

hw0306

作用

setjmp和**longjmp**函數可以讓程式往回跳到函式呼叫堆疊的某個函數中，就像是一種跨函數的goto。

int setjmp(jmp_buf env)

void longjmp(jmp_buf env, int value) value的值從longjmp傳遞給setjmp。

setjmp儲存當前程式狀態到jmp_buf這個變數，其會在隨後程式執行的某處被longjmp用於恢復程式的狀態到setjmp呼叫所儲存到jmp_buf時的原樣。

實作

要記得 **#include <setjmp.h>**

setjmp可以在程式中標示一個目標，這個目標就是跳躍的目的地，在要跳躍的地方呼叫**longjmp**就可以跳過去了。

```
1  #include <stdio.h>
2  #include <setjmp.h>
3  j
4  mp_buf jmpbuffer;
5
6  int fun_a(int v){
7      int a=v+1;
8      if(a>10){
9          longjum(jmpbuffer,1);
10     }
11     return r;
12 }
13
14 int fun_b(int v) {
15     int r=fun_a(v)+1;
16     if(r>10) {
17         longjmp(jmpbuffer, 2);
18     }
19     return r;
20 }
21
22 int main(){
23     int jmp_val=setjmp(jmpbuffer);
24     if( jmp_val==1){
25         printf(" <10.\n ");
26     }else if( jmp_val==1 ){
27         printf(" >10.\n ");
28     }
29     printf("Great nothing~\n");
30     return 0;
31 }
```

重點是，這兩個函式能幫助 code 做深層函式錯誤處理。

