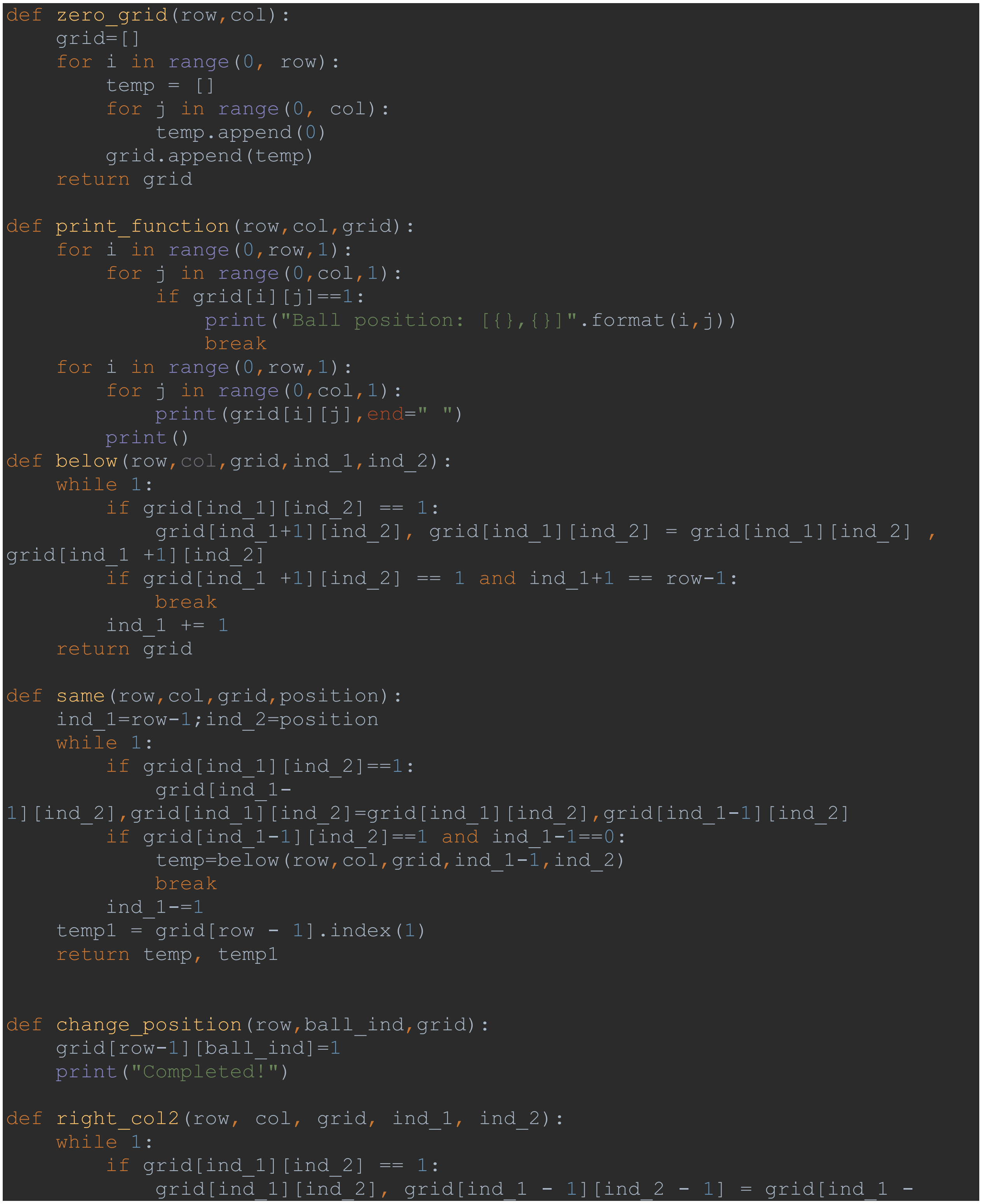
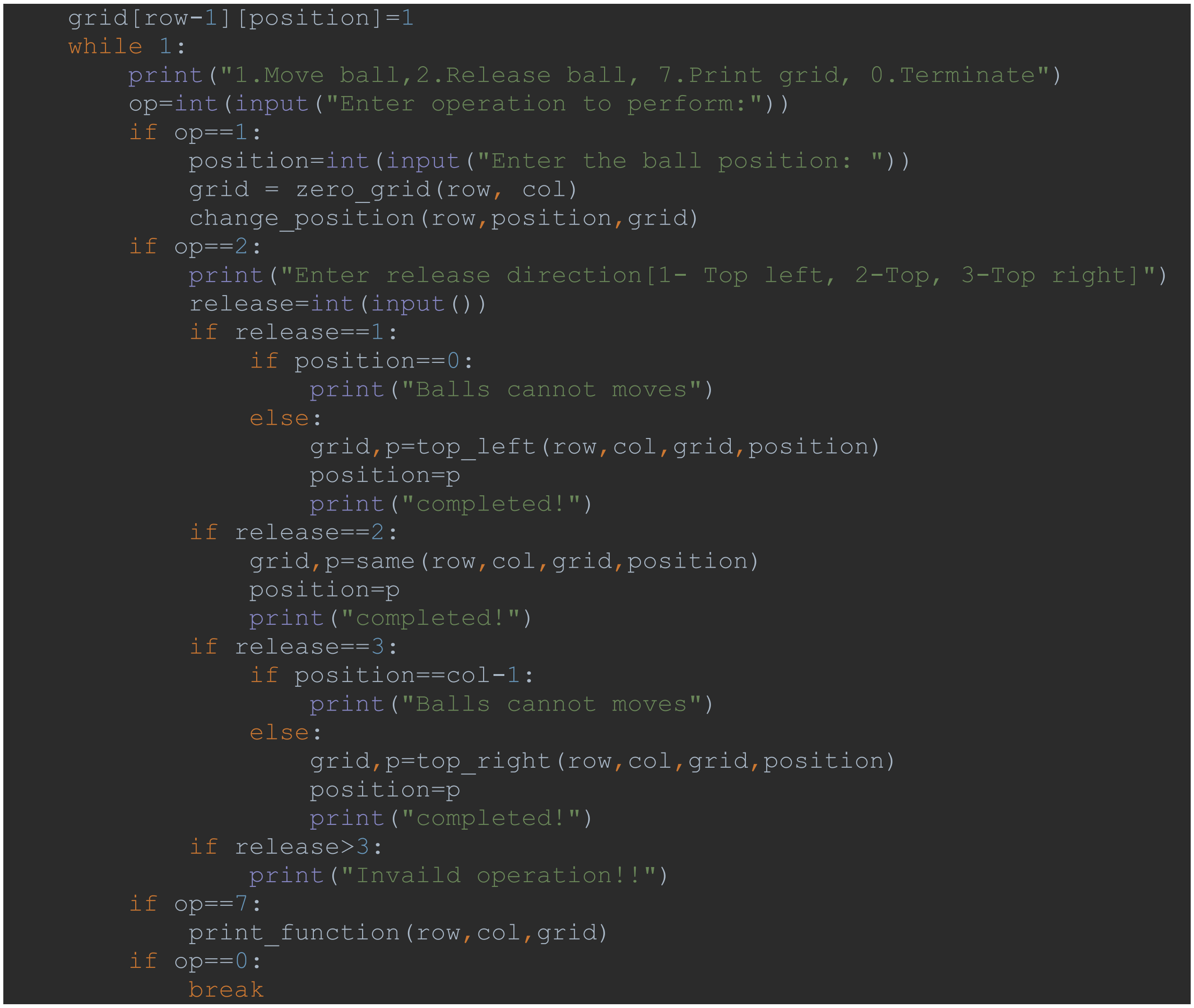
Bounce Ball Game



```
def zero_grid(row,col):
    grid=[]
    for i in range(0, row):
        temp = []
        for j in range(0, col):
            temp.append(0)
        grid.append(temp)
    return grid

def print_function(row,col,grid):
    for i in range(0,row,1):
        for j in range(0,col,1):
            if grid[i][j]==1:
                print("Ball position: [{},{}]".format(i,j))
                break
    for i in range(0,row,1):
        for j in range(0,col,1):
            print(grid[i][j],end=" ")
        print()
def below(row,col,grid,ind_1,ind_2):
    while 1:
        if grid[ind_1][ind_2] == 1:
            grid[ind_1+1][ind_2], grid[ind_1][ind_2] = grid[ind_1][ind_2] ,
grid[ind_1 +1][ind_2]
        if grid[ind_1 +1][ind_2] == 1 and ind_1+1 == row-1:
            break
        ind_1 += 1
    return grid

def same(row,col,grid,position):
    ind_1=row-1;ind_2=position
    while 1:
        if grid[ind_1][ind_2]==1:
            grid[ind_1-
1][ind_2],grid[ind_1][ind_2]=grid[ind_1][ind_2],grid[ind_1-1][ind_2]
        if grid[ind_1-1][ind_2]==1 and ind_1-1==0:
            temp=below(row,col,grid,ind_1-1,ind_2)
            break
        ind_1-=1
    temp1 = grid[row - 1].index(1)
    return temp, temp1


def change_position(row,ball_ind,grid):
    grid[row-1][ball_ind]=1
    print("Completed!")

def right_col2(row, col, grid, ind_1, ind_2):
    while 1:
        if grid[ind_1][ind_2] == 1:
            grid[ind_1][ind_2], grid[ind_1 - 1][ind_2 - 1] = grid[ind_1 -
1][ind_2 - 1], grid[ind_1][ind_2]
        if grid[ind_1-1][ind_2-1]==1 and ind_1-1==0:
