

## Last Lecture

1

### □ โครงสร้างการทำซ้ำ Loop Structure

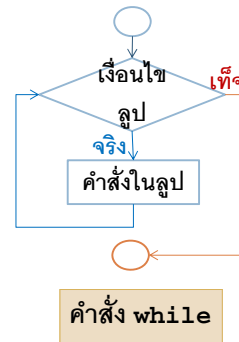
#### □ While Statement

##### □ ลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ

- รู้จำนวนรอบ
- ทำซ้ำจนกระทั่งเงื่อนไขการทำซ้ำเป็นเท็จ

##### □ การหยุดวนรอบ

- Break, Continue, Exit, และ Return



## This Lecture

2

### □ โครงสร้างการทำซ้ำ Loop Structure

- ทบทวนลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ
- ความแตกต่างของการหยุดวนรอบโดยใช้คำสั่ง
  - Break, Continue, Exit, และ Return
- การใช้คำสั่ง For
  - ความแตกต่างของคำสั่ง For และ While
- การใช้คำสั่ง do While
- การใช้ Nested Loop

## ลักษณะการควบคุมลูป (ทบทวน)

3

เรามีแนวทางใช้ลูปเพื่อควบคุมการจบลูป อยู่ 2 วิธีคือ

### 1. เมื่อทราบจำนวนรอบ

- ควบคุมด้วยจำนวนรอบ (Counter-Controlled Loop)

### 2. เมื่อไม่ทราบจำนวนรอบ

- ควบคุมด้วยข้อมูลปัด (Sentinel-controlled)
- ใช้ลูปที่ควบคุมด้วย flag (flag-controlled loops)
- ควบคุมด้วย EOF (end of file)

## ลักษณะการควบคุมลูป

4

### เมื่อทราบจำนวนรอบ:

- ต้องกำหนดตัวแปรและค่าเริ่มต้นสำหรับนับจำนวนครั้งในการทำซ้ำ
- เงื่อนไขในการทำซ้ำถูกกำหนดโดยใช้ตัวแปรดังกล่าว
- ต้องมีการกำหนดการเพิ่มค่าหรือลดลงของค่าในตัวแปร

ตัวอย่าง : โปรแกรมแสดงค่าของตัวแปรสำหรับนับรอบการทำซ้ำ

```
#include <stdio.h>

int main ()
{
    /* counter */
    int a = 10;
    /* while loop execution */
    while( a < 20 )
    {
        printf("value of a: %d\n", a);
        a++;
    }

    return 0;
}
```

counterControl.c

## ลักษณะการควบคุมลูป

#include <stdio.h>

เมื่อไม่ทราบจำนวนรอบ:

- ต้องกำหนดเงื่อนไขการทำงานซ้ำโดยใช้ตัวแปร
- ตัวแปรดังกล่าวต้องเปลี่ยนค่าภายในลูป
  - เพื่อหยุดการทำงานซ้ำ

ตัวอย่าง : โปรแกรมให้ผู้ใช้เดาตัวเลข โดยกำหนดให้ตัวเลขที่ถูกต้องคือ 3

เงื่อนไข

while(number != 3)

```
{
    printf("You are wrong. Try again.\n");
    printf("Guess Number [1-5]:");
    scanf("%d",&number);
}
```

การเปลี่ยนค่าตัวแปร

scanf("%d",&number);

return 0;

guessNum.c

CS 102 Week 06 By Natsuda Kaothanthong

## This Lecture

### โครงสร้างการทำซ้ำ Loop Structure

- ▣ ทบทวนลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ
- ▣ การหยุดวนรอบ
  - Break, Continue, Exit, และ Return
- ▣ For
- ▣ do While
- ▣ Nested Loop

CS 102 Week 06 By Natsuda Kaothanthong

## การหยุดวนรอบ

หยุดอย่างปกติ

- ▣ หยุดการวนรอบเมื่อเงื่อนไขให้วนรอบ (loop condition) เป็นเท็จ (false)

