資工 113 許証淵 40947071S

```
1  #include<stdio.h>
2  #include<stdint.h>
3  int32_t p(int32_t, int32_t);
4  int main()
5  {
6      int32_t a=0, b=0;
7      printf("Please enter a and b: ¥n");
8      scanf("%d %d", &a, &b);
9      p(a, b);
10     return 0;
11 }
12 int32_t p(int i, int N)
13 {
14     return (i < N && printf("%d ¥n", i) && !p(i+1, N))
15             || printf("%d ¥n", i);
16 }
```

在原先ｃｏｄｅ裡第９行有多放一行

Printf (“Result: %d \n”, p(a , b));

接著執行以下兩種測資：

(一) ａ ＞ ｂ ：印出 a 至 b 的值接著是 1(布林值裡的 true)

由於ｉ＞Ｎ（ａ＞ｂ）

在 14 行的部分會先判斷出 i > N，不符合條件，因此後面的&&部分都不會執行，直接跳到||printf(“%d \n”, i) ;

接著只要有印出 i 值，printf 會回傳一個 return value，因此會變成 0 || 1 = 1。

(二) ｂ＞ａ ：印出 a 至 b 的所有值，在印出 b 至 a 的值 接著是 1 (布林值裡的 true)

由於ｉ＜Ｎ（ａ＜ｂ）

i < N 與 printf(“%d”,n) 皆會回傳 return value 1，接著執行到 ! p(i+1, N)時，程式會先呼叫裡面的函式，因此將會重複「i < N 回傳 1，printf 輸出 1 並回傳 1」的程式，直到 i>=N 時(也就是(一)的情況)最終回傳 1，接著原本的 p 函式便會收到 !1=0，使整個 AND 判斷式回傳 0(false)。

最後的判斷➔0||1=1