            temp=left_row1(row, col, grid, ind_1-1, ind_2-1)
            break
        ind_1-=1;ind_2-=1
    return temp

def left_col1(row, col, grid, ind_1, ind_2):
    while 1:
        if grid[row-1].count(1)==1:
            break
        if grid[ind_1][ind_2]==1:

grid[ind_1][ind_2],grid[ind_1+1][ind_2+1]=grid[ind_1+1][ind_2+1],grid[ind_1][
ind_2]

        ind_1+=1;ind_2+=1
    return grid
def left_row1(row, col, grid, ind_1, ind_2):
    while 1:
        if grid[ind_1][ind_2]==1:
            grid[ind_1][ind_2],grid[ind_1+1][ind_2-1]=grid[ind_1+1][ind_2-
1],grid[ind_1][ind_2]
        if grid[ind_1+1][ind_2-1]==1 and ind_2-1==0:
            temp=left_col1(row, col, grid, ind_1+1, ind_2-1)
            break
        if grid[row-1].count(1)==1:
            temp=grid
            break
        ind_1+=1;ind_2-=1
    return temp


def left_col_2(row,col,grid,ind_1,ind_2):
    while 1:
        if grid[row-1].count(1)==1:
            grid=grid
            break
        if grid[ind_1][ind_2]==1:
            grid[ind_1][ind_2],grid[ind_1+1][ind_2-1]=grid[ind_1+1][ind_2-
1],grid[ind_1][ind_2]
        ind_1+=1;ind_2-=1
    return grid

def row_right1(row,col,grid,ind_1,ind_2):
    while 1:
        if grid[ind_1][ind_2]==1:
            grid[ind_1][ind_2], grid[ind_1 +1][ind_2 + 1] = grid[ind_1 +
1][ind_2 + 1], grid[ind_1][ind_2]
        if grid[ind_1 + 1][ind_2 + 1]==1 and ind_2+1==col-1:
            temp=left_col_2(row,col,grid,ind_1+1,ind_2+1)
            grid =temp
            break
        if grid[row-1].count(1)==1:
            grid=grid
            break
        if grid[ind_1 + 1][ind_2 + 1]==1 and ind_2+1==col-1:
            temp=left_col_2(row,col,grid,ind_1+1,ind_2+1)
