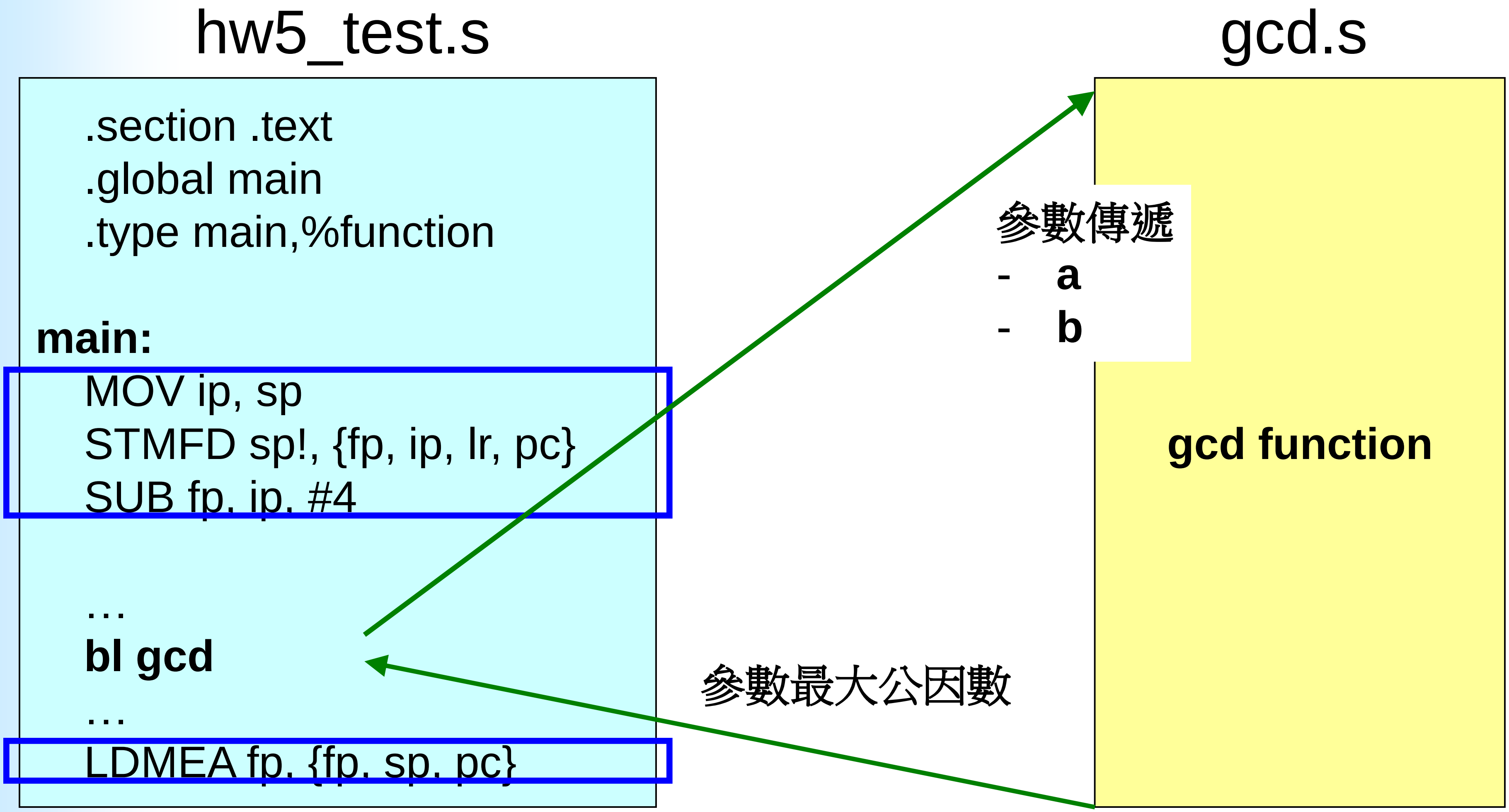
Homework #5 (1)

- Use ARM assembly to write a function called **gcd** that reads two integers and returns their greatest common divisor.

