

Dare un controesempio semplice da cui si evince che i criteri di test per le decisioni e quelli per le condizioni sono tra loro indipendenti.

Scrivere la segnatura e l'opportuno contratto JML per un metodo che data una sequenza di interi, la ripartisca in due sequenze di interi positivi ed interi negativi.

Descrivere le possibili classificazioni di testing a seconda del livello a cui si effettua, del tipo di accesso e degli aspetti da testare.

Descrivere una macchina UML sequenziale precisando le caratteristiche per il suo funzionamento. Portare un semplice esempio.

Dare la definizione di un macchina di stato UML. Precisare tutte le informazioni che in fase di modellazione si possono specificare in stato e transizione.

Descrivere i modelli di ciclo di vita adatti per lo sviluppo di software sicuro

Dato il seguente pseudo-codice:

```
read (a,b,c)
while(a > 0 & b > 0 & c > 0) do
    if (a>2 || (a>b & a >c))
        then foo1(a,b,c);
        else foo2(a,b,c);
    endif
    a:=a-10;
endwhile
end
```

*** Disegnare il grafo di flusso di controllo:**

*** Stabilire un criterio di copertura per ottenere il branch coverage;**

*** Usare l'MCDC per trovare un test set per la copertura della guarda dell'if.**

Determinare il grafo di flusso e definire criteri di copertura per le condizioni per il seguente programma

```
read (x,y)
a:= x; b:= y;
while a<>b do
    if a > b the a:= a-b;
    else b:= b-a;
    endif
endwhile
end
```

Data la seguente FSM

$S = \{s_1, s_2, s_3, s_4\}$, $I = \{a, b\}$, $O = \{0, 1\}$, $T = \{(s_1, a, 1, s_4), (s_2, b, 0, s_3), (s_3, a, 1, s_4), (s_4, a, 0, s_2)\}$
darne la rappresentazione grafica e quella in state pattern.

Dare la definizione di test set affidabile, valido e ideale.

Data la seguente funzione raddoppia che restituisce $2x$ per input positivi ed x altrimenti:

```
int raddoppia (int x) {  
    if (x>0) return 2+x  
    else return x;  
}
```

classifica i seguenti test set in affidabili, validi e ideali

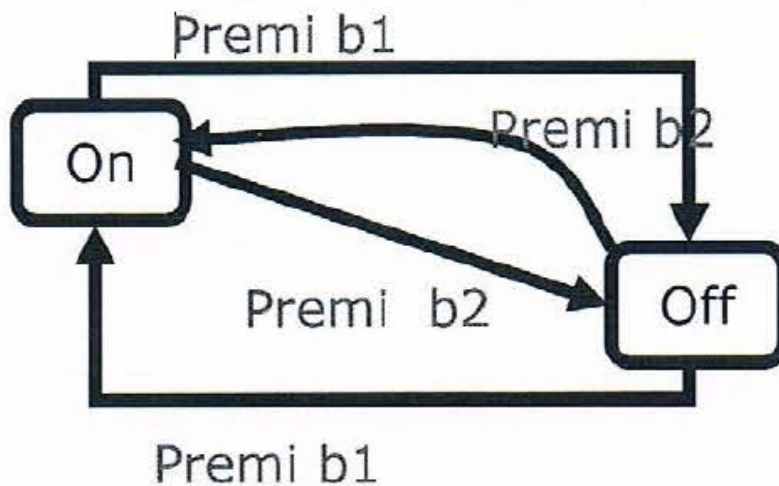
$T_1 = \{\{0\}, \{0,2\}\}$; $T_2 = \{\{0,1\}, \{2,3\}\}$; $T_3 = \{t \mid t \text{ numero pari}\}$; $T_4 = \{t \mid t \text{ numero dispari}\}$

Determinare i casi di test

1. per l'intera decisione e
2. per le singole condizioni

nella seguente espressione logica: $((x > 1) \& (y \neq 0)) \parallel (z < 0)$

Data la seguente FSM, dare la sua rappresentazione in state pattern



Dare la definizione di test set affidabile, valido e ideale. Dato il seguente programma:

```
program raddoppia(input, output)  
    var x,y: integer;  
    begin  
        read(x)  
        y:= x*x;  
        write(y)  
    end
```

classificare i seguenti test set in affidabili, validi ed ideali.

