CSES Problem Set

Rectangle Cutting

Submission details

Task:	Rectangle Cutting
Sender:	Rodry
Submission time:	2021-11-28 07:04:57
Language:	C++11
Status:	READY
Result:	ACCEPTED

Test results

test	verdict	time	
#1	ACCEPTED	0.01 s	»
#2	ACCEPTED	0.01 s	»
#3	ACCEPTED	0.01 s	»
#4	ACCEPTED	0.01 s	»
#5	ACCEPTED	0.01 s	»
#6	ACCEPTED	0.08 s	»
#7	ACCEPTED	0.05 s	»
#8	ACCEPTED	0.01 s	»
#9	ACCEPTED	0.04 s	»
#10	ACCEPTED	0.01 s	»
#11	ACCEPTED	0.03 s	»
#12	ACCEPTED	0.02 s	»
#13	ACCEPTED	0.08 s	»
#14	ACCEPTED	0.01 s	»

Dynamic Programming

...

Array Description –

Counting Towers –

Edit Distance –

Rectangle Cutting ✓

Money Sums –

Removal Game –

Two Sets II –

Increasing Subsequence –

...

Your submissions

2021-11-28 06:52:21 ✓

test	verdict	time	
#15	ACCEPTED	0.01 s	»
#16	ACCEPTED	0.06 s	»
#17	ACCEPTED	0.01 s	»
#18	ACCEPTED	0.03 s	»
#19	ACCEPTED	0.01 s	»
#20	ACCEPTED	0.22 s	»
#21	ACCEPTED	0.22 s	»
#22	ACCEPTED	0.01 s	»
#23	ACCEPTED	0.01 s	»
#24	ACCEPTED	0.06 s	»
#25	ACCEPTED	0.01 s	»
#26	ACCEPTED	0.23 s	»

Code ▲

```
#include <bits/stdc++.h>

#define INF 999999

using namespace std;

int RectangleCutting(int a, int b){

        int rectangulo[a+1][b+1];

        for(int i=0; i<=a; i++){
                for (int j=0; j<=b; j++) {

                        //Diagonalizaci?n
                        if (i == j) {
                                rectangulo[i][j] = 0;
                        }

                else{

                        int k=1;
                        rectangulo[i][j] = INF;

                        for(k = 1; k<j; k++){
```

```
                                rectangulo[i][j] = min(rectangulo[i][j], rectangu
                                }

                        for(k = 1; k<i; k++){
                                rectangulo[i][j] = min(rectangulo[i][j], rectangu
                                }

                    }
                }
            }
            return rectangulo[a][b];
}


int main() {

            int a;
            int b;
            cin>>a>>b;
            cout<<RectangleCutting(a,b);

}
```

Share code to others

Test details ▲

Test 1

Verdict: ACCEPTED

input
2 8

correct output
3

user output
3

Test 2

Verdict: ACCEPTED

input
4 4

correct output
0

user output
0

Test 3

Verdict: ACCEPTED

input
1 4

correct output
3

user output
3

Test 4

Verdict: ACCEPTED

input
5 8

correct output
4

user output
4

Test 5

Verdict: ACCEPTED

input
5 10

correct output
1

user output
1

Test 6

Verdict: ACCEPTED

input
404 288

correct output
10

user output
10

Test 7

Verdict: ACCEPTED

input
349 234

correct output
13

user output
13

Test 8

Verdict: ACCEPTED

input
180 137

correct output
12

user output
12

Test 9

Verdict: ACCEPTED

input
201 348

correct output
10

user output
10

Test 10

Verdict: ACCEPTED

input
127 36

correct output
11

user output
11

Test 11

Verdict: ACCEPTED

input
159 399

correct output
10

user output
10

Test 12

Verdict: ACCEPTED

input
136 284

correct output
9

user output
9

Test 13

Verdict: ACCEPTED

input
415 270

correct output
10

user output
10

Test 14

Verdict: ACCEPTED

input
34 162

correct output
11

user output
11

Test 15

Verdict: ACCEPTED

input
81 229

correct output
13

user output
13

Test 16

Verdict: ACCEPTED

input
297 336

correct output
11

user output
11

Test 17

Verdict: ACCEPTED

input
132 46

correct output
9

user output
9

Test 18

Verdict: ACCEPTED

input
425 94

correct output
14

user output
14

Test 19

Verdict: ACCEPTED

input
216 114

correct output
10

user output
10

Test 20

Verdict: ACCEPTED

input
499 500

correct output
15

user output
15

Test 21

Verdict: ACCEPTED

input
500 499

correct output
15

user output
15

Test 22

Verdict: ACCEPTED

input
1 500

correct output
499

user output
499

Test 23

Verdict: ACCEPTED

input
1 1

correct output
0

user output
0

Test 24

Verdict: ACCEPTED

input
425 225

correct output
7

user output
7

Test 25

Verdict: ACCEPTED

input
218 91

correct output
11

user output
11

Test 26

Verdict: ACCEPTED

input
500 500

correct output
0

user output
0