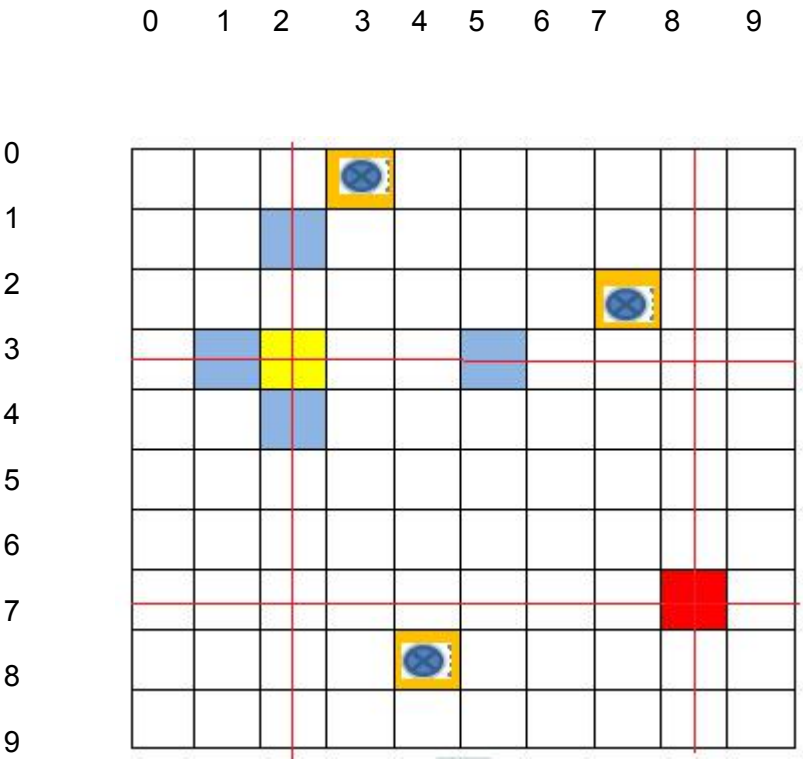
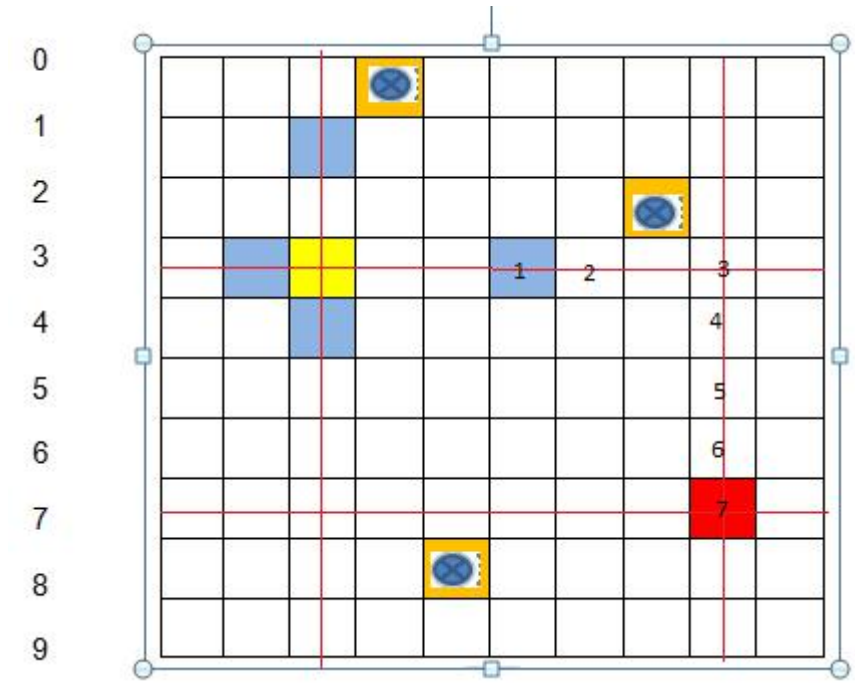
Project 11

Look at the first Figure. We have a 10*10 board. The rows and columns are numbered from 0 to 9. The picture shows the way to calculate the next move. The yellow square (2,3) is the start position, and the blue squares are the possible positions resulting from the yellow square. Normally, you can move one unit to each direction. If your move is in the same row or in the same column with the bombs (ex:(0,3)), then you should skip and move two units ex: (2,1) or sometimes up to 3 units ex:(5,3).





Now if you are given the start position (2,3) and you want to take it to a destination square (8,7) it is possible to generate a sequence of moves to reach the destination. For example look at the second figure and numbers towards goal (red). There may be alternative paths to reach the destination square.

Write a program that will calculate the minimum number of moves required to go from a source-square to a destination square. This is a search problem. You have to implement it in **DFS.**

Input: You will be given a text file called **problem.txt**. The file will be located in the current directory. The first line contains an integer **n** specifying the number of problems in the file. Each line will contain a problem. In each problem line, the first pair of integers is the source position of the knight. Next pair specifies the destination.

Please see the following sample **problem.txt** file.

Line number Contents:

1. 1

2. 2.3 8 7

Your code is supposed to create 3 random bombs (mines) after getting the start and end points but in a way that it won't correspond to start and end points from any row and column. Check the red lines on the figures.

Output: All outputs should be written in a file called **solution.txt** in the current directory. For each problem in **problem.txt**, there will be a **corresponding line** in **solution.txt** containing one integer: the minimum number of moves required for the corresponding problem and the path and also the location of the randomly created mines. You can put 0 if there is no solution and number of steps if there is a solution. The sample **solution.txt** for the sample **problem.txt** is as follows:

Line number Contents:

1. ***7 (***3,0)-(4,8)- (7,2); (2,3)(5,3)(6,3)(8,3)(8,4)(8,5)(8,6)(8,7)

OUTPUT ON THE BOARD: (see the general Project guidelines and customize it for your project)