

CI0112 Programación 1  
Segundo parcial  
2 horas

1. Implemente el método recursivo ***int obtengaSumaImpar(...)*** que trabaja sobre el arreglo ***v*** y que devuelve la suma de los números impares en dicho vector. (25%)

Ejemplo:

```
v=new int[]{2,4,6,1,7,8}; //instanciación para pruebas en el constructor
```

```
System.out.println(obj.System.out.println(r.obtengaSumaImpar(...));
```

```
//obj = objeto actual y ... parámetro(s) del método
```

Salida:

8

2. Dada una matriz ***m*** de enteros (ya instanciada), haga una aplicación que recorriendo recursivamente dicha matriz, obtenga la suma de todos los valores en ella. Ojo Al igual que al manejar 2 dimensiones, se usan 2 for anidados, si la implementación es recursiva, puede requerirse el anidamiento de métodos recursivos. Ejemplo: siendo ***m*** la siguiente matriz: (35%)

m	0	1	1
	2	1	1
	3	2	1

si se llama el método recursivo ***sumeCeldas(...)*** acorde con la siguiente instrucción:

```
System.out.println("Suma de celdas = "+obj.sumeCeldas(...)); //obj es el nombre del objeto definido
```

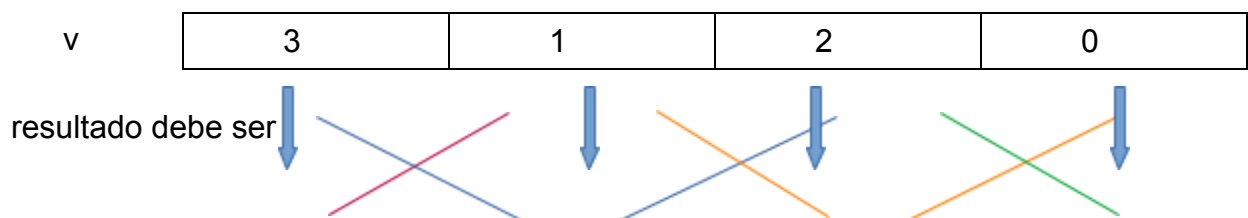
se debe desplegar Suma de celdas = 12

3. Dado el vector global ***v*** de enteros ya instanciado, por ejemplo:

```
int v[];
```

```
v=new int{3,1,2,0}; //dentro del constructor
```

defina el método recursivo ***int deFuerzaVecinal()***, que recibe entre los parámetros: el índice de fila y de la columna actual. Para cada celda en el vector, retorna la fuerza de sus celdas vecinas, definida ésta como la suma de las 2 celdas alrededor. Ejemplo: (25%)



fortalez  
aVecina

1	5	1	2
---	---	---	---

haga un método que muestre el vector inicial y la fuerza vecinal. (5%)

4. Idem al anterior, pero haciendo una implementación iterativa (no recursiva) (10%)