Name:Omar youssef

ID:43

Name:Islam Mostafa gaber

ID:13

Name:Omar Emam

ID:42

# Assignment 2

## Steps :

1-we create the board of the game by using 2d array

2-We made some functions to check that the input location is true

Is valid location() :return true if our input in place equal zero in the two demension array

Drop_circle(): put the piece in the board game

Get_next_empty_row(): to check next empty location in the raw.

Winning_move():to check if there are four in diagonal or horizontal or vertical

Score_position():

Count num of circles in row ,column or diagonal two or three .

Pick_best_move():to get the best location to put circle if found three it put the fourth one to win .

Score variable try to make ai better.

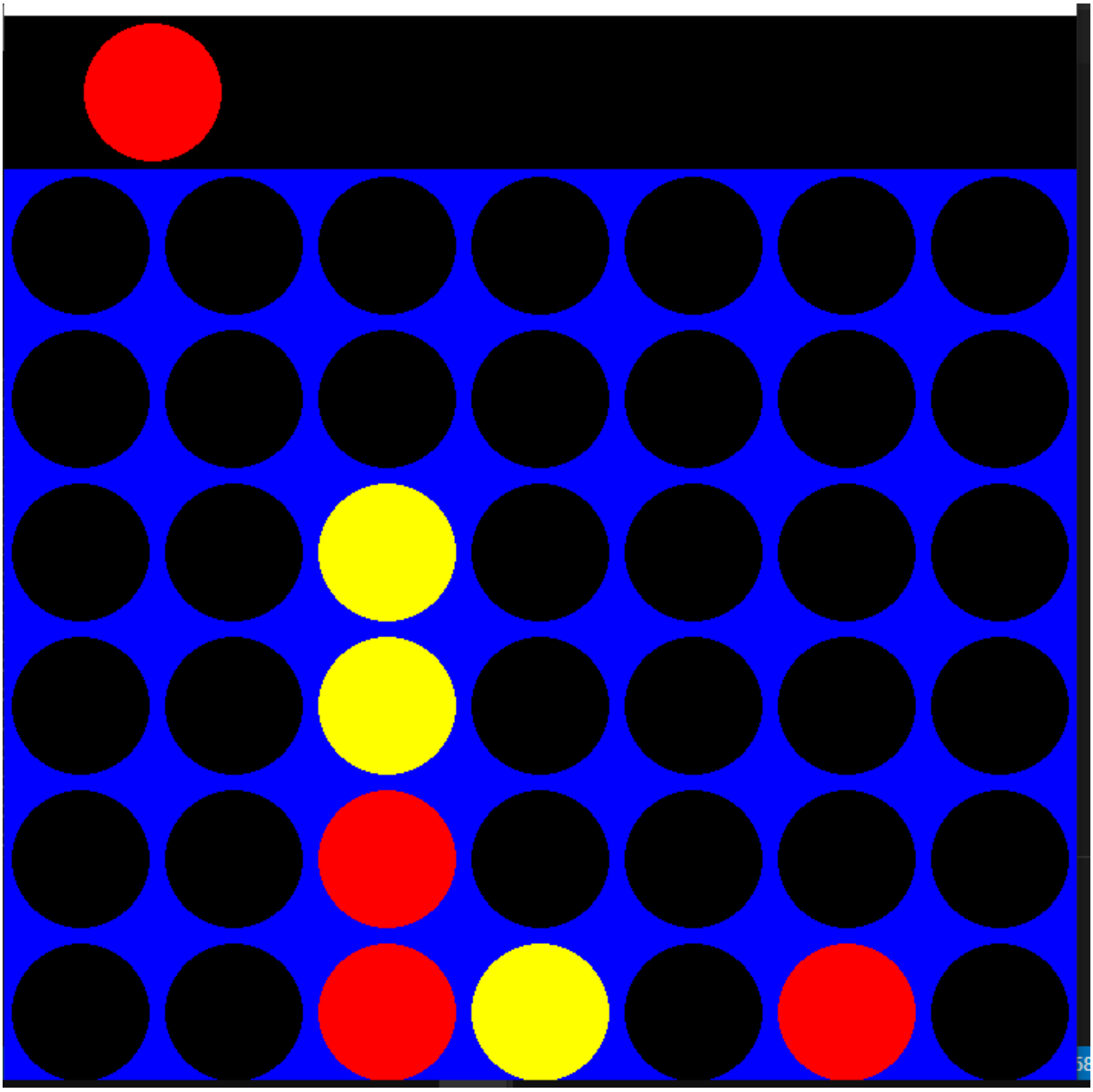
## We made two algorithms implementing the game :