            grid=temp
        ind_1+=1;ind_2+=1
    return grid
def right_col1(row,col,grid,ind_1,ind_2):
    while 1:
        if grid[ind_1][ind_2]==1:
            grid[ind_1][ind_2],grid[ind_1-1][ind_2+1]=grid[ind_1-
1][ind_2+1],grid[ind_1][ind_2]
        if grid[ind_1-1][ind_2+1]==1 and ind_1-1==0:
            temp= row_right1(row, col, grid, ind_1 - 1, ind_2 + 1)
            break
        ind_1-=1;ind_2+=1
    return temp

def top_left(row,col,grid,position):
    ind_1=row-1;ind_2=position
    while 1:
        if grid[ind_1][ind_2]==1:
            grid[ind_1][ind_2],grid[ind_1-1][ind_2-1]=grid[ind_1-1][ind_2-
1],grid[ind_1][ind_2]
        if grid[ind_1 - 1][ind_2 - 1] == 1 and ind_1 - 1 == 0 and ind_2-1==0:
            temp=row_right1(row, col, grid, ind_1 - 1, ind_2 - 1)
            break
        if grid[ind_1 - 1][ind_2 - 1]==1 and ind_2-1==0:
            temp= right_col1(row, col, grid, ind_1 - 1, ind_2 - 1)
            break
        if grid[ind_1 - 1][ind_2 - 1]==1 and ind_1-1==0:
            temp=left_row1(row, col, grid, ind_1 - 1, ind_2 - 1)
            break
        ind_1-=1;ind_2-=1
    temp1=grid[row-1].index(1)
    return temp,temp1

def top_right(row,col,grid,position):
    ind_1=row-1;ind_2=position
    while 1:
        if grid[ind_1][ind_2]==1:
            grid[ind_1][ind_2],grid[ind_1-1][ind_2+1]=grid[ind_1-
1][ind_2+1],grid[ind_1][ind_2]
        if grid[ind_1-1][ind_2+1]==1 and ind_1-1==0 and ind_2+1==col-1:
            temp=left_row1(row, col, grid, ind_1 - 1, ind_2 +1)
            break
        if grid[ind_1-1][ind_2+1]==1 and ind_2+1==col-1:
            temp=right_col2(row, col, grid, ind_1 - 1, ind_2 +1)
            break
