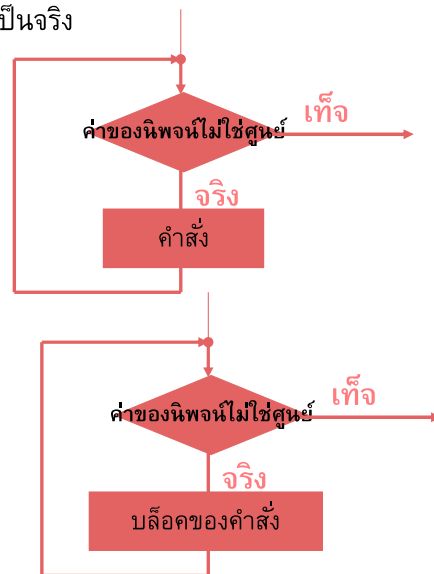


while statements

▶ คำสั่งที่ทำให้เกิดการซ้ำในขณะที่เงื่อนไขเป็นจริง

▶ while (นิพจน์)
คำสั่ง;

▶ while (นิพจน์) {
คำสั่ง1; คำสั่ง2;
... คำสั่งn;
}



ตัวอย่าง

- ▶ จงเขียนโปรแกรมเพื่อทำการนับจำนวนหลักของเลขจำนวนเต็มที่ไม่เป็นลบจนกว่าไม่ต้องการทำต่อ
- ▶ วิเคราะห์
 - ผลลัพธ์ จำนวนหลัก (digits)
 - ข้อมูลนำเข้า เลขจำนวนเต็ม 1 จำนวน (number)
ทำงานต่อหรือไม่ (ans)
 - วิธีประมวลผล
 - รับข้อมูล
 - จำนวนหลักของเลขจำนวนเต็มใดๆ จะเท่ากับจำนวนรอบของการหารด้วย 10 จนกระทั่งตัวตั้งของการหารเป็น 0
 - แสดงผล
 - ถามความต้องการทำงานต่อหรือไม่?

ตัวอย่าง (ต่อ)

```
#include <stdio.h>
int main() {
    int digits = 0, number;

    printf("Enter any integer number : ");
    scanf("%d", &number);

    while (number > 0) {
        number /= 10;
        digits += 1;
    }

    printf("No. of digits of your input is %d\n", digits);

    return 0;
}
```

do - while statements

- ▶ คำสั่งที่ทำให้เกิดการซ้ำถ้าเงื่อนไขยังคงเป็นจริง

- ▶ do

คำสั่ง;

while (นิพจน์);

- ▶ do {

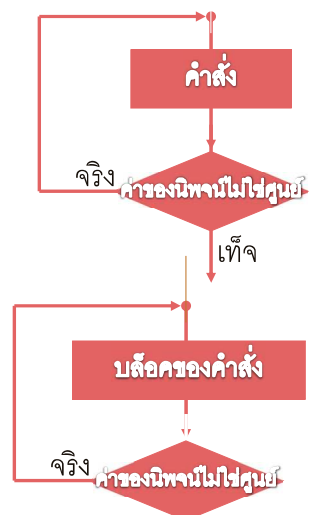
คำสั่ง1;

คำสั่ง2;

...

คำสั่งn;

} while (นิพจน์);



112 Structured
5 Programming

5

```
#include <stdio.h>
```

```
int main ( ) {
```

```
    int    digits = 0, number;
```

```
    char  ans;
```

```
    do {
```

```
        printf("Enter any integer number : ");
```

```
        scanf("%d", &number);
```

```
        while (number > 0) {
```

```
            number /= 10;
```

```
            digits += 1;
```

```
        }
```

```
        printf("No. of digits of your input is %d\n", digits);
```

```
        printf("Do you want to continue (y/Y)? ");
```

```
        scanf("%[yY]", &ans)
```

```
    } while (ans == 'y' || ans == 'Y');
```

```
    return 0;
```

```
}
```

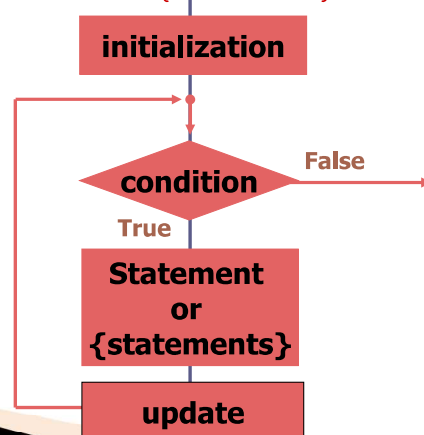
6

for statements

- ▶ คำสั่งที่ทำให้เกิดการซ้ำที่ทราบจำนวนรอบ

- ▶ for (initialization; condition; update)

statement; หรือ {statements}



7