1- Minimax without alpha-beta pruning
2- 2- Minimax with alpha-beta pruning

# How game start :

First appear two buttons to choose which  algorithm we want to run :

Then an open window and we start to play:



## How it work:

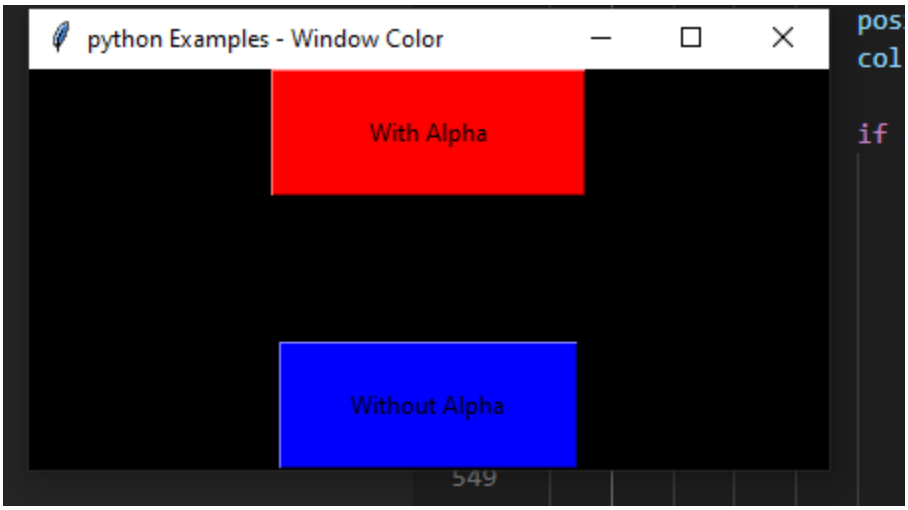
The computer start playing with the yellow circle then the human with red circle