$T1 = \{\{0\}, \{0,2\}\}$; $T2 = \{\{0,1\}, \{2,3\}\}$; $T3 = \{t | t > 3\}$; $T4 = \{t | t \text{ numero dispari}\}$

Data la macchina di stato M avente

$S = (s_0, s_1, s_2)$; $I = (a, b, c)$; $0 = (0, 1)$;

$T = (\langle s_0, a, 1, s_1 \rangle, \langle s_1, a, 0, s_0 \rangle, \langle s_1, b, 0, s_2 \rangle, \langle s_2, a, 0, s_2 \rangle, \langle s_2, c, 1, s_0 \rangle)$

rappresentare M mediante state pattern.

Dato il seguente programma, disegnare il grafo di flusso e determinare una test suite per le decisioni e per le condizioni mediante MCDC:

```
foo (int x, int y ) {
    if (x < 2 && (y == 5 || x == 0)) {
        z++;
    } else {z--;}
}
```

Determinare il grafo di flusso e definire criteri di copertura per le condizioni per il seguente programma

```
read (x,y)
a:= x; b:= y;
while a <> b do
    if a > b then a:= a-b;
    else b:= b - a;
endif
endwhile
end
```

Dare la definizione di test set ideale, valido e affidabile. Computare un test set ideale per il seguente programma che dovrebbe restituire il valore della funzione $f(x,y) = x*(y*10)$ per valori x,y interi, crescenti e maggiori di 1, 0 altrimenti, ma che contiene un errore:

```
int foo (int x, int y ) {
    if (x > 1 & y > x) return x*(y+10)
    else return 0;
}
```

Determinare il grafo di flusso e definire criteri di copertura per le condizioni per il seguente programma

```
read (x,y)
a:= x; b:= y;
while a <> b do
    if a > b then a:= a-b;
    else b:= b - a;
```

```
endif
endwhile
end
```

Dare la definizione di test set valido, affidabile ed ideale. Dato il seguente programma che dovrebbe restituire il valore della funzione $foo(x,y)=2(x+y+z)$ per ogni valore x,y interi:

```
int foo (int x, int y, int z) {
    if (x > 0 & y > x & z > y)
        return 2*x+y+z
    else return 2*(x+y+z);
}
```

fare un esempio di test set ideale.

Dare la definizione di test set valido, affidabile ed ideale. Dato il seguente programma che dovrebbe restituire il valore della funzione $foo(x,y)=2(x+y)$ per ogni valore x,y interi:

```
int foo (int x, int y ) {
    if (x > 0 & y > x) return 2*x+y else return 2*(x+y);
}
```

fare un esempio di test set ideale.

Dato il seguente programma `add(x,y)` che dovrebbe implementare la funzione $z=x+y$.

```
program add (x:integer, y:integer) begin z:= x*y; return(z); end
```

classificare i seguenti test set in affidabili, validi ed ideali. $T1 = \{\{0,0\},\{-1,-1\}\}$; $T2 = \{\{0,1\},\{2,3\}\}$; $T3 = \{t|t \text{ numero pari a } 0 \text{ compreso}\}$

Definire le pre-condizioni e le post-condizioni di un metodo Java che calcola la somma dei valori di un array di interi positivi. Descrivere come gli elementi che costituiscono questo contratto possono essere utilizzati per provare la correttezza del metodo.

Modellare tramite macchine di stato di comunicazione il comportamento di due processi writer a reader che comunicano mediante un buffer di al più lunghezza 3. Disegnare il grafo di raggiungibilità dell'equivalente FSM.