AI Lab Test - 1

1BM18CS064
Nitesh Garg

```
def print_instructions():
    print_board (pos)
    print ("C1 will use X")
    print ("C2 will use O")
print("positions are 1 2 3 4 5 6 7 8 9")
      flag = input()
      compfirst (flag)

def print_board ();
    print ("a[1]", "a[2]", "a[3]")
    print ("a[4]", "a[5]", "a[6]")
    print ("a[7]", "a[8]", "a[9]")

def compfirst (flag):
    if flag == S
         start compfirst()
    else
       print ("Invalid")

def start compfirst():
    turn = 0
     for i in range (9):
          if turn %2 == 0
               print ("computer 1 is playing")
               v = "x"
               if i == 0:
                    p = 9
                    pos[p] = v
                    print_board (pos)
                    turn = 1
                    continue
```

```
        else :
            p= compwin (i)
            pos[p] = v
            print_board (pos)
            winner = check_win (v)
            if winner == "nobody"
                  turn = 1
                  continue
            else:
              print ("computer 1: wins")
              break
   else:
      print ("computer 2 is playing")
      p=logic ("o")
      v= "o"
      pos[p] =v
      print_board (pos)
      Winner = checkwin (v)
       if Winner = "nobody"
            turn =0
             continue
       else
            print ("Computer 2 wins")
            break
 else:
    print("game is tied")
```

```
def compwin(i):
    if pos[5] == " " and i==2 :
        if pos[1] == " "
            return 1
        elif pos[3] == " "
            return 3
        elif pos[7] == " "  return 7
        elif pos[9] == " "  return 9

    if pos[5] != " " and i== 2 :
        return 1

    for t in winning-conditions:
        if pos[t[0]] == "x" and pos[t[1]] == "x" and pos[t[2]] == " "
            return t[2]
        if pos[t[0]] == " " and pos[t[1]] == "x" and pos[t[2]] = "x"
            return t[0]
        if pos[t[0]] == "x" and pos[t[1]] == " " and pos[t[2]] == "x"
            return t[1]

    for t in winning conditions:
        if pos[t[0]] == "o" and pos[t[1]] == "o" and pos[t[2]] == " "
            return t[2]
        if pos[t[0]] == " " and pos[t[1]] == "o" and pos[t[2]] == "o"
            return t[0]
        if pos[t[0]] == "o" and pos[t[1]] == " " and pos[t[2]] = "o"
            return t[1]

    if pos[7] == " " and pos[5] == " "
        return 7
else return pos.index(" ")
```

```
def checkwin(v):
    for i in winning_conditions
        if (pos[i[0]], pos[i[1]], pos[i[2]]) == (v, v, v):
            ~~winner = players[o]~~
                ~~break~~
            ~~elif (pos~~
                if v == "x"
                    winner = player[0]  break {
                else
                    winner = player[1]  break ~~{~~

        else
            winner = "nobody"
    return winner

def logic(v):
    if "o" not in pos:
        if pos[5] == " "
            return 5
        else return 1
    else:
        for t in winning_conditions:
            c = 0
            awin = 0
            if pos[t[0]] == "o" and pos[t[1]] == "o" and pos[t[2]] == "
                return ~~p~~ t[2]
            elif pos[t[0]] == " " and pos[t[1]] == "o" and pos[t(2]] == "o
                return ~~{~~ t[0]
            elif pos[t[0]] == "o" and pos[t[1]] == " " and pos[t[2]] == "o
                return t[1]
    if  pos[3] == " " :
        return  3
    elif  pos[6] == " " : return 6
    elif  pos[9] == " " : return 9

    return  pos.index(" ")

pos = [" ", " ", " ", " ", " ", " ", " ", " ", " "]
players = [ 'Comp1', 'Comp 2']
winning_conditions = [ (1,2,3), (4,5,6), (7,8,9),
                       (1,4,7), (2,5,8), (3,6,9),
                       (1,5,9), (3,5,7)]
```