        if grid[ind_1-1][ind_2+1]==1 and ind_1-1==0:
            temp=row_right1(row, col, grid, ind_1 - 1, ind_2 +1)
            break
        ind_1-=1;ind_2+=1
    temp_1=grid[row-1].index(1)
    return temp,temp_1

if __name__=="__main__":
    row,col=map(int,input("Enter grid[r,c]: ").split())
    grid = zero_grid(row, col)
    position=int(input("Enter the ball's initial position: "))
```



```
    grid[row-1][position]=1
    while 1:
        print("1.Move ball,2.Release ball, 7.Print grid, 0.Terminate")
        op=int(input("Enter operation to perform:"))
        if op==1:
            position=int(input("Enter the ball position: "))
            grid = zero_grid(row, col)
            change_position(row,position,grid)
        if op==2:
            print("Enter release direction[1- Top left, 2-Top, 3-Top right]")
            release=int(input())
            if release==1:
                if position==0:
                    print("Balls cannot moves")
                else:
                    grid,p=top_left(row,col,grid,position)
                    position=p
                    print("completed!")
            if release==2:
                grid,p=same(row,col,grid,position)
                position=p
                print("completed!")
            if release==3:
                if position==col-1:
                    print("Balls cannot moves")
                else:
                    grid,p=top_right(row,col,grid,position)
                    position=p
                    print("completed!")
            if release>3:
                print("Invaild operation!!")
        if op==7:
            print_function(row,col,grid)
        if op==0:
            break
```