<https://blog.csdn.net/qunxuan/article/details/40887643>

在使用c语言写程序时，可能遇到错误：error ：**dereferencing pointer to incomplete type**。其实，这个错误是指针指向的结构体类型没有定义。

原因可能有很多，但最多情况可能下面两种：

1，使用库函数或内核等提供的结构体时，没有包含相应的头文件。解决方法很简单，就是包含对应头文件就ok了。

2，若是自己定义的结构体，而且这个结构体恰恰是定义在.c文件中。在这种情况下，在其他.c文件中使用该结构体指针时，也会出现类似错误。

就第二种情况，笔者找到两种解决方法：

1，将结构体定义放到.h文件中，然后在使用该结构体的.c文件中分别包含这个文件。这种方法较符合常规。

在这里需要注意，在定义头文件时，要使用预编译语句将头文件包围起来。这样做是防止出现头文件被重复包含的错误。

例如：

#ifndef  \_ABC\_H

#define \_ABC\_H

结构体定义

#endif

2，将结构体定义从上一个.c文件中拷至出错的.c文件中，因为结构体定义只是声明了结构体的大小并不真实分配空间，所以这样做不会有问题。

示例代码：

**a.c**

#include <stdio.h>  
  
struct obj\_t {  
int x;  
int y;  
};  
  
struct obj\_t obj;  
  
struct obj\_t \* get\_obj()  
{  
obj.x = 11;  
obj.y = 22;  
  
return &obj;  
}  
  
  
void show(struct obj\_t \*p)  
{  
printf("a=%d\nb=%d\n", p->x, p->y, \_\_func\_\_);  
}

**b.c**

void call\_show()

{  
struct obj\_t \*p = get\_obj();  
  
p->x = 111;  
p->y = 222;  
  
show(p);  
}

**main.c**

int main(int argc, char \*\*argv)  
{  
call\_show();  
}

执行命令：gcc a.c b.c main.c -o test

出现如下错误：

b.c: In function ‘call\_show’:  
b.c:10: warning: initialization makes **pointer** from integer without a cast  
**b.c:12: error: dereferencing pointer to incomplete type  
b.c:13: error: dereferencing pointer to incomplete tyd;"**

如下修改b.c后，错误消失

struct obj\_t {  
int x;  
int y;  
};  
  
void call\_show()  
{  
struct obj\_t \*p = get\_obj();  
  
p->x = 111;  
p->y = 222;  
  
show(p);  
}