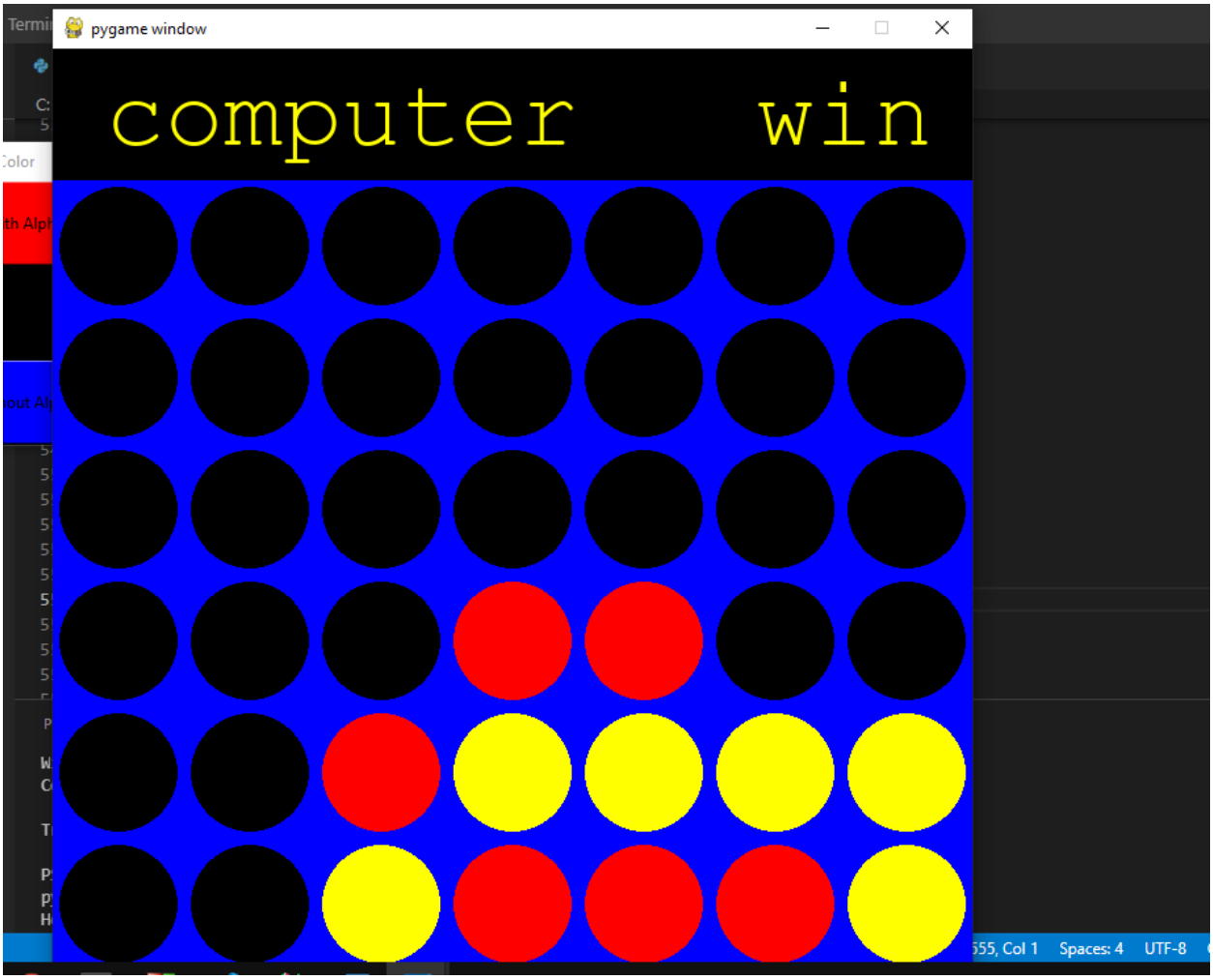
## The two algorithms :

