

2022

Soluzione compito 120922

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/>



Esercizio 1, 12 punti (4 per modello)

Il processo B2B di un'azienda di IngSw serve le richieste di supporto (RichiestaS) provenienti dai committenti utilizzando le competenze di vari studi (professionali). Nel sistema informativo sono registrati committenti, studi, le aree trattate dagli studi. Ai gestori (ruolo di staff) fanno capo le aree: ogni area si riferisce ad un gestore. Ricevuta una richiestaS relativa ad un'area il gestore può considerarla completa o incompleta.

Nel secondo caso il gestore rimanda la richiestaS al mittente. Il mittente rinvia la richiestaS modificata e il gestore può considerarla completa o incompleta. Se è incompleta la rimanda al mittente che risponde come prima e così via.

Se considera la richiestaS completa il gestore la invia al committente e la associa ad almeno 5 studi che trattano l'area della richiestaS (1); il processo invia la richiestaS agli studi associati alla richiestaS.

Ciascuno studio può inviare un supporto oppure una richiesta di chiarimento alla quale il gestore risponde con un chiarimento; ricevuto il chiarimento lo studio invia un supporto. Ricevuti tutti i supporti il processo mediante un riduttore emette la richiestaS e poi la invia al mittente insieme con i supporti. Il mittente indica il supporto accettato. Il processo (mediante un mapping relativo alla richiestaS) permette al gestore di confermare il supporto accettato e di respingere gli altri; il processo informa gli studi dei loro supporti (accettato o respinto).

(1) Si esprima il vincolo con un invariante.

Committente Processo

RichiestaS ref Area ->

loop

alt

rc, RichiestaS <-
break

ri, RichiestaS <-
rm, RichiestaS ->

rs, RichiestaS - Supporto <-
sa, Supporto ->

rc, ri, rm, rs: r. completa, incompleta, modificata, con supporti
sa, sr = supporto accettato, respinto