- Two arguments will be passed into your function by stack
  - **Two integers a, b (for example)**

- **The result of gcd (a, b)**
  - **The greatest common divisor of a and b**



hw5_test.s

```
    .section .text
    .global main
    .type main,%function

main:
    MOV ip, sp
    STMFD sp!, {fp, ip, lr, pc}
    SUB fp, ip, #4

    …
    bl gcd
    …
    LDMEA fp, {fp, sp, pc}
```

gcd.s

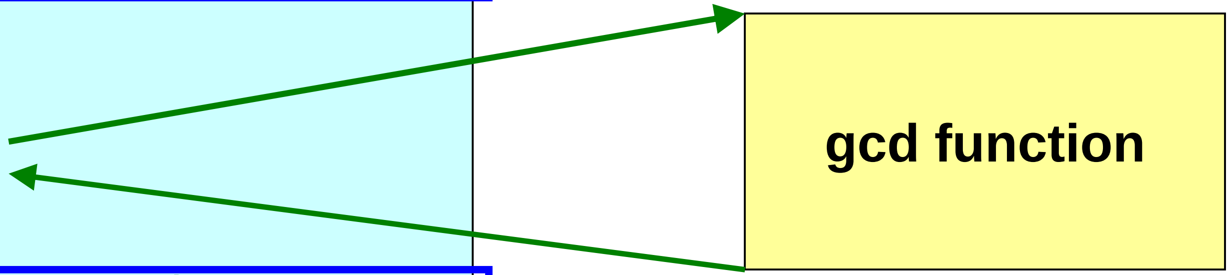
Homework #5 (2)

```
    .section .text
    .global main
    .type main,%function

main:
    MOV ip, sp
    STMFD sp!, {fp, ip, lr, pc}
    SUB fp, ip, #4

    …
    bl gcd
    …
    LDMEA fp, {fp, sp, pc}
```

A ARM assembly program which uses your procedure demos your gcd function.



gcd function

hw5_test.s

```
 .section .text
        .global main
        .type main,%function
main:
        MOV ip, sp
        STMFD sp!, {fp, ip, lr, pc}
        SUB fp, ip, #4

        /* --- begin of your function --- */
        MOV r0, #100     /* set argument value */
        MOV r1, #200     /* set argument value */

        bl  gcd
        nop
        /* --- end of your function --- */

        LDMEA fp, {fp, sp, pc}
        .end
```

傳回值最大公因數存放在**r0**

Homework #5 (3)

gcd.s

```
        .section .text
        .global gcd
        .type gcd,%function

gcd:
        /* function start */
        MOV ip, sp
        STMFD sp!, {r4-r11, fp, ip, lr, pc}
        SUB fp, ip, #4

        /* --- begin your function --- */
        /* 傳入值會放在r0, r1 */

        /* DO gcd */

        /* 把傳回值 (最大公因數) 放在r0 */
        /* --- end of your function --- */
        /* function exit */
        LDMEA fp, {r4-r11, fp, sp, pc}
        .end
```

請留意callee saved registers

參數傳遞

Write your function

How to Compile Your Program?

```
$ arm-non-eabi-gcc -g -O0 hw5_test.s gcd.s -o hw5.exe
```

Homework #5 (4)

- Program should be assembled and linked by gcc
  - 使用於作業一所安裝完成的cross compiler與cross binutils
- Program should be executed under **GDB ARM simulator**
- 程式中應有適當的說明（註解）
- You should turn in to **ECOURSE**
  - “**README.txt**” file: 文字檔，描述你程式的內容、如何編譯程式、如何執行你的程式
  - Your ARM assembly procedure，**檔名為：gcd.s**
  - An ARM assembly program which uses your gcd function，**檔名為：hw5_test.s**
  - Makefile
  - Any file needed in your work
- **Deadline: December 7(Friday), 2018**