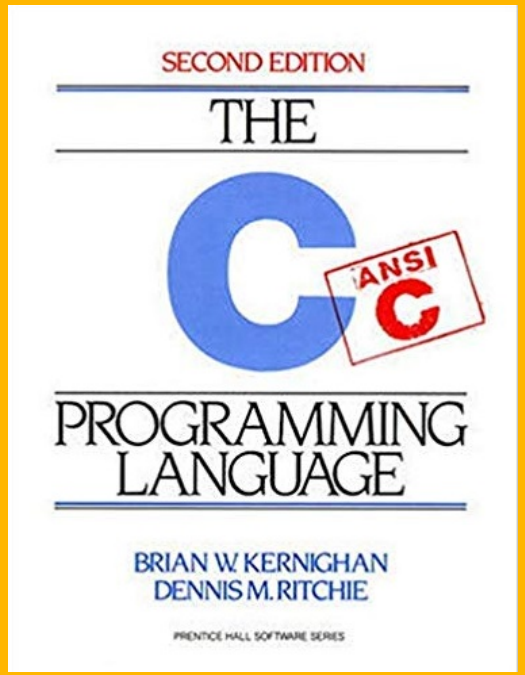


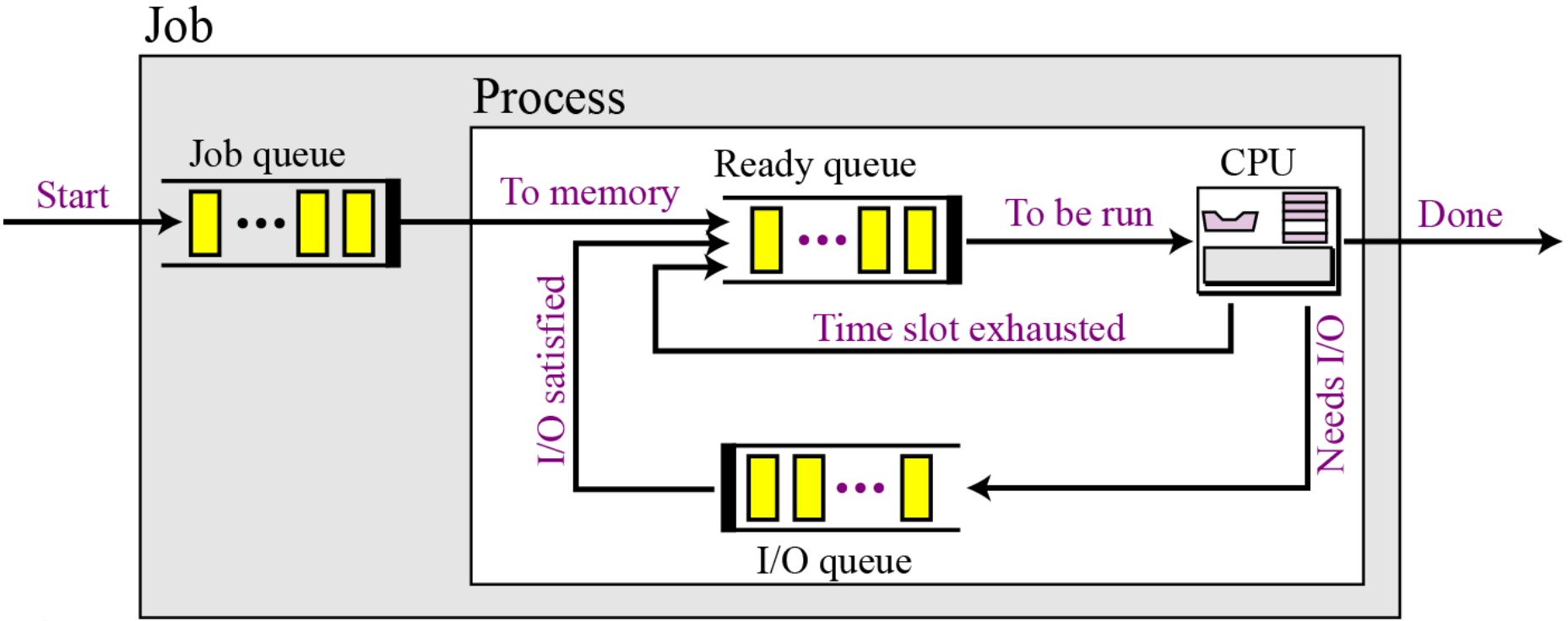
Introduction to Computer Science Fall 2022 #19