Modello delle collaborazioni

Processo

Studio

-> RichiestaS

opt

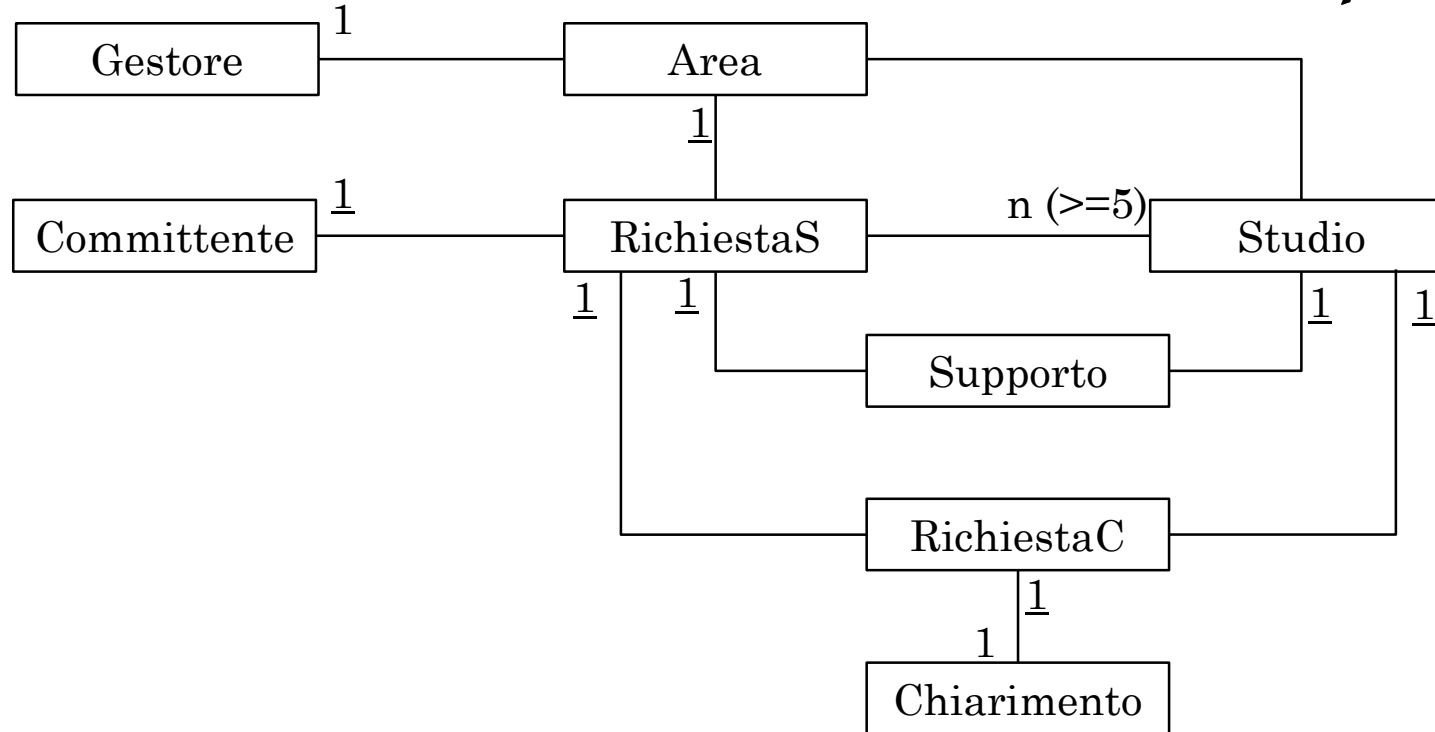
<- RichiestaC
-> Chiarimento

<- Supporto

alt

-> sa, Supporto
-> sr, Supporto

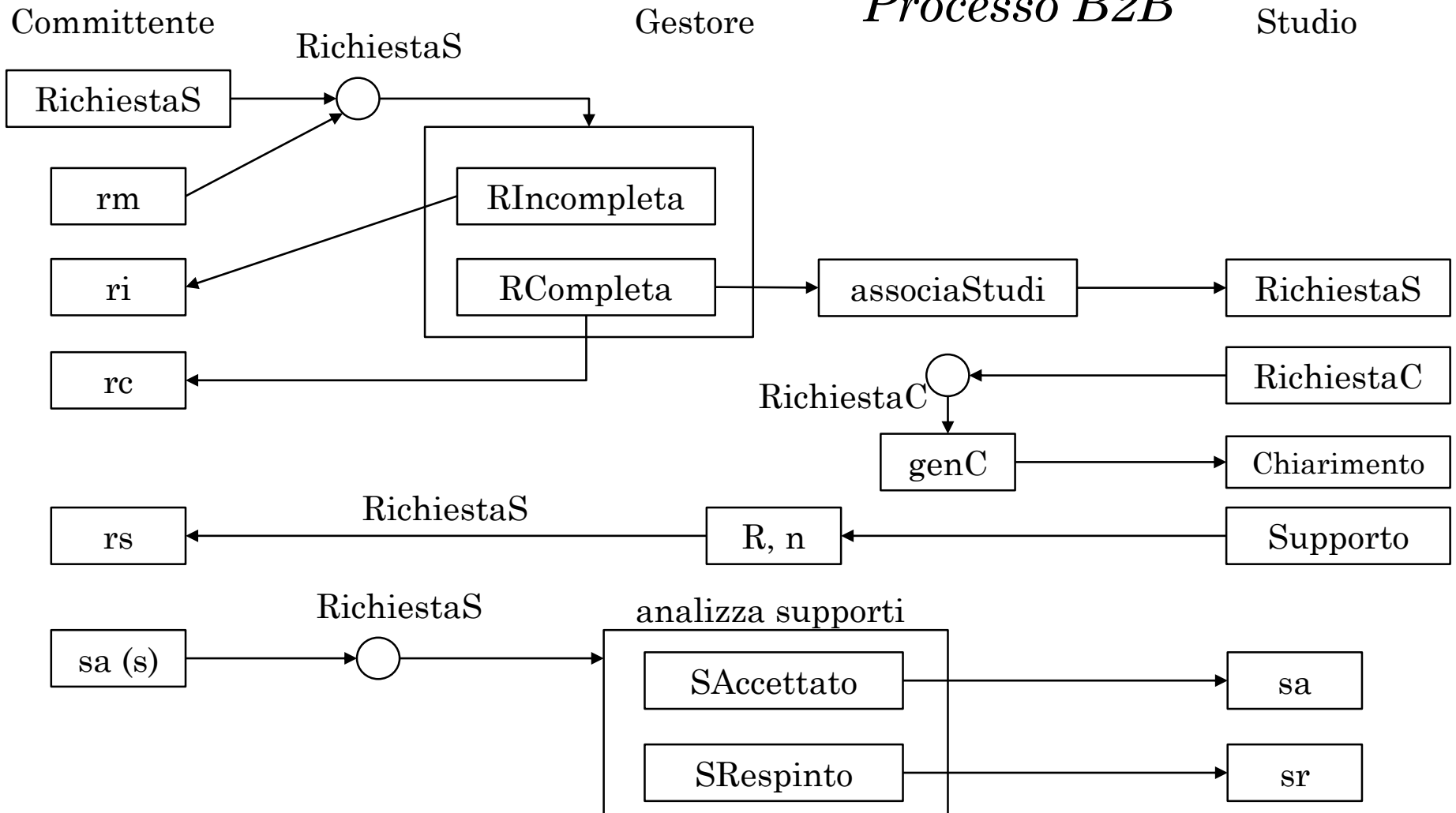
Modello informativo



Attributi: Supporto: stato (accettato).

Invariante: richiestaS.studi in richiestaS.area.studi
(ci sono altre soluzioni)

Processo B2B

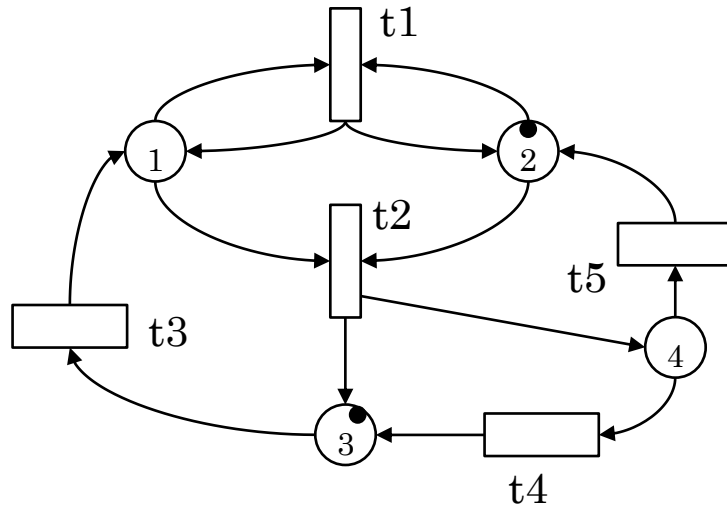


associaStudi: studi.def. genC: new Chiarimento.
 n = [richiestaS.studi]
 SAccettato: pre: stato == accettato.

Note: posti embedded

Es. 2 PN 8 punti

Si analizzi (senza modificarla) la rete data, che ha due token iniziali, uno nel posto 2 e l'altro nel posto 3, per rispondere alle domande.



- 1) Che tipo di rete è? 2) Ci sono sifoni con due o tre posti, se sì quali?
- 3) Ci sono trappole con due o tre posti, se sì quali? 4) La rete è live o no, e perché?
- 5) La rete ha un deadlock, se sì in quale marcatura??
- 6) La rete è unbounded, se sì in quali posti? 7) La rete è safe, se sì per quale motivo?
- 8) Quali sono le marcature raggiungibili da M_0 con 2 scatti delle transizioni abilitate in M_0 ? Si indichino anche le transizioni.

1) EFC in p_1, p_2 .

2) Sì: un sifone con due posti e due con tre: $[2, 4]$, $[1, 3, 4]$, $[2, 3, 4]$.

3) Sì: una trappola con due posti e una con tre: $[1, 3]$, $[1, 2, 3]$.

4) No: i sifoni $[2, 4]$ e $[2, 3, 4]$ non contengono trappole.

5) La rete ha un deadlock nella marcatura $[2, 0, 0, 0]$.

6) La rete è bounded: il n. dei token è 2. Le tr. t_1 e t_2 sono di tipo join/fork (2,2); le altre tr. sono passanti.

7) No perché ci sono posti, come p_1 , che possono contenere 2 token.

8) $[1, 1, 0, 0]$ con t_3 e t_1 , $[0, 0, 1, 1]$ con t_3 e t_2 .