หยุดด้วยคำสั่งในโปรแกรม

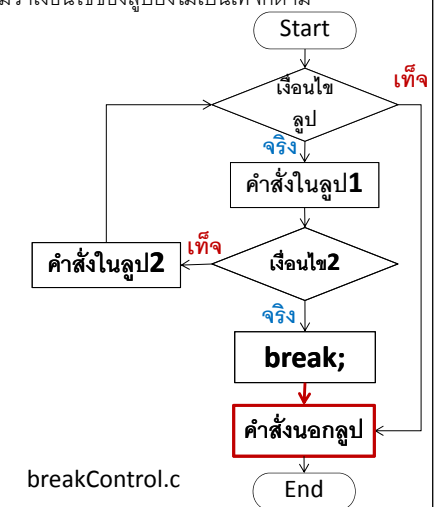
- ▣ คำสั่ง break
- ▣ คำสั่ง continue
- ▣ คำสั่ง exit
- ▣ คำสั่ง return

CS 102 Week 06 By Natsuda Kaothanthong

## The break Statement

- ▣ คำสั่ง break บังคับโปรแกรมออกจากลูป ถึงแม้ว่าเงื่อนไขของลูปยังไม่เป็นเท็จก็ตาม

```
int main()
{
    int x = 0;
    while (x<10)
    {
        x = x+1;
        if (x==5)
        {
            break;
        }
        printf("%d ",x);
    }
    printf("Out of loop\n");
    return 0;
}
```



breakControl.c

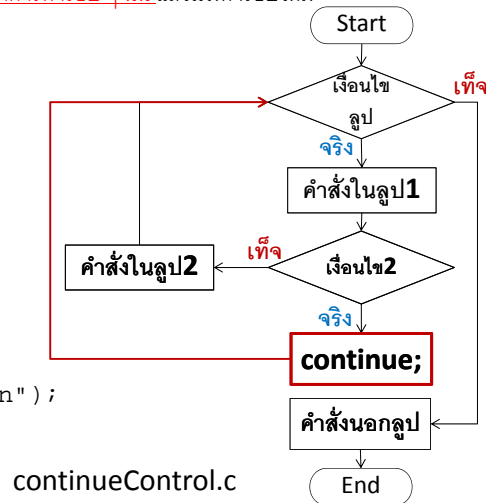
CS 102 Week 06 By Natsuda Kaothanthong

## The `continue` Statement

9

- คำสั่ง `continue` บังคับโปรแกรมหยุดการทำการรอบ ๆ นั้น แล้วเริ่มทำการรอบใหม่

```
int main()
{
    int x = 0;
    while (x<10)
    {
        x = x+1;
        if (x==5)
        {
            continue;
        }
        printf("%d ",x);
    }
    printf("Out of loop\n");
    return 0;
}
```



continueControl.c

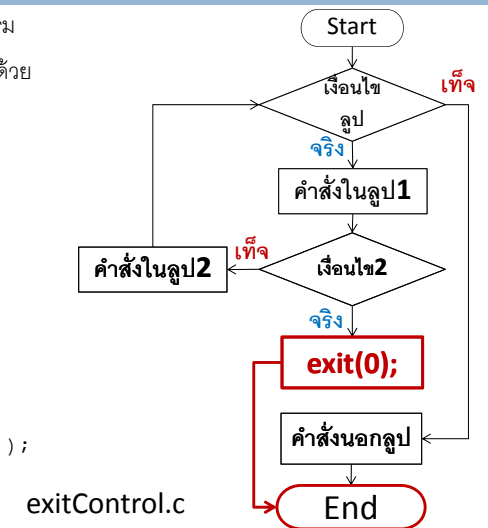
CS 102 Week 06 By Natsuda Kaothanthong

## The `Exit` Statement

10

- คำสั่ง `exit` บังคับให้ออกจากโปรแกรม
- ต้องใส่ `#include <stdlib.h>` ด้วย

```
int main()
{
    int x = 0;
    while (x<10)
    {
        x = x+1;
        if (x==5)
        {
            exit(0);
        }
        printf("%d ",x);
    }
    printf("Out of loop\n");
    return 0;
}
```