Chi-Jen Wu

期末考資訊

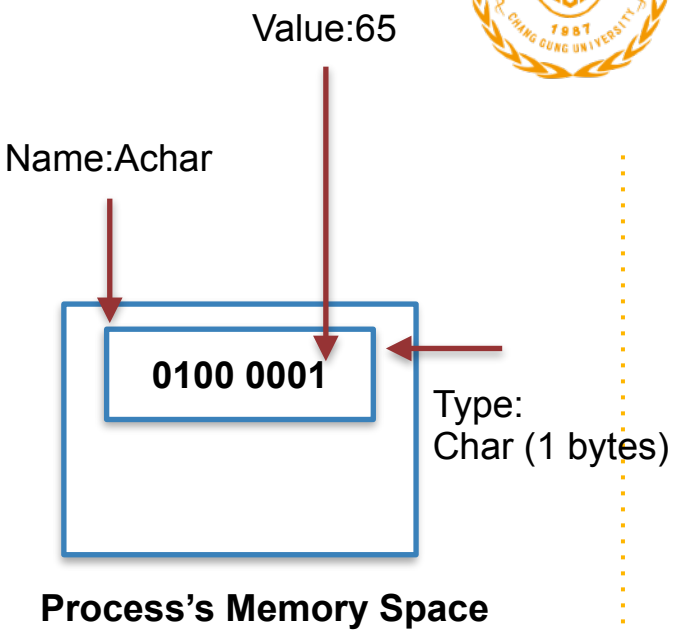
- 2023年 1月 12 號 星期四 下午三點
- **手寫考卷**
- **五題 C語言程式題 一題20分**
  - Variables & Flow of Control
  - Function
  - 包含以上範圍

程式編譯好後開始執行


Job
Process
Job queue
Start
To memory
Ready queue
CPU
To be run
Done
Time slot exhausted
I/O satisfied
Needs I/O
I/O queue

Variables

- 值 (Value)
  - 在記憶體實際的二進位值
- 名字 (Name)
  - 在程式裡的代號
- 型別 (Type)
  - 在記憶體所佔的空間



Variables 補遺

- Global
  - it is visible (hence accessible) throughout the program
- Static
  - Remains in memory while the program is running
- Const
  - The constant variables can be initialized once only
  - Read-only variable
- Volatile
  - Real-time update
  - Memory-mapped peripheral registers

```c
#include <stdio.h>
/* global variable declaration */
int rows = 5;
void print_spaces(int r) {
    /* local variable declaration */
    int i;
}

void print_stars(int r) {
    /* local variable declaration */
    int i;
}

int main() {
    /* local variable declaration */
    int j;
    for (int i = rows; i >= 1; --i) {
        print_spaces(i);
        print_stars(i);
    }

    return 0;
}
```


Function Scope Rules

在C語言裡，很嚴謹的方式規範變數的參照範圍 (scope)