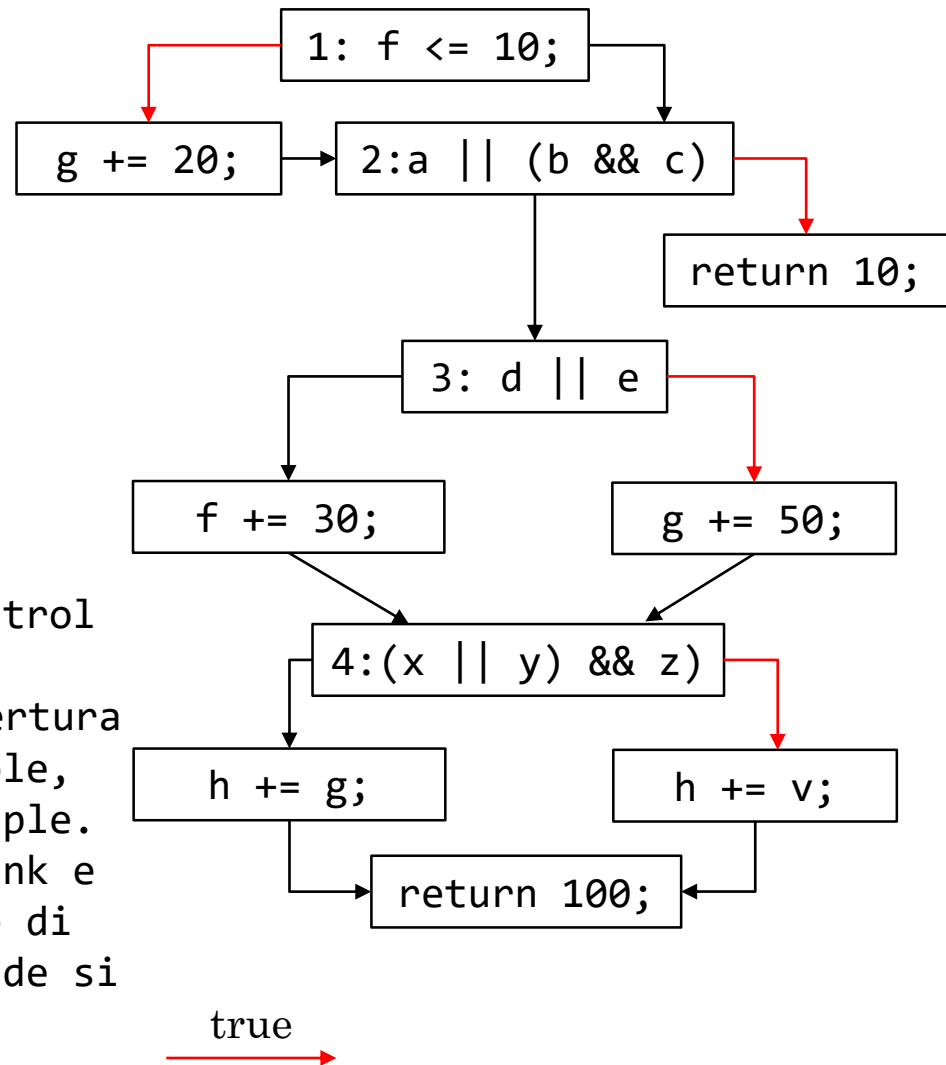
```
public static int wbt120922(boolean a,
boolean b, boolean c, boolean d, boolean
e, boolean x, boolean y, boolean z,
int f, int g, int h){
```

```
    if (f <= 10) g += 20;           //1
    if (a || (b && c)) return 10;    //2
    if (d || e) g += 50;           //3
    else f += 30;
    if ((x || y) && z) h += f;       //4
    else h += g;
    return 100;
}
```

Control flow graph / 3 punti

Per il metodo seguente, si definisca il control flow graph e si risponda alle domande: qual è il numero minimo di test per la copertura di: nodi, link, percorsi, condizioni multiple, 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. Una risposta senza spiegazione vale 0 punti.

Es. 3 - WBT (8 punti)



Nodi: 3;

1T 2T; 1F 2F 3T 4T; 1F 2F 3F 4F.

Link: come i nodi

Percorsi: 10, 5 con 1T e 5 con 1F.

1T 2T; 1T 2F 3 4; 1F 2T; 1F 2F 3 4.

