

# *2020*

## *Ingegneria del software:*

### *Soluzione compito 220620*

*Giorgio Bruno*

*Dip. Automatica e Informatica  
Politecnico di Torino  
email: giorgio.bruno@polito.it*

Quest'opera è stata rilasciata sotto la licenza Creative Commons  
Attribuzione-Non commerciale-Non opere derivate 3.0 Unported.  
Per leggere una copia della licenza visita il sito web  
<http://creativecommons.org/licenses/by-nc-nd/3.0/>



## Processo B2B 12 punti

Un'agenzia di intermediazione gestisce campagne d'acquisto mediante un processo B2B e interagisce con clienti e fornitori.

Nel sistema informativo dell'agenzia sono registrati clienti, fornitori, tipi (di prodotti), legami tra fornitori e tipi.

Le campagne provengono da collaborazioni non considerate; una campagna ha due scadenze, d1 e d2, si riferisce ad un tipo ed è associata ad un gestore (ruolo di staff).

Entro la scadenza d1 i clienti possono inviare adesioni ad una campagna. Se alla scadenza d1 ci sono meno di tre adesioni, il gestore annulla la campagna e il processo comunica ai clienti che le loro adesioni sono respinte. In caso contrario il gestore associa la campagna a n ( $\geq 3$ ) fornitori interessati al tipo relativo alla campagna (1). Il processo invia la campagna ai fornitori ciascuno dei quali può rispondere con un preventivo entro la scadenza d2.

Se alla scadenza d2 il numero dei preventivi è minore di 2, il gestore chiude la campagna: il processo comunica ai clienti che le loro adesioni sono respinte e informa l'eventuale fornitore che il suo preventivo è respinto.

Se il numero dei preventivi è  $\geq 2$  il gestore conferma la campagna e il processo manda ai clienti le loro adesioni con i preventivi.

# *B2B*

I clienti rispondono con un voto relativo al preventivo scelto. Il gestore accetta il preventivo con il maggior numero di voti e respinge gli altri. Il processo informa i fornitori con le interazioni preventivo accettato o preventivo respinto; inoltre segnala ai clienti il preventivo accettato.

(1) Si esprima il vincolo con un invariante.

Nella risposta scrivete i tre modelli con le tracce seguenti (da copiare e completare).

## **Collaborazioni**

Cliente	processo	processo	Fornitore
Adesione with Campagna, before d1 ->		Campagna ->	

## **Modello informativo + invariante**

Fornitore	Tipo
-----------	------

## **Processo**

Cliente	Gestore	Fornitore
Adesione --- (Campagna)		

# *Collaborazioni*

```
Cliente    processo
Adesione with Campagna, before d1 ->
alt
    <- ar, Adesione
---
    <- ap, Adesione - Preventivo
Voto ref Preventivo ->
    <- pa, Preventivo
```

ar, ap: adesione respinta, adesione con preventivi  
pa, pr: preventivo accettato, respinto

```
processo Fornitore
Campagna ->
opt before campagna.d2
    <- Preventivo
alt
    pa, Preventivo ->
    pr, Preventivo ->
```

Fornitore	Tipo <u>1</u>	Campagna	<u>1</u>	Gestore
Cliente 1	Adesione	1 Campagna		0,n(>=3) Fornitore
		Campagna <u>1</u>	Preventivo	<u>1</u> Fornitore
	Adesione <u>1</u>	Voto	<u>1</u>	Preventivo

Campagna: Date d1, Date d2.

Invariante (altre soluzioni possibili):

campagna.fornitori in campagna.tipo.fornitori.

