

```

17         dp[i][j]=0;
18         continue;
19     }
20
21     int min_vertical = i*j+1;
22     for (int k=1; k<=j; k++){
23         int min_vertical_new = dp[i][k] + dp[i][j-k];
24         if(min_vertical_new < min_vertical)
25             min_vertical = min_vertical_new;
26     }
27
28     int min_horizontal = i*j;
29     for (int k=1; k<=i; k++){
30         int min_horizontal_new = dp[k][j] + dp[i-k][j];
31         if(min_horizontal_new < min_horizontal)
32             min_horizontal = min_horizontal_new;
33     }
34
35     dp[i][j] = std::min(i*j, min_horizontal);
36     dp[i][j] = std::min(dp[i][j], min_vertical);
37 }
38 }
39
40 return dp[X][Y];
41 }

```

test.cpp

```

1 #include <iostream>
2 #include <fstream>
3 #include "the4.h"
4
5
6 void read_from_file(int& X, int& Y, bool** possible_plots){
7     int number_of_plots;
8     char addr[] = "inp03.txt"; // 01-10 are available
9     std::ifstream infile (addr);
10    if (!infile.is_open()){
11        std::cout << "File \"\"<< addr
12        << \"\ can not be opened. Make sure that this file exists.\" << std::endl;
13        return;
14    }
15    infile >> X;
16    infile >> Y;
17    infile >> number_of_plots;
18    possible_plots = new bool*[X+1];
19    for(int temp=0; temp < X+1; temp++) possible_plots[temp] = new bool[Y+1];
20    for(int idx=0; idx < X+1; idx++) for(int idy=0; idy < Y+1; idy++) possible_plots[idx][idy] = false;
21    for(int temp=0; temp < number_of_plots; temp++){
22        std::pair<int, int> plot;
23        infile >> plot.first >> plot.second;
24        possible_plots[plot.first][plot.second] = true;
25    }
26    infile.close();
27 }
28
29 int main(){
30     int X, Y;
31     bool** input_array;
32     int minimum_unused_land, plot_number=1;
33
34     read_from_file(X, Y, input_array);
35
36     std::cout << "X: " << X << ", Y: " << Y << std::endl;
37     for(int idx=0; idx < X+1; idx++)
38         for(int idy=0; idy < Y+1; idy++)
39             if(input_array[idx][idy])
40                 std::cout << "Plot " << plot_number++ << ": (" << idx << ", " << idy << ")" << std::endl;
41 }

```

Description

Submission

view

- > November 27 - December 3
- > December 3 - December 10
- > December 11 - December 17
- > December 18 - December 24
- > December 25 - December 31
- > January 1 - January 7
- > January 8 - January 14
- > CENG 315 Section 1

[-]For input 13:
m^2 of unused land: correct

[-]For input 14:
m^2 of unused land: correct

[-]For input 15:
m^2 of unused land: correct

[-]For input 16:
m^2 of unused land: correct

[-]For input 17:
m^2 of unused land: correct

[-]For input 18:
m^2 of unused land: correct

[-]For input 19:
m^2 of unused land: correct

[-]For input 20:
m^2 of unused land: correct

Submitted on Saturday, November 25, 2023, 10:01 PM (Download)

the4.cpp

```

1 #include "the4.h"
2
3 // do not add extra libraries here
4
5 int divide_land(int X, int Y, bool** possible_plots){
6
7     int** dp = new int*[X+1];
8     for(int i = 0; i <= X; i++){
9         dp[i] = new int [Y+1];
10        for (int j = 0; j <= Y; j++){
11            dp[i][j] = X*Y;
12        }
13    }
14    for(int i=1; i<=X; i++){
15        for(int j=1; j<=Y; j++){
16            if(possible_plots[i][j]){
17                dp[i][j]=0;
18                continue;
19            }
20
21            int min_vertical = i*j+1;
22            for (int k=1; k<=j; k++){
23                int min_vertical_new = dp[i][k] + dp[i][j-k];
24                if(min_vertical_new < min_vertical)
25                    min_vertical = min_vertical_new;
26            }
27
28            int min_horizontal = i*j;
29            for (int k=1; k<=i; k++){
30                int min_horizontal_new = dp[k][j] + dp[i-k][j];
31                if(min_horizontal_new < min_horizontal)
32                    min_horizontal = min_horizontal_new;
33            }
34
35            dp[i][j] = std::min(i*j, min_horizontal);
36            dp[i][j] = std::min(dp[i][j], min_vertical);
37        }
38    }
39
40    return dp[X][Y];
41 }

```

```

28     int min_horizontal = 1*j;
29     for (int k=1; k<=i; k++){
30         int min_horizontal_new = dp[k][j] + dp[i-k][j];
31         if(min_horizontal_new < min_horizontal)
32             min_horizontal = min_horizontal_new;
33     }
34     dp[i][j] = std::min(i*j, min_horizontal);
35     dp[i][j] = std::min(dp[i][j], min_vertical);
36 }
37 }
38 return dp[X][Y];
39 }
40 }
41

```

test.cpp

```

1  #include <iostream>
2  #include <fstream>
3  #include "the4.h"
4
5
6  void read_from_file(int& X, int& Y, bool**& possible_plots){
7      int number_of_plots;
8      char addr[] = "inp03.txt"; // 01-10 are available
9      std::ifstream infile (addr);
10     if (!infile.is_open()){
11         std::cout << "File \"<\" addr
12         << \"\ ' can not be opened. Make sure that this file exists.\" << std::endl;
13         return;
14     }
15     infile >> X;
16     infile >> Y;
17     infile >> number_of_plots;
18     possible_plots = new bool*[X+1];
19     for(int temp=0; temp < X+1; temp++) possible_plots[temp] = new bool[Y+1];
20     for(int idx=0; idx < X+1; idx++) for(int idy=0; idy < Y+1; idy++) possible_plots[idx][idy] = false;
21     for(int temp=0; temp < number_of_plots; temp++){
22         std::pair<int, int> plot;
23         infile >> plot.first >> plot.second;
24         possible_plots[plot.first][plot.second] = true;
25     }
26     infile.close();
27 }
28
29 int main(){
30     int X, Y;
31     bool** input_array;
32     int minimum_unused_land, plot_number=1;
33
34     read_from_file(X, Y, input_array);
35
36     std::cout << "X: " << X << ", Y: " << Y << std::endl;
37     for(int idx=0; idx < X+1; idx++){
38         for(int idy=0; idy < Y+1; idy++){
39             if(input_array[idx][idy])
40                 std::cout << "Plot " << plot_number++ << ": (" << idx << ", " << idy << ")" << std::endl;
41         }
42     }
43     minimum_unused_land = divide_land(X, Y, input_array);
44
45     std::cout << "Unused land: " << minimum_unused_land << " m^2" << std::endl;
46     for(int idx=0; idx<X+1; idx++) delete[] input_array[idx];
47     delete[] input_array;
48     return 0;
49 }

```

the4.h

```

1  #ifndef THE4_THE4_H
2  #define THE4_THE4_H
3  #include <vector>
4  #include <utility>
5  #include <algorithm>
6  #include <climits>
7
8  //updating this file will not change the execution in the VPL
9
10 int divide_land(int X, int Y, bool** possible_plots);
11
12 #endif //THE4_THE4_H

```

VPL

You are logged in as [mustafa baris emektar](#) (Log out)

[CENG 315 ALL Sections](#)

[Get the mobile app](#)