Static Variables

- 滿常用變數型態
  - 最好記起來！
  - 一開始是零
- 很像global var
  - Scope 不一樣
  - 只能在func裡

```c
#include<stdio.h>
int gCOUNT = 0;
int fun() {
    static int count = 0;
    count++;
    gCOUNT++;
    return count;
}

int main() {
    for (int i=0; i< 10;i++) {
        printf("fun count=%d gCOUNT=%d\n", fun(), gCOUNT);
    }
    return 0;
}
```

```
fun count=1 gCOUNT=0
fun count=2 gCOUNT=1
fun count=3 gCOUNT=2
fun count=4 gCOUNT=3
fun count=5 gCOUNT=4
fun count=6 gCOUNT=5
fun count=7 gCOUNT=6
fun count=8 gCOUNT=7
fun count=9 gCOUNT=8
fun count=10 gCOUNT=9


...Program finished with exit code 0
Press ENTER to exit console.
```

Const Variables

- 常用變數型態
  - 不可變的數
  - Pi
  - E
  - 常數
- 不可更動
  - 當你希望**變數**不要被別人改變時
  - 例如校名！

```c
#include<stdio.h>

int main() {
    int const i = 10;
    printf("const i=%d\n", i);
    i = 1;
    return 0;
}
```

input

Compilation failed due to following error(s).

```
main.c:6:7: error: assignment of read-only variable 'i'
    6 |     i = 1;
      |       ^
```

Volatile Variables

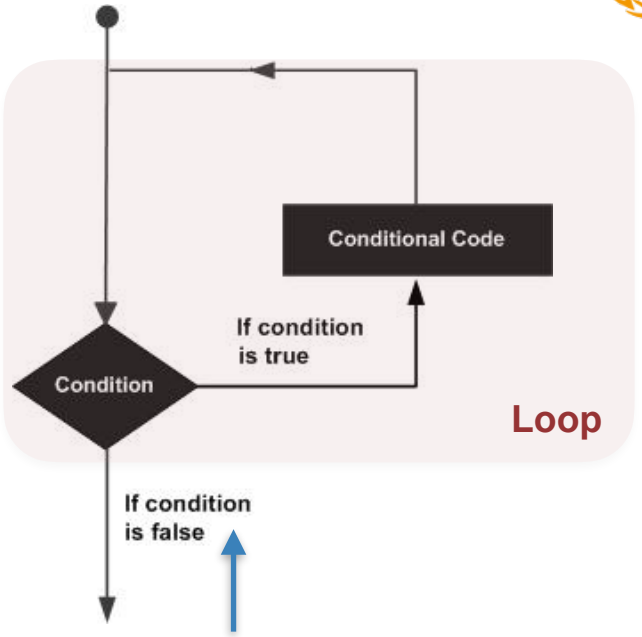
- 少用
  - 但硬體公司每天用
- 如果變數需要即時更新
  - 例如0.0000000001
  - 內要更新
  - 那你就要加上這個

```c
#include<stdio.h>

int main() {
    volatile int a=0;
    int volatile b=19;
    printf("%d %d\n", a, b);
    a=10;
    b=9;
    printf("%d %d\n", a, b);

    return 0;
}
```

```
0 19
10 9


...Program finished with exit code 0
Press ENTER to exit console.
```

Flow of Control

- if
- If else
- If else if else
- Switch case
- While
- For
- Do while loop
- For loop
- Do …. While loop
- Nested loops
- Break
- Continue

