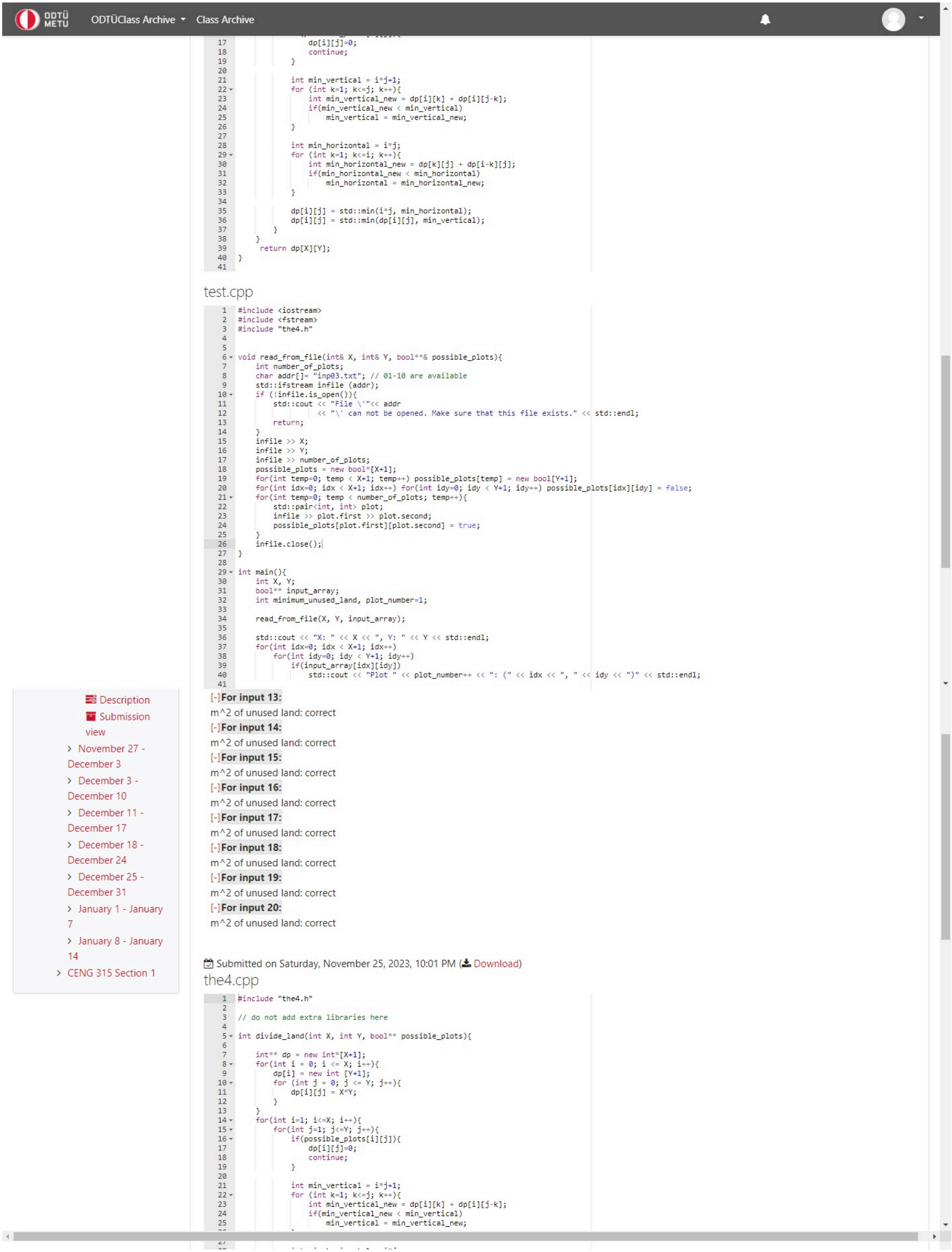


```
                dp[i][j]=0;
                continue;
            }

            int min_vertical = i*j+1;
            for (int k=1; k<=j; k++){
                int min_vertical_new = dp[i][k] + dp[i][j-k];
                if(min_vertical_new < min_vertical)
                    min_vertical = min_vertical_new;
            }

            int min_horizontal = i*j;
            for (int k=1; k<=i; k++){
                int min_horizontal_new = dp[k][j] + dp[i-k][j];
                if(min_horizontal_new < min_horizontal)
                    min_horizontal = min_horizontal_new;
            }

            dp[i][j] = std::min(i*j, min_horizontal);
            dp[i][j] = std::min(dp[i][j], min_vertical);
        }
    }
     return dp[X][Y];
}

