TESTING

JAVA Spiegazione del codice TODO

CASI di TEST scenari

1. un caso di test che mostri che è possibile spegnere due sensori non adiacenti.

Fatto. Caso di test test1

2. un caso di test in cui provi a spegnere tutti e quattro ma controlli che non è possibile

Fatto.

Ho controllato sempre che non ci fosse pericolo con assertFalse(alarm.pericolo());

La copertura è relativamente bassa:

Name	Statement	Branch	Loop	Term
> 🗁 GARGANTINI_TESTING	75,0 %	62,5 %	?	56,2 %

CASI di test stament coverage.

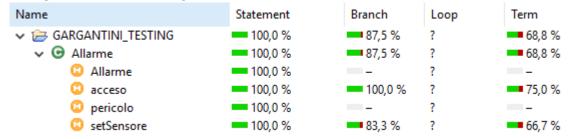
Controllo dove ho bassa copertura per i branch e statment

```
public boolean setSensore(int sensore, boolean acceso) {
        if (sensore >= 0 && sensore < 4 && sensori[sensore] != acceso) {</pre>
                if (acceso) {
                        sensori[sensore] = true
                } else {
                        // controlla se si può spegnere
                        int dx = (sensore - 1 + sensori.length) % sensori.length;
int sx = (sensore + 1) % sensori.length;
if (sensori[dx] && sensori[sx]) {
                                sensori[sensore] = false;
                                return true;
                        }
                }
        return false;
}
public boolean acceso(int sensore) {
        if (sensore < 0 || sensore > 3)
                         <mark>false;</mark> // <u>Ho</u> solo 4 <u>sensori</u>
        return sensori[sensore];
}
```

Aggiungo due casi in cui (1) accendo un sensore (prima lo spengo) e (2) chiamo con indici sbagliati il secondo metodo metodo.

StatementTest

Ottengo lo statement coverage

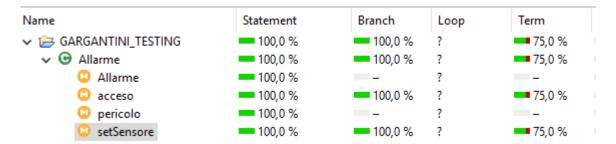


Ma ancora bassa branch coverage

In particolare ho il branch



sempre vero, faccio una caso in cui chiamo il metodo con indice sbagliato: assertFalse(alarm.setSensore(-1, false));



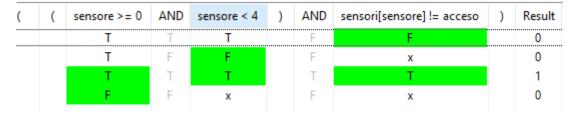
Ho 100% branch

MCDC

Alcune condizioni non sono coperti. Provo con una decisione, ad esempio

(sensore >= 0 && sensore < 4 && sensori[sensore] != acceso)</pre>

Essendo tre condizioni in AND, ne provo una false alla volta. Ottengo la copertura del 100% della decisione



Ci sono ancora due decisioni non coperta al 100% da MCDC ... TODO

Inserisco tutti i contratti. Dove facevo controlli sugli indici, lo metto come precondizione.

Ad esempio

```
public boolean acceso(int sensore) {
    if (sensore < 0 || sensore > 3)
        return false; // Ho solo 4 sensori
    return sensori[sensore];
}
Diventa

//@ requires sensore >= 0 && sensore <= 3;
public boolean acceso(int sensore) {
    return sensori[sensore];
}</pre>
```

Aggiungo anche contratti