exitControl.c

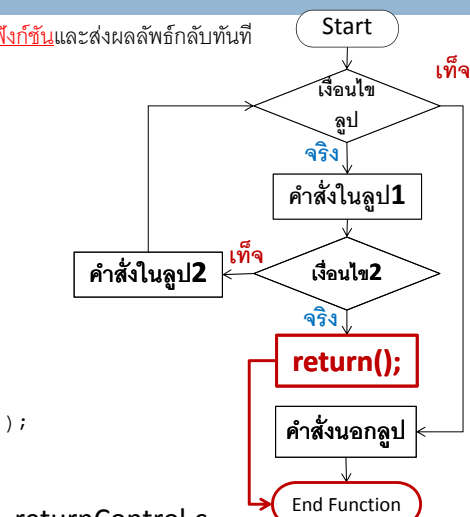
CS 102 Week 06 By Natsuda Kaothanthong

## The `return` Statement

11

- คำสั่ง `return` เป็นการบังคับให้จบฟังก์ชันและส่งผลลัพธ์กลับทันที

```
int main()
{
    int x = 0;
    while (x<10)
    {
        x = x+1;
        if (x==5)
        {
            return(0);
        }
        printf("%d ",x);
    }
    printf("Out of loop\n");
    return 0;
}
```



returnControl.c

CS 102 Week 06 By Natsuda Kaothanthong

## This Lecture

12

- โครงสร้างการทำซ้ำ Loop Structure
  - ทบทวนลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ
  - การหยุดวนรอบ
    - Break, Continue, Exit, และ Return
  - For
  - do While
  - Nested Loop

CS 102 Week 06 By Natsuda Kaothanthong

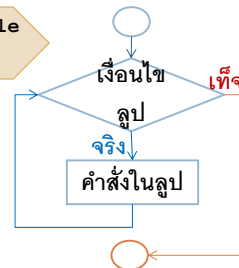
## คำสั่งทำซ้ำ For

13

ส่วนประกอบสำคัญในการใช้คำสั่งทำซ้ำคือ

1. การกำหนดค่าเริ่มต้นให้กับตัวแปรที่ควบคุมจำนวนการทำซ้ำ
2. เงื่อนไขที่ต้องการตรวจสอบ
3. การปรับค่าตัวแปรที่ใช้ในกำควบคุมการทำซ้ำ

คำสั่ง while  
และ for



□ รูปแบบการใช้คำสั่ง For ในภาษาซี

```
for( กำหนดค่าเริ่มต้นให้กับตัวแปร ; เงื่อนไขที่ต้องการตรวจสอบ ; ปรับค่าตัวแปร )
{
    คำสั่งในลูป ;
}
```

## คำสั่งทำซ้ำ For

14

```
for( กำหนดค่าเริ่มต้นให้กับตัวแปร ; เงื่อนไขที่ต้องการตรวจสอบ ; ปรับค่าตัวแปร )
{
    คำสั่งในลูป ;
}
```

**For**

```
#include <stdio.h>
int main ()
{
    /* counter */
    int a;
    /* for loop execution */
    for( a = 10; a < 20; a++)
    {
        printf("value of a: %d\n", a);
    }
    return 0;
}
```

ต้องประกาศตัวแปรก่อนใช้

**While**

```
#include <stdio.h>
int main ()
{
    /* counter */
    int a = 10;
    /* while loop execution */
    while( a < 20 )
    {
        printf("value of a: %d\n", a);
        a++;
    }
    return 0;
}
```

## ตัวอย่างแปลงคำสั่ง while เป็นคำสั่ง for

15

□ ต้องการอ่านตัวเลขเข้ามา 5 ตัวและแสดงผลทีละตัว ทำได้ 2 วิธี

นับขึ้น

```
i = 0;
while (i < 5)
{
    scanf("%d", &data);
    printf("%d ", data);
    i++;
}
```

```
for ( ___; ___; ___ )
{
    scanf("%d", &data);
    printf("%d ", data);
}
```