```

test.cpp

```
#include <iostream>
#include <fstream>
#include "the4.h"


void read_from_file(int& X, int& Y, bool**& possible_plots){
    int number_of_plots;
    char addr[]= "inp03.txt"; // 01-10 are available
    std::ifstream infile (addr);
    if (!infile.is_open()){
        std::cout << "File \'"<< addr
                  << "\' can not be opened. Make sure that this file exists." << std::endl;
        return;
    }
    infile >> X;
    infile >> Y;
    infile >> number_of_plots;
    possible_plots = new bool*[X+1];
    for(int temp=0; temp < X+1; temp++) possible_plots[temp] = new bool[Y+1];
    for(int idx=0; idx < X+1; idx++) for(int idy=0; idy < Y+1; idy++) possible_plots[idx][idy] = false;
    for(int temp=0; temp < number_of_plots; temp++){
        std::pair<int, int> plot;
        infile >> plot.first >> plot.second;
        possible_plots[plot.first][plot.second] = true;
    }
    infile.close();
}

int main(){
    int X, Y;
    bool** input_array;
    int minimum_unused_land, plot_number=1;

    read_from_file(X, Y, input_array);

    std::cout << "X: " << X << ", Y: " << Y << std::endl;
    for(int idx=0; idx < X+1; idx++)
        for(int idy=0; idy < Y+1; idy++)
            if(input_array[idx][idy])
                std::cout << "Plot " << plot_number++ << ": (" << idx << ", " << idy << ")" << std::endl;

```

[-]**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

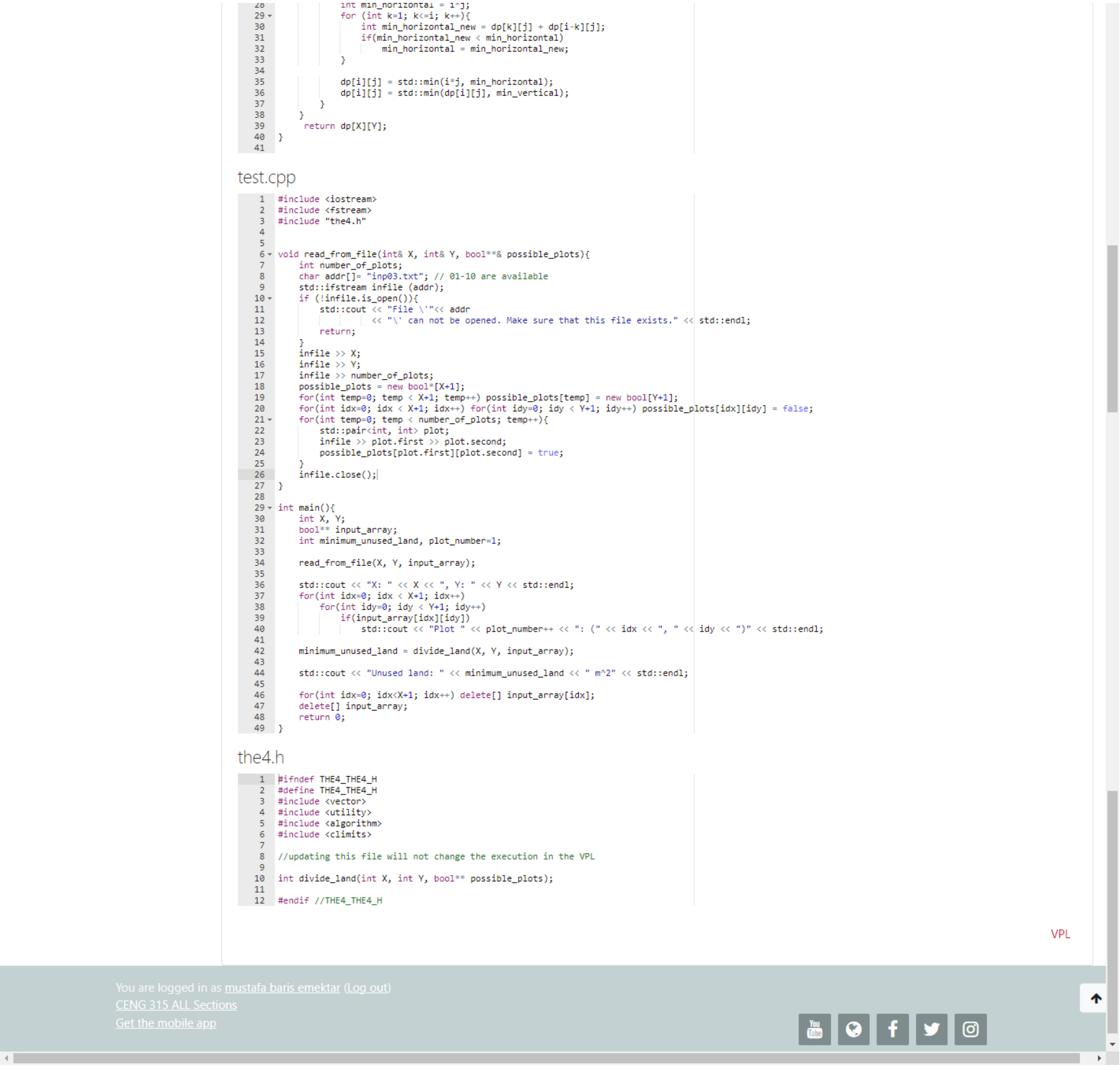
```
#include "the4.h"

// do not add extra libraries here

int divide_land(int X, int Y, bool** possible_plots){

    int** dp = new int*[X+1];
    for(int i = 0; i <= X; i++){
        dp[i] = new int [Y+1];
        for (int j = 0; j <= Y; j++){
            dp[i][j] = X*Y;
        }
    }
    for(int i=1; i<=X; i++){
        for(int j=1; j<=Y; j++){
            if(possible_plots[i][j]){
                dp[i][j]=0;
                continue;
            }

            int min_vertical = i*j+1;
            for (int k=1; k<=j; k++){
                int min_vertical_new = dp[i][k] + dp[i][j-k];
                if(min_vertical_new < min_vertical)
                    min_vertical = min_vertical_new;
```



```cpp
            int min_horizontal = i*j;
            for (int k=1; k<=i; k++){
                int min_horizontal_new = dp[k][j] + dp[i-k][j];
                if(min_horizontal_new < min_horizontal)
                    min_horizontal = min_horizontal_new;
            }

            dp[i][j] = std::min(i*j, min_horizontal);
            dp[i][j] = std::min(dp[i][j], min_vertical);
        }
    }
     return dp[X][Y];
}

