

# Linguaggi di Programmazione per la Sicurezza

## Esame del 24 Febbraio 2011 – (Parte Scritta)

1. Descrivere *almeno tre* tra i principi guida di un'architettura sicura che riguardano malfunzionamenti ed errori di sistema. [pt. 3]
2. Descrivere le possibili classificazioni di testing a seconda del *livello* a cui si effettua, del tipo di *accesso* e degli *aspetti* da testare. [pt. 3]
3. Dato il seguente programma che dovrebbe fare la somma di due numeri, ma che contiene un difetto:

```
int somma (int x, int y ) {  
    if (x > y) return x*y else return x+y;  
}
```

fornire un test set affidabile ed uno ideale. [pt. 3]

4. Utilizzando le macchine di comunicazione, modellare il comportamento di un *elevatore* comandato da un *controller*. Il controller comanda l'elevatore in base alla richiesta dell'elevatore ai piani top, middle, low che avviene attraverso segnalazione del sensore *request*. Su segnalazione del sensore, il controller invia all'elevatore il comando di raggiungere la posizione richiesta. Raggiunto il livello, l'elevatore apre le porte ed attende il carico. Quando viene richiamato ad un livello diverso, prima di muoversi, chiude le porte, e reagisce al comando. [pt. 4]
5. Data la macchina di stato FSM avente  
S=(s0,s1,s2); I=(a,b); O=(0,1);  
T=(<s0,a,1,s1>,<s1,a,0,s0>,<s1,b,0,s2>,<s2,c,0,s2>,<s2,c,1,s0>)}  
darne la codifica Java mediante state pattern. [pt. 4]
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 ) {  
    if (x <2 && (y ==5 || x ==0)) {  
        z++;  
    } else {  
        z--;  
    }  
}
```