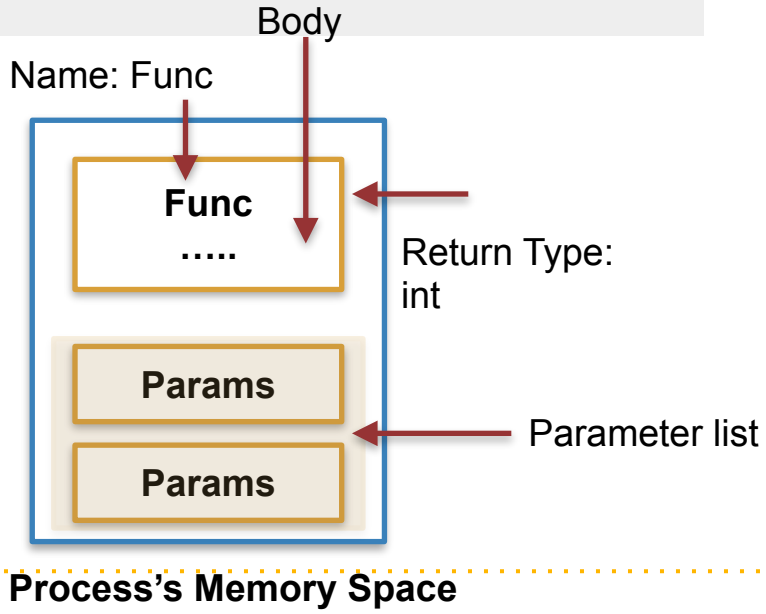


Condition 決定要不要轉彎

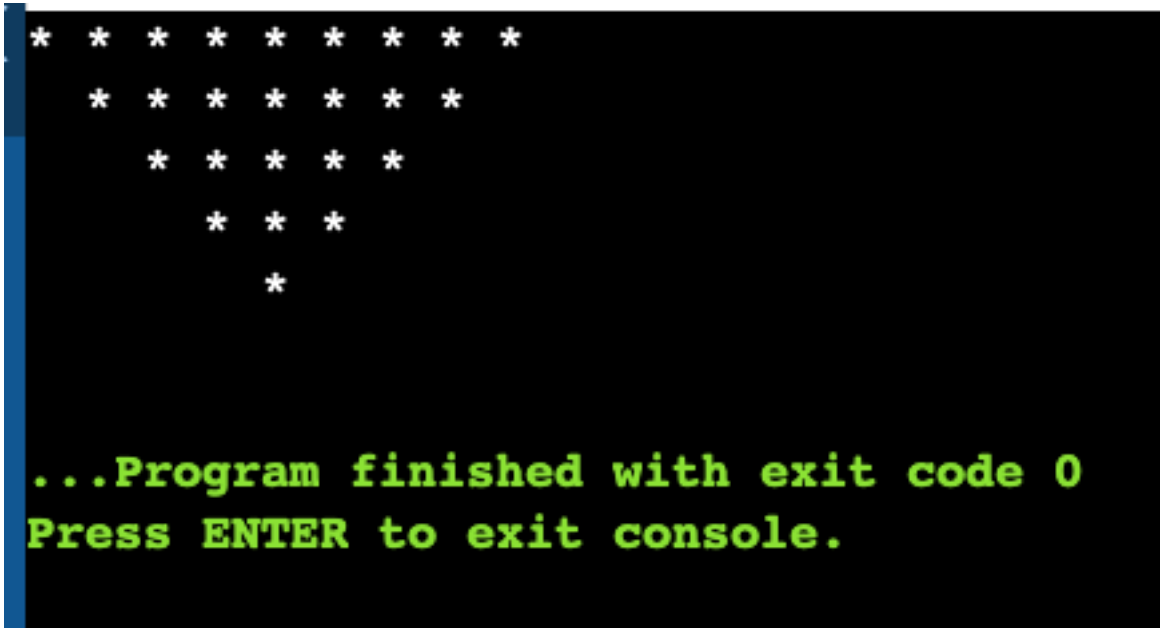
Function Basics

```
return_type function_name( parameter list ) {
   body of the function
}
```

- Return **Type**
- Function **Name**
- **Parameters**
- Function **Body**



Inverted pyramid of *, rows = 5



```c
#include <stdio.h>

void print_spaces(int r, int rows) {
    for (int i = r; i<rows; i++) {
        printf("  ");
    }
}

void print_stars(int r) {
    for (int i = 0; i < r*2-1; i++) {
        printf("* ");
    }
    printf("\n");
}

int main() {
    int rows = 5;
    for (int i = rows; i>=1; i--) {
        print_spaces(i, rows);
        print_stars(i);
    }
    return 0;
}
```

input

```
* * * * * * * * *
  * * * * * * *
    * * * * *
      * * *
        *

...Program finished with exit code 0
Press ENTER to exit console.
```