```

test.cpp

```cpp
#include <iostream>
#include <fstream>
#include "the4.h"


void read_from_file(int& X, int& Y, bool**& possible_plots){
    int number_of_plots;
    char addr[]= "inp03.txt"; // 01-10 are available
    std::ifstream infile (addr);
    if (!infile.is_open()){
        std::cout << "File \'"<< addr
                  << "\' can not be opened. Make sure that this file exists." << std::endl;
        return;
    }
    infile >> X;
    infile >> Y;
    infile >> number_of_plots;
    possible_plots = new bool*[X+1];
    for(int temp=0; temp < X+1; temp++) possible_plots[temp] = new bool[Y+1];
    for(int idx=0; idx < X+1; idx++) for(int idy=0; idy < Y+1; idy++) possible_plots[idx][idy] = false;
    for(int temp=0; temp < number_of_plots; temp++){
        std::pair<int, int> plot;
        infile >> plot.first >> plot.second;
        possible_plots[plot.first][plot.second] = true;
    }
    infile.close();
}

int main(){
    int X, Y;
    bool** input_array;
    int minimum_unused_land, plot_number=1;

    read_from_file(X, Y, input_array);

    std::cout << "X: " << X << ", Y: " << Y << std::endl;
    for(int idx=0; idx < X+1; idx++)
        for(int idy=0; idy < Y+1; idy++)
            if(input_array[idx][idy])
                std::cout << "Plot " << plot_number++ << ": (" << idx << ", " << idy << ")" << std::endl;

    minimum_unused_land = divide_land(X, Y, input_array);

    std::cout << "Unused land: " << minimum_unused_land << " m^2" << std::endl;

    for(int idx=0; idx<X+1; idx++) delete[] input_array[idx];
    delete[] input_array;
    return 0;
}
```

the4.h

```cpp
#ifndef THE4_THE4_H
#define THE4_THE4_H
#include <vector>
#include <utility>
#include <algorithm>
#include <climits>

//updating this file will not change the execution in the VPL

int divide_land(int X, int Y, bool** possible_plots);

#endif //THE4_THE4_H
```

VPL

You are logged in as mustafa baris emektar (Log out)
CENG 315 ALL Sections
Get the mobile app