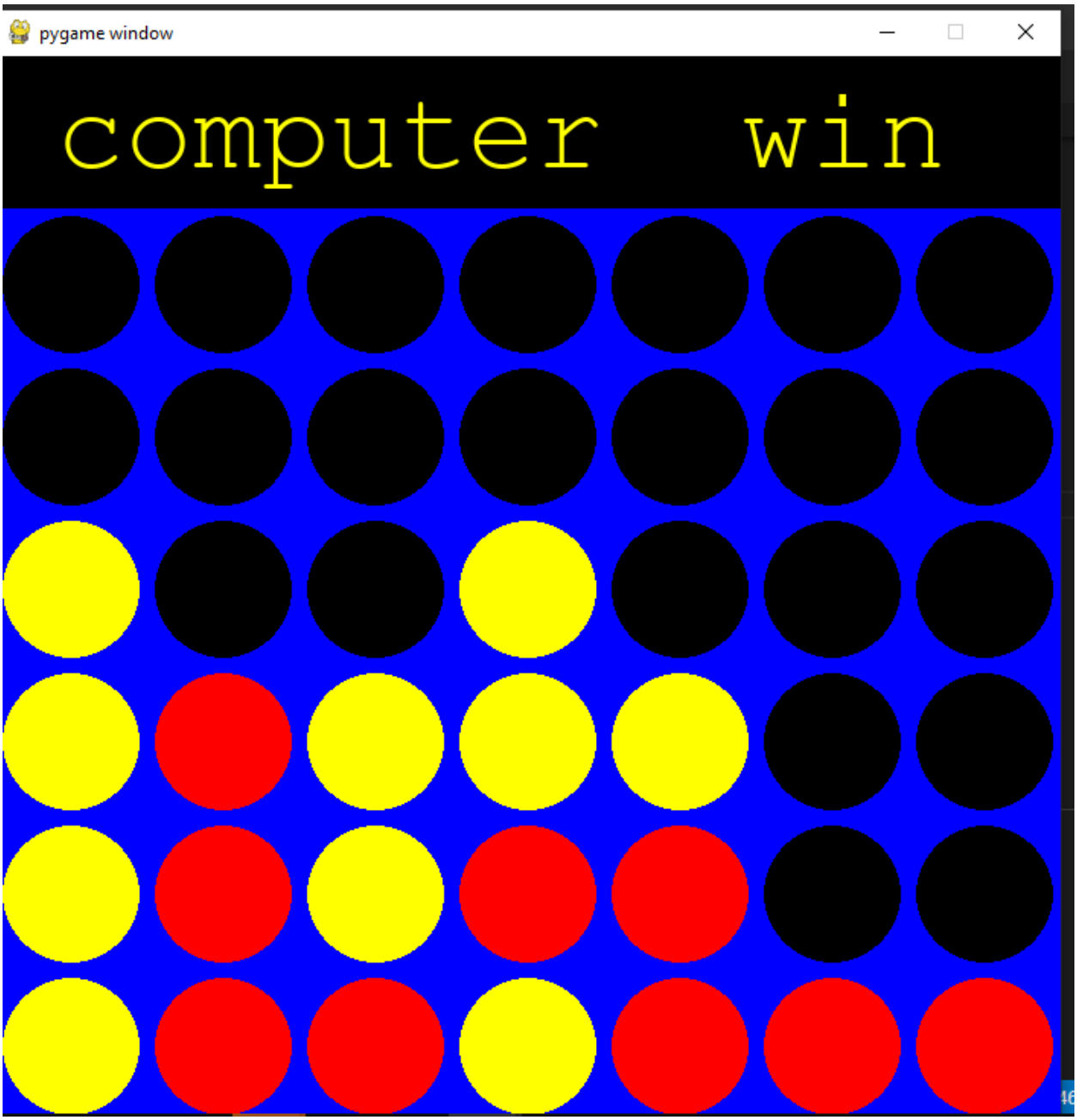
Minimax without alpha-beta pruning:

