

# **INGENIERÍA DEL SOFTWARE 2 - SEGUNDO EJERCICIO TEORÍA**

## **ET.02.02 Primer Problema**

### **Autores:**

**-Diego Sepúlveda Milán**

**-David Valbuena Ríos**

---

### **Índice**

- 1. Código.**
- 2. Variables implicadas.**
- 3. Valores de prueba.**
- 4. Máximo posible de casos de prueba.**
- 5. Casos de pruebas para cumplimentar each use.**
- 6. Casos de pruebas para alcanzar cobertura pairwise.**
- 7. Conjunto de casos de prueba, del código que incluye decisiones, para alcanzar cobertura decisiones.**
- 8. Conjunto de casos de prueba, del código que incluye decisiones, para alcanzar cobertura MC/DC.**

## 1. Código.

### Main.java

```
1
2 public class Main {
3
4     public static void main(String[] args) {
5         Estado estadoPrueba= new Estado(true,false,false, true, true, 31, 0, false, false, false);
6         System.out.println(estadoPrueba.hacerRecomendacion());
7     }
8
9 }
10
```

### Estado.java

```
1
2 public class Estado {
3     //atributos respectivos a la persona
4     boolean sana;
5     boolean sintomas;
6     boolean contactoCovid;
7     boolean covidSuperado;
8     boolean cartillaVacuRegla;
9     //atributos respectivos al ambiente
10    int temperatura;
11    int humedad;
12    boolean nieve;
13    boolean lluvia;
14    boolean niebla;
15
16    //constructor
17    public Estado(boolean sana, boolean sintomas, boolean contactoCovid, boolean covidSuperado,
18        boolean cartillaVacuRegla, int temperatura, int humedad, boolean nieve, boolean lluvia, boolean niebla) {
19        super();
20        this.sana = sana;
21        this.sintomas = sintomas;
22        this.contactoCovid = contactoCovid;
23        this.covidSuperado = covidSuperado;
24        this.cartillaVacuRegla = cartillaVacuRegla;
25        this.temperatura = temperatura;
26        this.humedad = humedad;
27        this.nieve = nieve;
28        this.lluvia = lluvia;
29        this.niebla = niebla;
30    }
31
32    //Metodos necesarios para efectuar recomendación
33
34    //Devuelve true si por salud puede hacer alguna actividad y en caso contrario false
35    public boolean estadoSaludable() {
36        if (sana && !sintomas && !contactoCovid && covidSuperado && cartillaVacuRegla) {
37            return true;
38        } else {
39            return false;
40        }
41    }
42
43    //Devuelve las actividades que puede realizar segun el estado de salud y de tiempo
44    public String hacerRecomendacion() {
45        String recomendacion="";
46        if(estadoSaludable()) {
47            if(temperatura<0) {
48                if(humedad<15) {
49                    if(nieve || lluvia) {
50                        recomendacion+="-Por motivos metereologicos es mejor que permanezca en casa.\n";
51                    } else {
52                        recomendacion+="-Puede ir a esquiar, si no se supera el aforo permitido por la legislación pertinente.\n";
53                    }
54                }
55            }
56            if(temperatura>=0 && temperatura<15) {
57                if(humedad<15) {
58                    if(!lluvia) {
59                        recomendacion+="-Es posible ir a hacer senderismo, si no se supera aforo del espacio previsto.\n";
60                    }
61                }
62            }
63            if(temperatura>=15 && temperatura<25) {
64                if(!lluvia && !niebla) {
65                    if(humedad <=0) {
66                        recomendacion+="-Puede ir a hacer turismo al aire libre, si la ciudad no tiene restricciones de confinamiento.\n";
67                    }
68                }
69            }
70            if(temperatura>=25 && temperatura<35 && !lluvia) {
71                recomendacion+="-La recomendación es irse de cañas, si el establecimiento no tiene problemas de aforo.\n";
72            }
73            if(temperatura>30 && !lluvia) {
74                recomendacion+="-La recomendación es irse a la playa o a la piscina. La piscina no puede superar el aforo permitido.\n";
75            }
76        } else {
77            recomendacion="Por motivos de salud, no puede realizar ninguna de las actividades, quedese en casa.\n";
78        }
79        return recomendacion;
80    }
81}
```

## 2. Variables implicadas.

Variable	Tipo	Rango
sana	Boolean	True/False
Sintomas	Boolean	True/False
contactoCovid	Boolean	True/False
covidSuperado	Boolean	True/False
cartillaVacuRegla	Boolean	True/False
temperatura	int	$[-2^{31}, (2^{31})-1]$
humedad	int	$[-2^{31}, (2^{31})-1]$
nieve	Boolean	True/False
lluvia	Boolean	True/False
niebla	Boolean	True/False