นับลง

```
i = 5;
while (i > 0)
{
    scanf("%d", &data);
    printf("%d ", data);
    i--;
}
```

```
for ( ___; ___; ___ )
{
    scanf("%d", &data);
    printf("%d ", data);
}
```

## การแปลงลูป while เป็น for

```
#include <stdio.h>
int main()
{
    float nextNum, sum = 0.0;
    int count, totalNumbers;

    scanf("%d", &totalNumbers);

    count = 0;
    while (count < totalNumbers)
    {
        scanf("%f", &nextNum);
        sum += nextNum;
        count++;
    }
    printf("Sum was %f\n", sum);
    printf("Mean was %f\n",
           sum/count);

    return 0;
}
```

```
#include <stdio.h>
int main()
{
    float nextNum, sum = 0.0;
    int count, totalNumbers;

    scanf("%d", &totalNumbers);

    printf("Sum was %f\n", sum);
    printf("Mean was %f\n",
           sum/count);

    return 0;
}
```

## ตัวอย่าง

17

- เขียนโปรแกรมสูตรคูณแม่ 2 โดยใช้ For

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

```
int main()
```

```
{
```

```
    int i; // Declare counting variable
```



```
}
```

2	*	1	=	2
2	*	2	=	4
2	*	3	=	6
2	*	4	=	8
2	*	5	=	10
2	*	6	=	12
2	*	7	=	14
2	*	8	=	16
2	*	9	=	18
2	*	10	=	20
2	*	11	=	22
2	*	12	=	24

## ตัวอย่าง

18

1. จงบอกผลของการการทำงานของโปรแกรมนี้

```
for(i = 1; i < 18; i += 3)
{
    printf("Value: %d", i);
}
```

## ตัวอย่าง

19

2. จงบอกผลของการการทำงานของโปรแกรมนี้

```
for(i = 1; i < 10; i++)
{
    if (i == 4)
    {
        break;
    }
    else
    {
        printf("%d ", i);
    }
}
printf("Out of loop\n");
```

breakControl.c

## ตัวอย่าง

20

3. จงบอกผลของการการทำงานของโปรแกรมนี้

```
for(i = 1; i < 10; i++)
{
    if (i == 4)
    {
        continue;
    }
    else
    {
        printf("%d ", i);
    }
}
printf("Out of loop\n");
```

continueControl.c

## ตัวอย่าง

21

### 4. จงบอกผลของการกระทำของโปรแกรมนี้

```
int found = 1;
int i;
for(i = 1; found; i+=3)
{
    if (i >= 10)
    {
        found = 0;
    }
    else
    {
        printf("%d ", i);
    }
}
printf("Out of loop\n");
```

CS 102 Week 06 By Natsuda Kaothanthong

flagControl.c

## This Lecture

22

### โครงสร้างการทำซ้ำ Loop Structure

- ▣ ทบทวนลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ
- ▣ การหยุดวนรอบ
  - Break, Continue, Exit, และ Return
- ▣ For
- ▣ do While
- ▣ Nested Loop

CS 102 Week 06 By Natsuda Kaothanthong

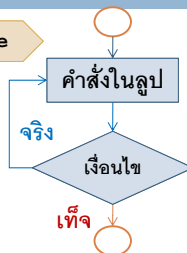
## คำสั่งทำซ้ำ Do While

23

```
do
{
    คำสั่ง;
}
while (expression);
```

มี semicolon ปิด ;

คำสั่ง do while



- ▣ รูปแบบ do..while ทำ **statement** หนึ่งรอบแล้วจึงเทียบเงื่อนไขเพื่อวนรอบซ้ำ
- ▣ รูปแบบนี้ไม่มีเงื่อนไขดักก่อนเข้าลูปเหมือน while และ for

