

The 2014 ACM-ICPC

Asia Tokyo Regional Contest

国際大学対抗プログラミングコンテスト

International Collegiate Programming Contest



<http://icpc.iisf.or.jp/2014-waseda/>

申し込み締切： 国内予選（インターネット）： アジア地区予選（東京大会）：

06/27 ▶ 07/11 ▶ 10/18~20

会場：国立オリンピック記念青少年総合センター（東京都渋谷区）

問い合わせ窓口

Sponsored by

Mail: icpc2014@list.waseda.jp
Twitter: @icpc_tokyo2014

Waseda University - 早稲田大学
IISF - 公益財団法人 情報科学国際交流財団



例題

(2013 年 国内予選 問題 A)

整長方形

高さ h と幅 w がともに正の整数である長方形を考えよう。そのような長方形を整長方形と呼ぶ。以下、この問題では横長の整長方形、すなわち $w > h$ である整長方形のみを考える。

横長整長方形の大小関係を次のように定めよう。

- 対角線の長さが短いほうが小さい。
 - 対角線の長さが同じならば、高さの低いほうが小さい。
- 与えられた横長整長方形より大きい最小の横長整長方形を求めよ。

$h = 5, w = 6$



対角 = $\sqrt{61}$

$h = 1, w = 8$



対角 = $\sqrt{65}$

$h = 4, w = 7$



対角 = $\sqrt{65}$



①Input

入力は複数行（100 行以下）。各行は空白一つで区切られた高さ h と幅 w 。入力の終わりは h と w が共に 0 の行。

②Output

解となる長方形の高さ h と幅 w を空白ひとつで区切って 1 行ずつ出力せよ。出力にはこれら以外の文字があってはならない。

解となる横長整長方形の高さと幅は共に 150 を超えないとわかっている。

①Sample Input

```
1 2
1 3
4 7
98 100
99 100
0 0
```

②Sample Output

```
1 3
2 8
3 140
89 109
```

```
#include <algorithm>
#include <iostream>
using namespace std;
int H, W;
```

```
// 長方形の対角線の二乗
int tkj(int h, int w) {
    return(h*h + w*w);
}
```

```
int main() {
    while(cin >> H >> W && H > 0) {
        const pair<int, int> given(tkj(H, W), H);
        pair<int, int> best(tkj(150, 150), 150), ans;
        for(int h=1; h<=150; ++h) {
            for(int w=1; w<=150; ++w) {
                if(w<=h) continue;
                // 幅の方が長い長方形を扱う
                pair<int, int> x(tkj(h, w), h);
                // pair<int, int> による比較
                if(given < x && x < best) {
                    best=x;
                    ans= make_pair(h, w);
                }
            }
        }
        cout << ans.first << ' ' << ans.second << endl;
    }
}
```

C++ コードによる解答例

`pair<int, int>` の利用で長方形の比較を簡潔に実現。
(`pair1 < pair2`) では最初の要素で大小比較。
それが同じ大きさなら二番目の要素で大小比較。
`given` や `x`, `best` では `pair` (対角線二乗, 高さ) となっている。

・他にも解き方はあるかな？
・もっと速くできるかな？
CやJavaでも解いてみよう！