

作業 #10

■#練習：請定義 twoDPoint、threeDPoint 類別，能夠計算出二維和三維空間中兩點之間的距離。

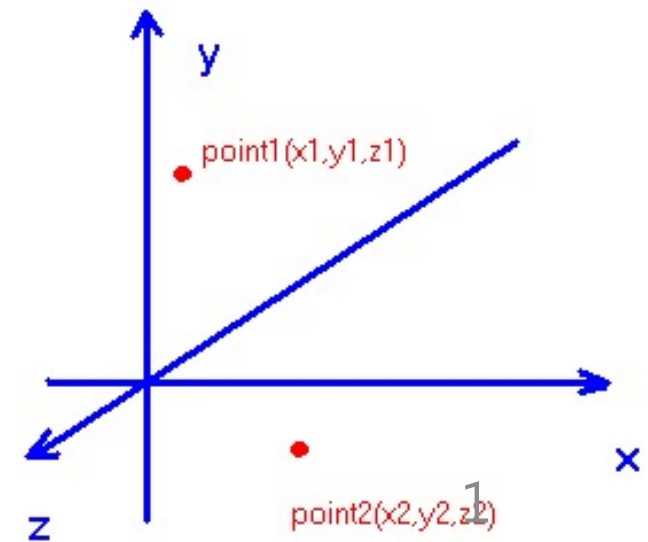
■Requirements：

1. 定義一個 twoDPoint 類別代表二維空間中的點（包含 x 和 y 座標）
2. 定義由 twoDPoint 繼承的 threeDPoint 類別代表三維空間中的點（z 座標）
3. 兩個類別中分別利用「-」運算子多載成計算兩點的距離