CS 102 Week 06 By Natsuda Kaothanthong

## คำสั่งทำซ้ำ Do While

24

```
int value = -999;
int sum = 0;
do
{
    printf("Enter value: ");
    scanf("%d",&value);
    if(value!=-999)
    {
        sum = sum+value;
    }
}while(value!=-999);
printf("Out of Loop\n");
```

sum\_doWhile.c

```
int value = -999;
int sum = 0;
while(value!=-999)
{
    printf("Enter value: ");
    scanf("%d",&value);
    if(value!=-999)
    {
        sum = sum+value;
    }
}
printf("Out of Loop\n");
```

sum.c

CS 102 Week 06 By Natsuda Kaothanthong

```

1  #include <stdio.h>
2
3  int main()
4  {
5      int num;
6      char choice;
7      do
8      {
9          printf("Guess a number [1-5]:");
10         scanf("%d", &num);
11         if((num<3) || (num>3))
12         {
13             if(num<3)
14             {
15                 printf("Too small\n");
16             }
17             else if(num>3)
18             {
19                 printf("Too large\n");
20             }
21             choice = 'Y';
22         }
23         else
24         {
25             printf("Correct.\n");
26             choice = 'N';
27         }
28     }while(choice=='Y');
29     return 0;
30 }

```

CS 102 Week 06 By Natsuda Kaothanthong

## This Lecture

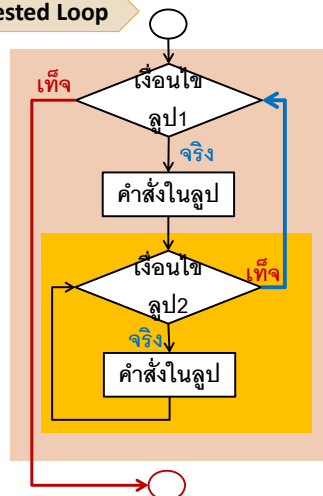
- โครงสร้างการทำซ้ำ Loop Structure
  - ▣ ทบทวนลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ
  - ▣ การหยุดวนรอบ
    - Break, Continue, Exit, และ Return
  - ▣ For
  - ▣ do While
  - ▣ Nested Loop

CS 102 Week 06 By Natsuda Kaothanthong

## Nested Loop

การใช้ลูป ซ้อน ลูป เพื่อแก้ปัญหา

### Nested Loop



```

int main()
{
    int outer = 1;
    int inner = 1;
    while(outer <= 3)
    {
        printf("Line: %d : ", outer);
        inner = 1;
        while(inner <= 5)
        {
            printf("%d ", inner);
            inner++;
        }
        printf("\n");
        outer++;
    }
    return 0;
}

```

CS 102 Week 06 By Natsuda Kaothanthong

## Nested For

```

int i,j;
for(i = 1; i <= 3; i++)
{
    printf("---Multiplication of %d:--- \n", i);

    for(j = 1; j <= 12; j++)
    {
        printf("%d * %2d = %3d\n", i, j, i*j);
    }
}
return 0;

```

CS 102 Week 06 By Natsuda Kaothanthong

# Nested For to Nested While

29

```
int i,j;
for(i = 1; i <= 3; i++)
{
    printf("---Multiplication of %d:--- \n",i);
    for(j = 1; j <= 12; j++)
    {
        printf("%d * %2d = %3d\n", i,j, i*j);
    }
}
return 0;
```

For

# Conclusion

30

- โครงสร้างการทำซ้ำ Loop Structure
  - ทบทวนลักษณะการควบคุมจำนวนครั้งของการทำซ้ำ
    - การนับรอบ
    - การใช้เงื่อนไข และ ตัวแปรภายในลูปเพื่อกำหนดจำนวนครั้งในการทำซ้ำ
  - การหยุดวนรอบ
    - Break, Continue, Exit, Return
  - การทำซ้ำโดยใช้คำสั่ง For
    - ความแตกต่างของคำสั่ง For และคำสั่ง While
  - การทำซ้ำโดยใช้คำสั่ง do While
    - ความแตกต่างของคำสั่ง do While และคำสั่ง For และ While
  - Nested Loop
    - การทำลูปซ้อนลูป และ ตัวอย่างการใช้งาน