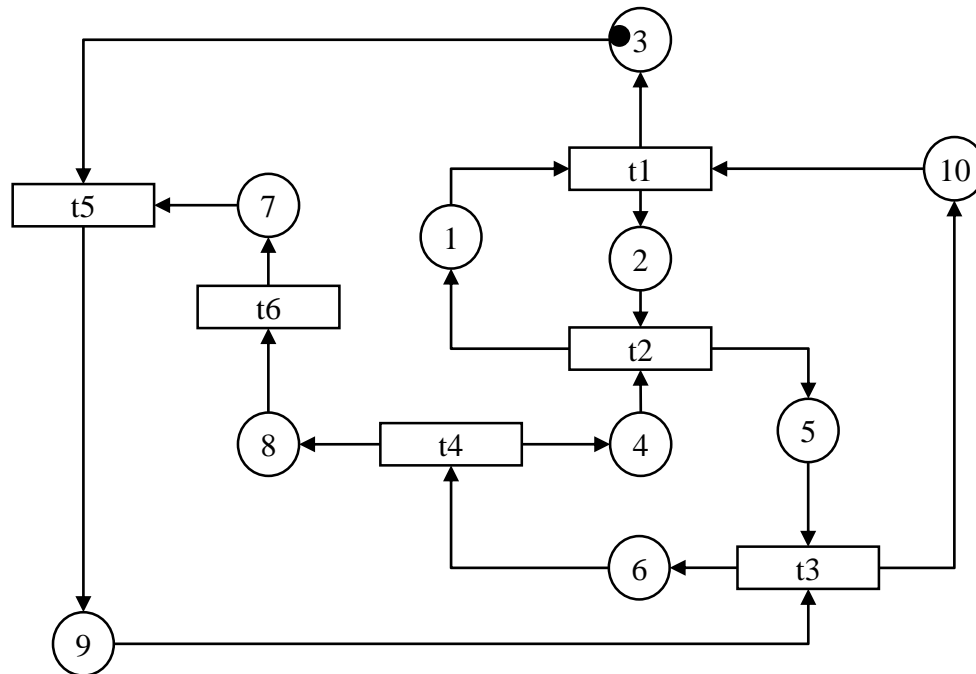
# Processo

Cliente	Gestore	Fornitore
Adesione --- (Campagna) at d1 ->	<b>bivio 1</b>	
ar	<- annulla	
	associaF ->	Campagna/(Preventivo)
	<b>bivio 2</b>	<-
ar	<- chiude ->	pr
ap/Voto	<- conferma	
->	<b>decide su preventivi</b>	
pa	<- accetta ->	pa
	respinge ->	pr

annulla: pre: [adesioni] < 3. chiude: pre: [preventivi] < 2.

associaF: fornitori def. accetta: pre: max([voti]).

## *Es. PN*



Si analizzi (senza modificarla) la rete data, che ha un token iniziale in P3, per rispondere alle domande seguenti.

# *Domande*

- 1) Quanti sono i circuiti?
- 2) Quanti e quali sono i circuiti che comprendono il posto 6?
- 3) Quali sono i circuiti di base?
- 4) Ci sono circuiti privi di token? Se sì, quanti sono?
- 5) Quanti e quali sono i circuiti privi di token che comprendono il posto 5?
- 6) Qual è il numero minimo di posti marcati inizialmente che rendono live e safe la rete e quali sono (incluso il posto P3)?
- 7) Qual è il tempo ciclo della rete con la marcatura precedente assumendo che tutte le transizioni abbiano durata unitaria tranne  $t_6$  che ha durata 2?
- 8) Con quale circuito si ottiene?



# *Risposte*

1) 6

2) 3: [1, 3, 9, 6, 4], [4, 5, 6], [6, 8, 7, 9]

3) [1, 2], [3, 9, 10], [4, 5, 6]

4) 4

5) 2: [2, 5, 10] [4, 5, 6]

6) 3: 2, 3, 6

7) 5

8) [6, 8, 7, 9]

## *Es. WBT*

Si analizzi il metodo seguente e si risponda alle domande.

```
public static int f1(boolean x, boolean y, boolean z,  
boolean u, boolean v, boolean w, int a, int b, int c, int d){  
    int k = 0; int j = 1;  
    if ((x||y) && z) {           //1  
        if (b > c) j = 5;        //2  
        else k += j;  
        if (c > d) return -1; //3  
    } else {  
        if (a > b) k = 10;        //4  
        else k = 20;  
    }  
    if ((u && v) || w)           //5  
        k += 1;  
    else k += -1;  
    return k;  
}
```

