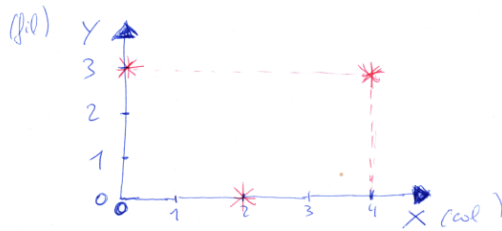


## Ejercicio U4\_B4D\_E1:



Necesito matriz 4x5  
 ↑    ↑  
 i    col

	0	1	2	3	4
0			*		
1					
2					
3	*				*

```

class PlanoJardin{
    //no se almacenan las dimensiones del plano, se deducen del tamaño de la matriz
    char[][] matriz;

    PlanoJardin(int x, int y){
        //x son las líneas verticales, es decir, columnas
        //y son las líneas horizontales, es decir, filas

        matriz= new char[y][x]; //ojo con orden y x
        //podría recorrer al revés, primero con columnas
        //aquí es indiferente.
        for(int i=0; i<matriz.length; i++){
            for(int j=0; j<matriz[i].length; j++){
                matriz[i][j]='-'; //significa sin árbol
            }
        }

        void plantarArbol(int x,int y){
            //x es columna e y es fila
            matriz[y][x]='A';
        }

        void imprimirPlano(){
            //empezamos a imprimir por última fila ya que la fila 0 es la que está más abajo en el eje x,y de un
            cuadrante
            for(int i=matriz.length-1; i>=0; i--){
                for(int j=0; j<matriz[i].length; j++){
                    System.out.print(matriz[i][j]+" ");
                }
                System.out.println("");
            }

            boolean hayArbol(int x,int y){
                return matriz[y][x]!='A';
            }
        }

        class Unidad4{

            public static void main(String[] args) {

```

```

        //se indica al constructor columnasxfilas
        PlanoJardin plano=new PlanoJardin(5,4);
        System.out.println("antes de plantar");
        plano.imprimirPlano();
        plano.plantarArbol(0, 3);
        plano.plantarArbol(4,3);
        plano.plantarArbol(2, 0);
        System.out.println("despues de plantar");
        plano.imprimirPlano();
        System.out.println(plano.hayArbol(0,3));
        plano.plantarArbol(2, 0);      System.out.println(plano.hayArbol(3,0));
    }
}

```

## Ejercicio U4\_B4D\_E2:

una traducción: Recuerda que X de matemáticas son las columnas de arrays, y las Y de matemáticas son las filas

$(X,Y)=[FILA][COLUMNA]$

donde

$FILA=Y+matriz.length/2$

$COLUMNA=X+matriz[0].length/2$

( suponemos todas las filas del mismo tamaño)

```

class PlanoJardin{
    //no es necesario almacenar las dimensiones plano jardín pues se deducen del tamaño de la matriz
    char[][] matriz;
    PlanoJardin(int x, int y){
        matriz= new char[y][x]; //ojo con orden y x
        //podría recorrer al revés, primero con columnas
        //aquí es indiferente.

        for(int i=0;i<matriz.length;i++){
            for(int j=0;j<matriz[i].length;j++){

                matriz[i][j]='-'; //significa sin arbol
            }
        }
        //en la coordenada "central" pongo la casa
        matriz[desplazarY()][desplazarX()='C';

    }
    void plantarArbol(int x,int y){
        //x es columna e y es fila
        x=x+desplazarX();
        y=y+desplazarY();
        //ojo observar que cambio orden de x y
        matriz[y][x]='A';
    }

    void imprimirPlano(){
        //empezamos a imprimir por última fila ya que la fila 0 es la que está más abajo en el eje x,y de un cuadrante
        for(int i=matriz.length-1;i>=0;i--){
            for(int j=0;j<matriz[i].length;j++){
                System.out.print(matriz[i][j]+" ");
            }
            System.out.println("");
        }
    }

    boolean hayArbol(int x,int y){
        x=x+desplazarX();
        y=y+desplazarY();
        return matriz[y][x]=='A';
    }
}

```

```

int desplazarX(){
    return matriz[0].length/2;
}
int desplazarY(){
    return matriz.length/2;
}
}
class Unidad4{

    public static void main(String[] args) {
        //se indica al constructor hileras verticales y horizontales
        PlanoJardin plano=new PlanoJardin(11,9);
        System.out.println("Antes de plantar");
        plano.imprimirPlano();
        plano.plantarArbol(2,3);
        plano.plantarArbol(2,-3);
        plano.plantarArbol(-2,3);
        plano.plantarArbol(-2,-3);
        System.out.println("Despues de plantar");
        plano.imprimirPlano();
        System.out.println(plano.hayArbol(2,-3));
        System.out.println(plano.hayArbol(3,0));

    }
}

```