

C++

程式語言（二）實習課

Introduction to Programming (II)

Practice Lessons

Structures: Rectangles

Joseph Chuang-Chieh Lin

Dept. CSE, NTOU

Platform/IDE

- Dev-C++



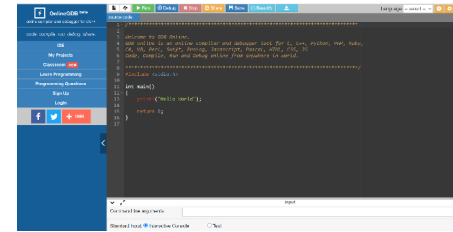
<https://www.pngegg.com/en/search?q=Dev-C>

- Codeblocks

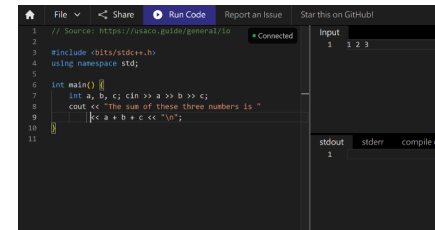


<https://icons8.com/icons/set/code-blocks>

- OnlineGDB (<https://www.onlinegdb.com/>)



- Real-Time Collaborative Online IDE (<https://ide.usaco.guide/>)



Useful Resources

- Tutorialspoint
 - <https://www.tutorialspoint.com/cplusplus/index.htm>
 - Online C++ Compiler
- Programiz
 - <https://www.programiz.com/cpp-programming>
- LEARN C++
 - <https://www.learncpp.com/>
- MIT OpenCourseWare - Introduction to C++
 - <https://ocw.mit.edu/courses/6-096-introduction-to-c-january-iap-2011/pages/lecture-notes/>
- Learning C++ Programming
 - <https://www.programiz.com/cpp-programming>
- GeeksforGeeks
 - <https://www.geeksforgeeks.org/c-plus-plus/>

Course TAs

- 薛凱駿
 - Email address: 11357063@o365.ntou.edu.tw
- 楊承霖
 - Email address: 11357072@o365.ntou.edu.tw
- 吳奇軒
 - Email address: 11357052@o365.ntou.edu.tw

Exercise 01: Resolving the naming conflict

- Suppose that we have
 - `main.cpp`
 - Files for NTOU:
 - `ntou.h`
 - `ntou.cpp`
 - Files for CMU
 - `cmu.h`
 - `cmu.cpp`

[Codes on OnlineGDB](#)

Class assignment:
Modify the code (line 19) to have
the following output:

```
Betty is playing in CMU!  
Betty is studying hard in CMU!
```

Exercise 01

- Using `struct` to compute the area of the input axis-parallel rectangle

```
struct Rect {  
    double leftUpCorner_x;  
    double leftUpCorner_y;  
    double rightDownCorner_x;  
    double rightDownCorner_y;  
};
```

leftUpCorner_x, leftUpCorner_y,



rightDownCorner_x, rightDownCorner_y

SAMPLE INPUT:

10 20 30 -10

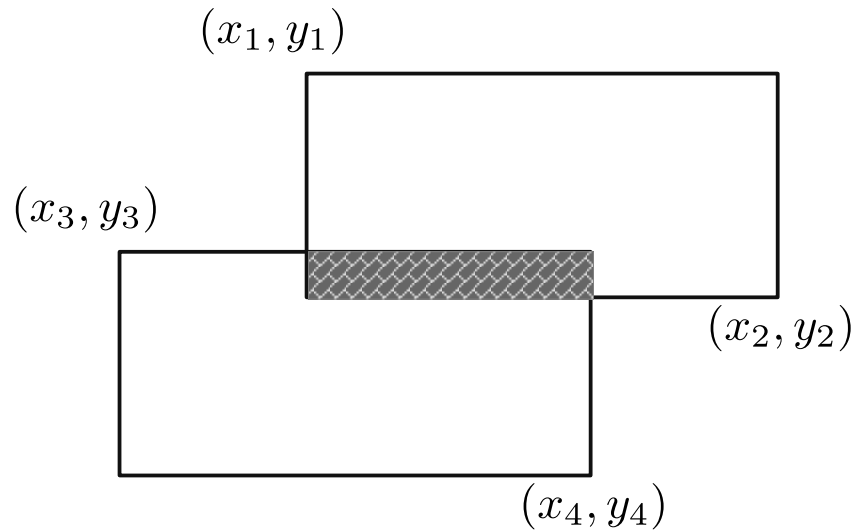
SAMPLE OUTPUT:

600

Exercise 02

- Compute the area of **union** of two axis-parallel rectangles

Input: $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$



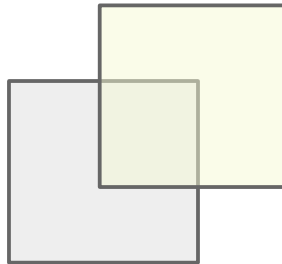
*Rakuten Data Scientist Position Online Codility Test

Exercise 02 (contd.)

```
struct Rect {  
    double leftUpCorner_x;  
    double leftUpCorner_y;  
    double rightDownCorner_x;  
    double rightDownCorner_y;  
};
```

```
int main() {  
    Problem p;  
    ...//read input  
    p.solve();  
}
```

```
struct Problem {  
    Rect r1, r2;  
    void solve() {  
        ... /* compute the area of the  
            union of r1 and r2 */  
    }  
};
```



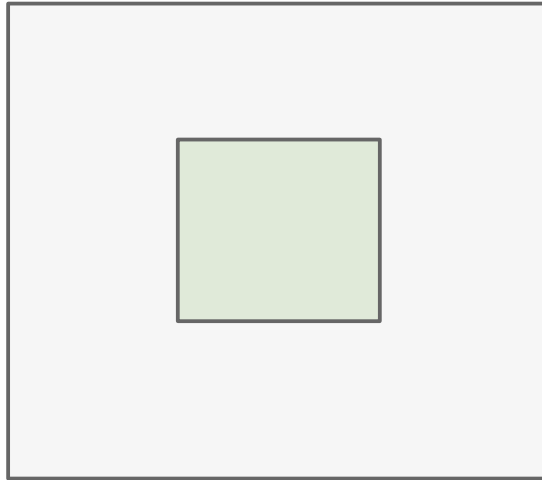
SAMPLE INPUT:

0 2 2 0 1 3 3 1

SAMPLE OUTPUT:

7

Exercise 02 (contd.)



SAMPLE INPUT:

1 4 5 1 2 3 3 2

SAMPLE OUTPUT:

12