請按照規劃的兩個function一個main

- Full Pyramid of *
- int rows = 10;
- **main1.c**

```c
#include <stdio.h>

void print_spec(int i, int n) {
    for (int j=i; j<n; j++) {
        printf(" ");
    }
}

void peint_stars(int i) {
    for (int j=0; j<i*2+1; j++) {
        printf("*");
    }
}

int main() {
    int n = 5;
    for (int i = 0; i<=n; i++) {
        print_spec(i, n);
        peint_stars(i);
        printf("\n");
    }
}
```

```
    *
   ***
  *****
 *******
*********
***********


...Program finished with exit code 0
Press ENTER to exit console.
```

- 計算一個**正整數(int)**有幾個1
  - int i = 15;
- Binary representation
  - 0000 1111 (15) 有4個 1
  - 0000 0001 (1) 有1個 1
  - 1000 0001 (129) 有2個 1
- **main2.c**

```
4  int main() {
5      int n = 7;
6      int count = 0;
7      do {
8          if (n&1) {
9              count++;
10         }
11     } while ((n>>=1)!=0);
12     printf("count = %d\n", count);
13 }
```

- **get_binary(int n)**
- 請寫一個function 印出對應的**正整數(int, 32bits)** binary representation
  - Output: binary representation
  - 4個bits 為單位
  - 6 —> 0110
  - 16 —> 0001 0000
  - 255 —> 1111 1111
- **main3.c**

```c
#include <stdio.h>

int main() {
    int n = 19;
    int k;
    int count = 0;
    for (int c = 31; c >= 0; c--) {
        k = n >> c;

        if (k & 1) {
            printf("1");
        } else {
            printf("0");
        }

        if (c%4 == 0) {
            printf(" ");
        }
    }
}
```

```
0000 0000 0000 0000 0000 0000 0001 0011

...Program finished with exit code 0
Press ENTER to exit console.
```

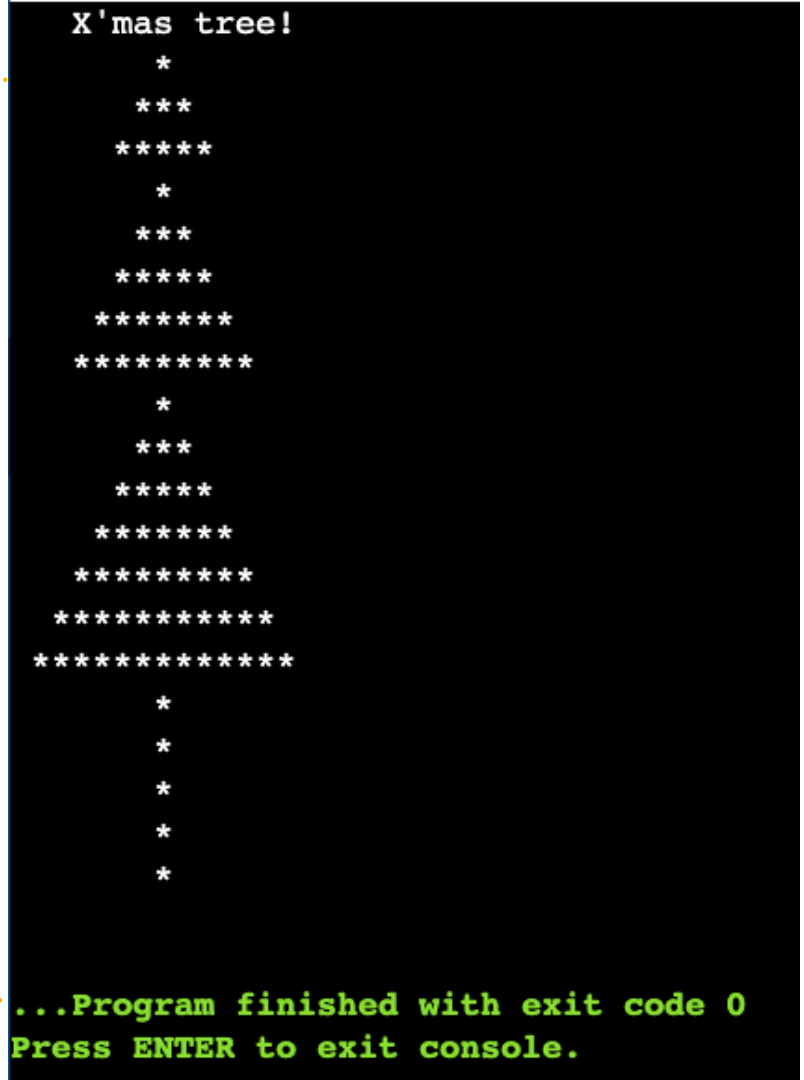
- **round_func(float f)**
- 實作四捨五入的函式, **不能用 LIB**
  - 1.4 -> 1
  - 2.6 —> 3
  - 2.34 —> 2
  - 9.2344567789 —> 9
- **main4.c**

```c
#include <stdio.h>

int round_func(float f) {
    return (f+=0.5);
}

int main() {
    printf("round_func = %d\n", round_func(1.4));
    printf("round_func = %d\n", round_func(2.6));
}
```

