

Progettazione del SW Sicuro

Linguaggi di Programmazione per la Sicurezza

Esame del 13 Marzo 2015 – (Parte Scritta on-line)

1. Dare la definizione di *stato* e di *transizione* nelle macchine di stato UML. [pt. 2]
2. Descrivere i principi su cui si basa il *design by contract* e gli elementi che costituiscono un contratto. Portare un esempio di contratto su un metodo pop di estrazione da una pila. [pt. 3]
3. Elencare le qualità di sicurezza di un prodotto software. [pt. 2]
4. Dare la definizione di test set affidabile, valido ed ideale. [pt. 2]
5. Sfruttando il meccanismo di sincronizzazione eventi-azioni di UML, modellare il funzionamento del seguente sistema termico. Il sistema è composto da un controllore e da una caldaia. In fase *giorno*, il controllore invia il comando di accensione alla caldaia se la temperatura ambiente è meno del valore standard settato dall'utente. L'utente può settare a piacere la temperatura dell'ambiente. Per default la temperatura ambiente è fissata a 20°. La caldaia si spegne automaticamente quando arriva in temperatura. In fase *notte*, il controllore invia alla caldaia un comando di accensione se la temperatura è inferiore ad un valore minimo. Tale valore può essere stabilito a piacere dall'utente. Per default, esso è fissato a 17°. Un eventuale guasto della caldaia, mette l'intero sistema in una situazione di *allarm*. [pt. 5]
6. Dato il seguente programma, disegnare il grafo di flusso e determinare una test suite per le condizioni mediante MCDC: [pt.3]

```
foo (int x, int y, int z) {  
    if (x < 2 && (y == 5 || x == 0)) {  
        z++;  
    } else {z--;}  
}
```
7. Data la macchina di stato FSM avente
S=(s0,s1,s2); I=(a,b,c); O=(0,1);
T=(<s0,a,1,s1>,<s1,a,0,s0>,<s1,b,0,s2>,<s2,a,0,s2>,<s2,c,1,s0>)}
darne la rappresentazione mediante state pattern. [pt. 3]