■Sample Input：參考下頁

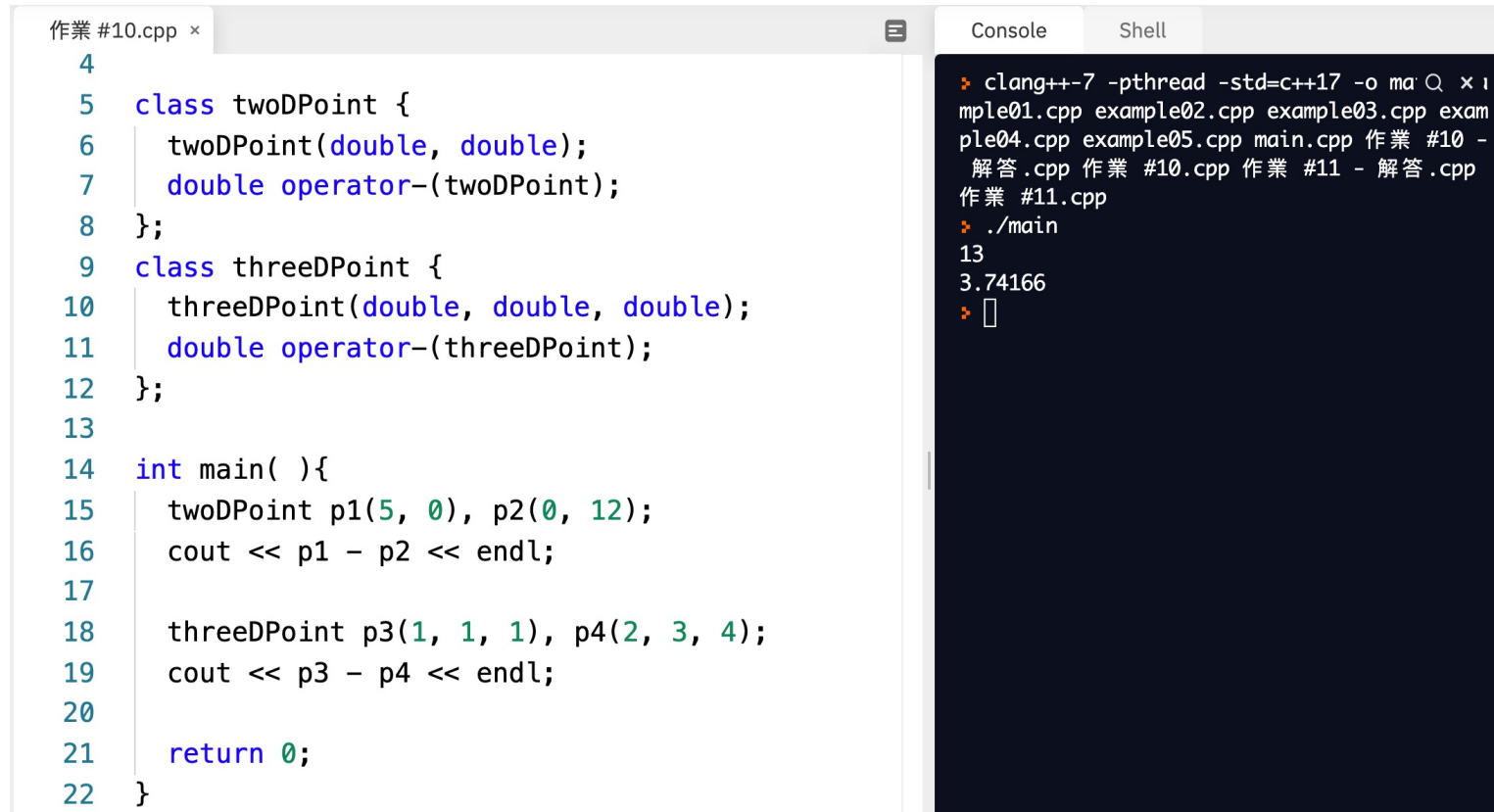
■Sample Output：參考下頁

■Note：僅限 04/27 上課繳交



參考程式碼與結果

■你可以參考下列程式碼修改，也可以自己從頭開始寫。只要執行結果符合題目要求即可。



The screenshot shows a C++ IDE with two panels. The left panel displays the source code for '作業 #10.cpp', and the right panel shows the console output.

```
作業 #10.cpp ×
4
5 class twoDPoint {
6     twoDPoint(double, double);
7     double operator-(twoDPoint);
8 };
9 class threeDPoint {
10    threeDPoint(double, double, double);
11    double operator-(threeDPoint);
12 };
13
14 int main( ){
15     twoDPoint p1(5, 0), p2(0, 12);
16     cout << p1 - p2 << endl;
17
18     threeDPoint p3(1, 1, 1), p4(2, 3, 4);
19     cout << p3 - p4 << endl;
20
21     return 0;
22 }
```

Console

```
> clang++-7 -pthread -std=c++17 -o main example01.cpp example02.cpp example03.cpp example04.cpp example05.cpp main.cpp 作業 #10 - 解答.cpp 作業 #10.cpp 作業 #11 - 解答.cpp 作業 #11.cpp
> ./main
13
3.74166
>
```

作業繳交說明

你需要繳交以下檔案到 **Portal** 作業：

- 1. cpp 程式碼
- 2. 程式碼內有文字說明的註解
- 3. 執行結果截圖

(若無法上傳多個檔案，請壓縮成 zip 或 rar 格式，
並且命名成「學號.zip」或「學號.rar」)

■ 上課驗收者只需上傳程式碼即可

■ 本次作業截止時間：

僅限 04/27 上課繳交

```
main.cpp
1 // 學號： 1091000 姓名：王大明
2 /* 文字說明：這個作業當中，我利用一個變數 x 存放名字並且利用
   cout 做印出。 */
3
4 #include <iostream>
5 #include <string>
6 using namespace std;
7
8 int main() {
9     string x = "王大明";
10    std::cout << "Hello World, " << x << endl;
11 }
```

```
> clang++-7 -pthread -std=c++17 -o main main.cpp
> ./main
Hello World, 王大明
> 
```