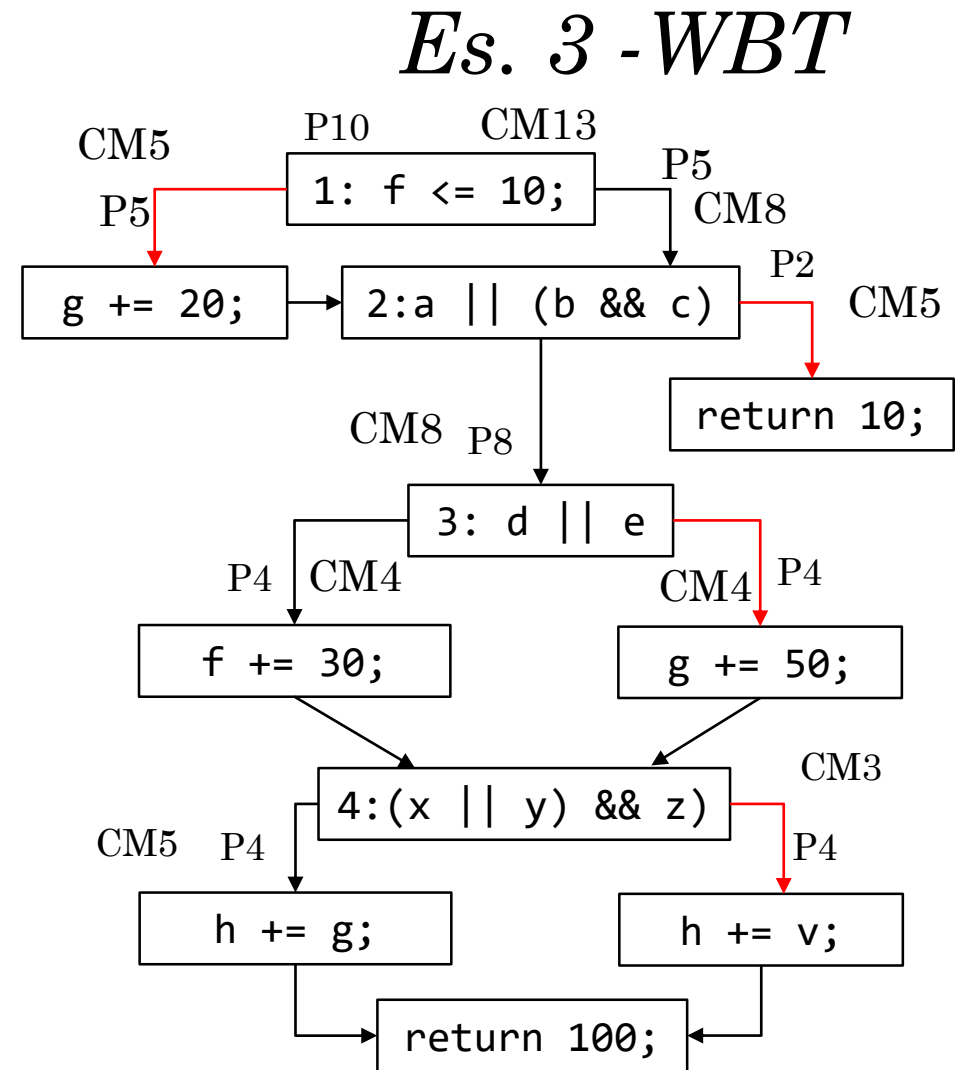
CM: 13. 5 casi escono con 2T; ne servono altri 5 per averne 8 in 4.

Tutti i criteri: **14**.

Per i percorsi servono 3 casi con 2T e 1 caso con 4F; in tutto 14 casi.

Per le CM serve un caso con 4T; in tutto 14 casi.

Nelle etichette del grafo, i percorsi sono preceduti da P e le CM da CM.



true
→

Es. 4 - Domande

Si risponda alle domande seguenti con Vero o Falso.

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

- 1) Una rete FC (Free Choice) che non è fortemente connessa non è necessariamente unbounded.
- 2) Il pattern composite non contiene alcuna relazione ricorsiva.
- 3) In un progetto Git l'oggetto master non punta all'ultimo commit del ramo principale.
- 4) Nei processi BPMN i cerchi (con caratteristiche anche diverse) sono usati esclusivamente per rappresentare inizio e fine del processo.

Risposte

- 1) V
- 2) F
- 3) F
- 4) F