## 3. Valores de prueba.

Parámetro	Clase de equivalencia		Conjunto de valores		Valores límite	Conjetura de error
sana	True	False	True	False		null
Sintomas	True	False	True	False		null
contactoCovid	True	False	True	False		null
covidSuperado	True	False	True	False		null
cartillaVacuRegla	True	False	True	False		null
temperatura	(-∞,0)	[0,15)	[15,25)	[25,35)	[35,∞)	[30,∞)
humedad	(-∞,15)	(-∞,60)	[60,∞)			
nieve	True	False	True	False		null
lluvia	True	False	True	False		null
niebla	True	False	True	False		null

## 4. Máximo posible de casos de prueba.

El número de casos de uso serían 16.777.216 sin contar los datos enteros. Contando con los datos enteros saldrían demasiadas posibilidades para realizar, por lo tanto, no se puede dar un número exacto (hay demasiadas posibilidades)

## 5. Casos de pruebas para cumplimentar each use.

-Criterios de cobertura

A={True, False}

B={-10, 0, 15, 20, 25, 30, 35, 40}

C={0, 15, 30, 60, 65}

New Estado(sana, síntomas, contactoCovid, covidSuperado, cartillaVacuRegla, temperatura, humedad, nieve, lluvia, niebla)

Test suite = { (False, False, False, False, False, 0, 0, False, False, False),

(True,False,False,True,True,-10,0,True,True,False),  
 (True,False,False,True,True,-10,0,False,False,False),  
 (True,False,False,True,True,0,0,True,False,False),  
 (True,False,False,True,True,20,30,False,False,False),  
 (True,False,False,True,True,30,0,False,False,False),  
 (True,False,False,True,True,35,0,False,False,False)}

## 6. Casos de pruebas para alcanzar cobertura pairwise.

Todas las posibles combinaciones de:

{ A + A + A + A + A + B + C + A + A + A }

Siendo: A={True, False}

B={-10, 0, 15, 20, 25, 30, 35, 40}

C={0, 15, 30, 60, 65}

## 7. Conjunto de casos de prueba, del código que incluye decisiones, para alcanzar cobertura decisiones.

<b>A</b>	sana	<b>J</b>	temperatura >= 0
<b>B</b>	sintomas	<b>K</b>	temperatura < 15
<b>C</b>	contactoCovid	<b>L</b>	temperatura >= 15
<b>D</b>	covidSuperado	<b>M</b>	temperatura < 25
<b>E</b>	cartillaVacuRegla	<b>N</b>	niebla
<b>F</b>	temperatura < 0	<b>O</b>	humedad < 60
<b>G</b>	humedad < 15	<b>P</b>	temperatura >= 25
<b>H</b>	nieve	<b>Q</b>	temperatura < 35
<b>I</b>	lluvia	<b>R</b>	temperatura > 30

Decisión 1

A	B	C	D	E	F	G	H	I	Output
0	0	0	0	0	0	0	0	0	F
...									
1	0	0	1	1	1	1	0	1	T
1	0	0	1	1	1	1	1	0	T
1	0	0	1	1	1	1	1	1	T
...									
1	1	1	1	1	1	1	1	1	F

Decisión 2

A	B	C	D	E	I	J	K	Output
0	0	0	0	0	0	0	0	F
...								
1	0	0	1	1	0	1	1	T
...								
1	1	1	1	1	1	1	1	F

Decisión 3

A	B	C	D	E	I	L	M	N	O	Output
0	0	0	0	0	0	0	0	0	0	F
...										
1	0	0	1	1	0	1	1	0	1	T
...										
1	1	1	1	1	1	1	1	1	1	F

Decisión 4

A	B	C	D	E	I	P	Q	Output
0	0	0	0	0	0	0	0	F
...								
1	0	0	1	1	0	1	1	T
...								
1	1	1	1	1	1	1	1	F

Decisión 5

A	B	C	D	E	I	R	Output
0	0	0	0	0	0	0	F
...							
1	0	0	1	1	0	1	T
...							
1	1	1	1	1	1	1	F

## 8. Conjunto de casos de prueba, del código que incluye decisiones, para alcanzar cobertura MC/DC.

-Método a comprobar "estadoSaludable":

```
"if (sana && !síntomas && !contactoCovid  
    && covidSuperado && cartillaVacuRegla)"
```

Caso n: (sana, síntomas, contactoCovid, covidSuper, cartillaRegla)

-Para la rama true:

Caso 0: (**true, false, false, true, true**)

-Para la rama false:

Caso 1: (**false**, false, false, true, true)

Caso 2: (true, **true**, false, true, true)

Caso 3: (true, false, **true**, true, true)

Caso 4: (true, false, false, **false**, true)

Caso 5: (true, false, false, true, **false**)

## 9. Conclusión.

9) Comente los resultados del número de los casos de pruebas conseguidos en los apartados 4, 5 y 6 ¿qué podría decirse algo de la cobertura alcanzada?