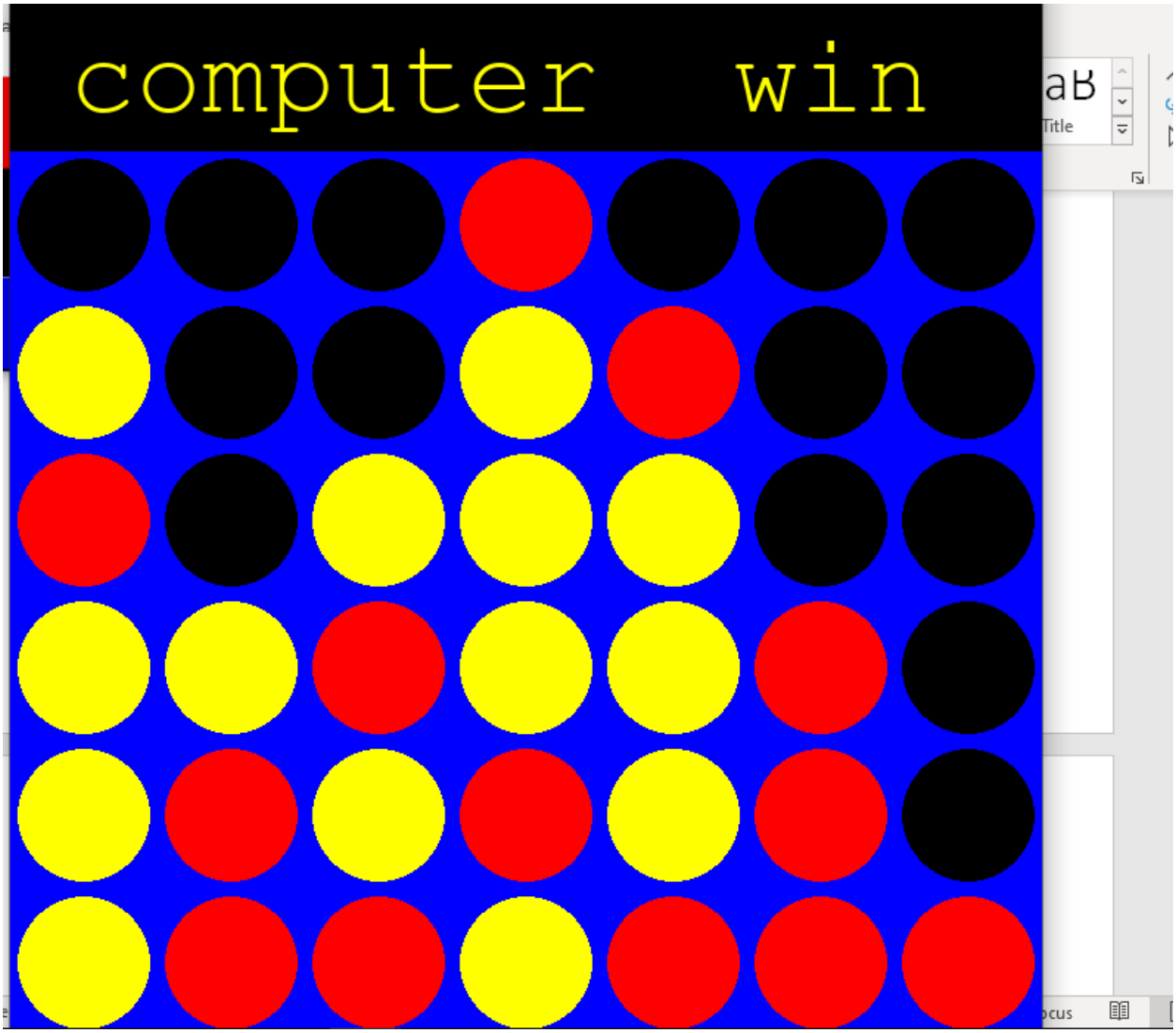
It take very long time to check all possible solutions as it check the whole tree but in the second algorithm Minimax with alpha-beta pruning

## Sample runs


python Examples - Window Color
With Alpha
Without Alpha
549
pos
col
if


pygame window
computer win
555, Col 1
Spaces: 4
UTF-8


pygame window
computer win




## Run time with alpha beta with k=5

```
[[0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 2. 0. 0. 0. 0.]
 [0. 0. 2. 2. 0. 0. 0.]
 [0. 0. 2. 1. 0. 0. 0.]
 [0. 1. 1. 2. 1. 0. 0.]
 [0. 1. 2. 1. 1. 1. 2.]]
Runtime of the program is 18.107486248016357
[[0. 0. 2. 0. 0. 0. 0.]
 [0. 0. 2. 0. 0. 0. 0.]
 [0. 0. 2. 2. 0. 0. 0.]
```

## Run time without alpha beta with k=5

```
 [0. 1. 1. 2. 0. 0. 0.]]
Runtime of the program is 60.53298830986023
[[0. 0. 0. 0. 0. 0. 0.]
```