# *Domande*

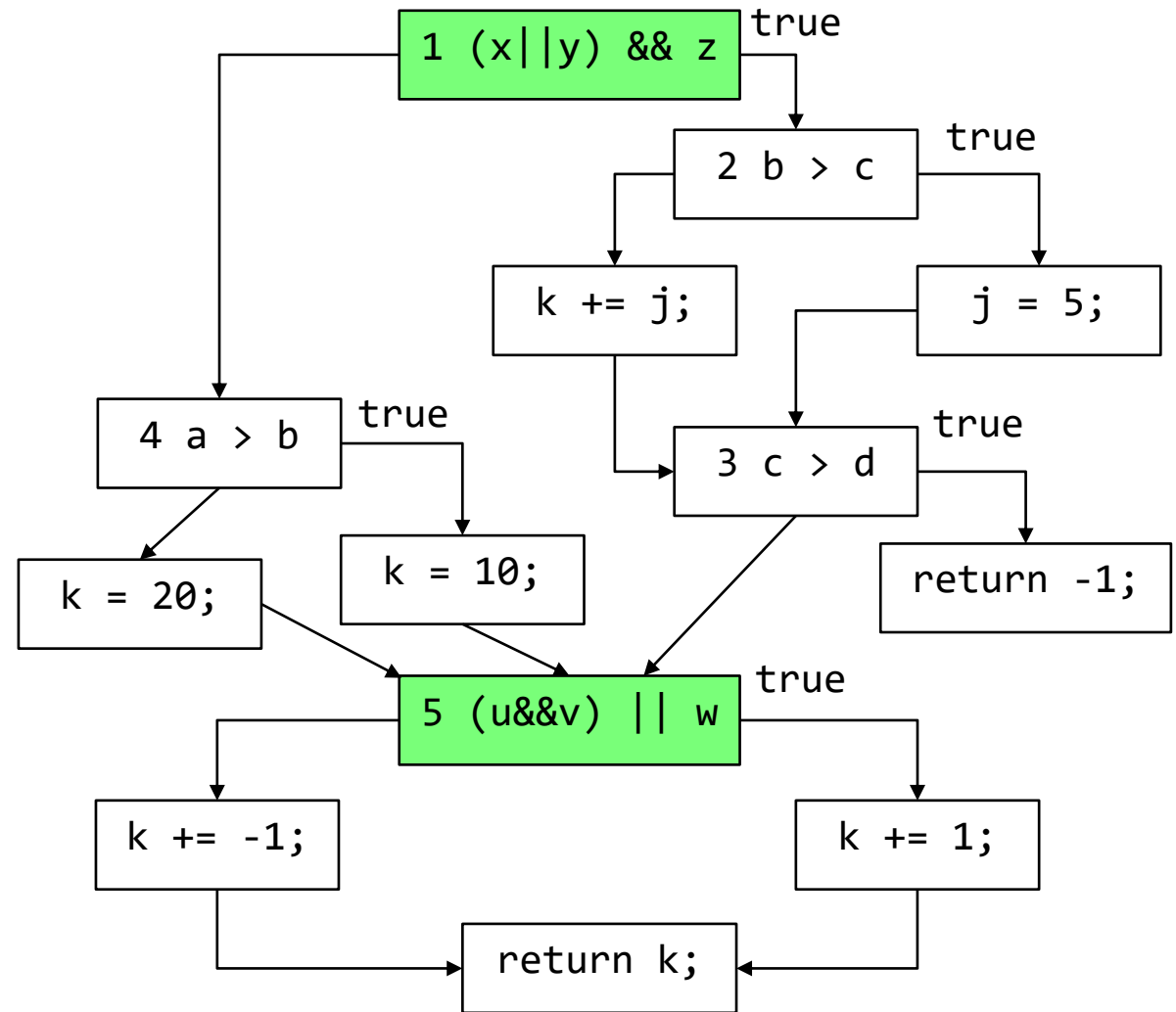
- 1) Qual è il numero minimo di test per la copertura dei nodi?
- 2) Qual è il numero minimo di test per la copertura dei link?
- 3) Qual è il numero minimo di test per la copertura dei percorsi?
- 4) Qual è il numero minimo di test per la copertura delle condizioni multiple?
- 5) Qual è il numero minimo di test per la copertura sia dei percorsi sia delle condizioni multiple?

Si indichino i test relativi ai nodi, ai link e ai percorsi (domande 1, 2 e 3) con sequenze di condizioni vere o false. Per le altre domande si spieghi il valore.

# Flow chart

Omesso il primo blocco

```
int k = 0; int j = 1;  
if ((x||y) && z) {           //1  
    if (b > c) j = 5;        //2  
    else k += j;  
    if (c > d) return -1;    //3  
} else {  
    if (a > b) k = 10;        //4  
    else k = 20;  
}  
if ((u && v) || w)           //5  
    k += 1;  
else k += -1;  
return k;  
}
```



## *Risposte*

1. Servono 4 casi di test, 2 con 1T e 2 con 1F.

1T 2T 3T, 1T 2F 3F 5T, 1F 4T 5T, 1F 4F 5F.

2. Servono 4 casi di test, come sopra.

3. Servono 10 casi di test, 6 con 1T e 4 con 1F.

1T (2) 3T, 1T (2) 3F (5)

1F (4) (5).

4. Bastano 8 casi di test: ci sono due triple sequenziali (il return -1 si può evitare).

5. Servono 11 casi di test.

Casi dei percorsi. Ne serve uno in più con 1F che può aggiungersi ai 4 con 5T. In tutto 11.

Casi delle CM. Ne servono 3 in più con 1T, 2 escono con 3T; 9 entrano in 5 e ne escono 5 con 5T e 4 con 5F. In tutto 11.

## *Domande varie*

Si risponda alle domande seguenti con Vero o Falso.

1 punto per risposta corretta, -1 per risposta errata, 0 se manca la risposta.

- 1) Un processo singleton non ha nessuna istanza.
- 2) In un modello dataflow (DFD) un attore esterno non può essere collegato ad un'attività composta.
- 3) Il pattern free choice non si trova soltanto nelle reti di categoria free choice.
- 4) Il termine bounded context indica un qualsiasi sottoinsieme di un domain model.

# *Risposte*

- 1) Falso
- 2) Falso
- 3) Vero
- 4) Falso

Un processo singleton ha una sola istanza.

Un bounded context non è un qualsiasi sottoinsieme di un domain model ma è un sottoinsieme costituito da classi strettamente correlate e con pochi collegamenti con altri bounded context.