

Esercizi partition and combinatorial testing

1. Partition testing

Dato un metodo che restituisce il numero di soluzioni per una equazione di secondo grado:

```
static public int numSolutions(int a, int b, int c){....}
```

- implementa il metodo
- Fai input domain modeling: modella l'insieme degli input in modo da coprirli secondo diversi criteri di copertura. Partiziona l'insieme $\text{int} \times \text{int} \times \text{int}$ (triple) in modi diversi:
 1. INTERFACE-BASED: considera una partizione per a, per b e per c e prendi dei valori significativi (però senza considerare il problema) per ognuna delle partizioni. Scrivi i casi di test in Junit per tutte le combinazioni delle partizioni (prodotto cartesiano) (all combinations). (o almeno parte di esse)
 2. FUNCTIONALITY-BASED: dividi il dominio in partizioni a seconda del risultato che ti aspetti dal metodo che devi testare. Prendi un rappresentante per partizione e scrivi i casi di test in JUnit.

2. Combinatorial testing con citlab

Applica il combinatorial testing al precedente modello (interface based approach)

Usa CITLAB:

<https://code.google.com/a/eclipselabs.org/p/citlab/>

La sintassi è la seguente:

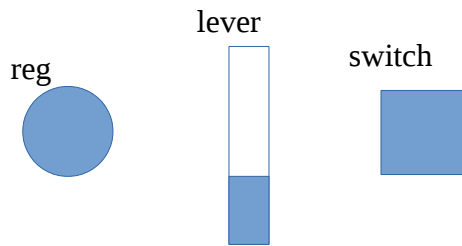
<https://code.google.com/a/eclipselabs.org/p/citlab/wiki/CitLabLanguage>

Prova a generare il pairwise usando citlab + ACTS

3. Combinatorial testing con constraints

Un sistema di controllo ha come input:

- uno switch
- una leva che può essere in 4 posizioni
- un regolatore da 1 a 5



Quando la leva è nella posizione piu' bassa lo switch deve essere off.

Prova a scrivere un modello citlab con i vincoli e creare una test suite combinatoriale

4. Software product lines

Una ditta sta sviluppando una applicazione APP per dispositivi mobili.

APP può andare su android, ios e windows phone.

Può funzionare sia con connessione che senza.

Si integra con il browser del telefonino (chrome, native, firefox).

Scrivi un modello con i relativi vincoli.

Genera tutti gli scenari in cui si dovrà testare il funzionamento di APP usando il pairwise.