input

```
round_func = 1
round_func = 3


...Program finished with exit code 0
Press ENTER to exit console.
```

- 印一顆聖誕樹
  - print_top
  - print_truck
  - print_spaces
  - print_stars
- **main5.c**



- 給 n>2 的數字
  - 屋頂高 n
  - 房子高 n
- **main6.c**

```c
#include <stdio.h>

int main() {
    int n = 4;
```

```
   *
  ***
 *****
*******
*******
*     *
*     *
*     *
*     *
*******


...Program finished with exit code 0
Press ENTER to exit console.
```

- 把整數數字(int)的加起來！
  - 有負數
  - 只有整數
- **main7.c**

```c
int main() {
    int a = -999;
    int b = 999;
    // a= -9+9+9 = 9
    // b= 9+9+9 = 18
```

```c
#include <stdio.h>

int get_digit(int num) {
    return num%10;
}

int main() {
    int num = -9990;
    int sum = 0;
    int negative = 0;

    if (num < 0) {
        negative = 1;
        num = 0 - num;
    }

    do {
        if (num < 10) {
            if (negative) {
                sum -= get_digit(num);
            }
        }
        else {
            sum += get_digit(num);
        }
        num /= 10;
    } while (num>0);
    printf("sum=%d\n", sum);
    return 0;
}
```

- 兩個三角型
  - 一個倒三角
  - 一個正三角
- **main8.c**

```c
#include <stdio.h>

int main() {
    int i = 4;
```

```
*******
 *****
  ***
   *
  ***
 *****
*******


...Program finished with exit code 0
Press ENTER to exit console.
```

- 右邊的程式會停嗎？
  - 如果會，為什麼？
- **main9.c**
- **main9.pdf**

未給初始值

```
5  int main() {
6      int i, j;
7      for (i=0; i<100, j!=3; i++, j++) {
8          printf("i=%d\n", i);
9      }
10     return 0;
11 }
12
```

```
i=0
i=1
i=2
```

- 把**正整數(int)**倒轉
- Input **1234**
- Output: 4321
- **main10.c**

```c
#include <stdio.h>

int get_digit(int n) {
    return n%10;
}

int main() {
    int n = 1234;
    int sum = 0;

    while (n!=0) {
        sum = sum*10 + get_digit(n);
        n /= 10;
    }

    printf("sum = %d\n", sum);
    return 0;
}
```

```
sum = 4321


...Program finished with exit code 0
Press ENTER to exit console.
```

Thanks!

Open for any questions

CJ Wu

cjwu@mail.cgu.edu.tw