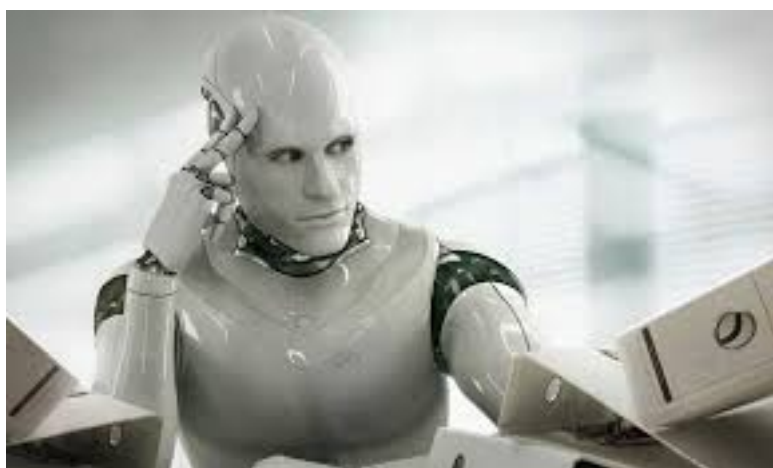


MATh.en.JEANS

# Catastrofe in cabina

Liceo Scientifico "E. Curiel"  
Padova

1 maggio 2018



## Sommario

Nel presente documento abbiamo affrontato il secondo dei due problemi che sono stati proposti all'interno del progetto MATH.en.JEANS: Catastrofe in Cabina!

Il problema richiedeva di verificare la possibilità di costruire delle sequenze di codice al fine di riprogrammare dei computer di bordo. Come è apparso chiaro fin da subito, il problema ha a che fare con la teoria dei grafi.

Gli autori ringraziano per il supporto e gli utili consigli il prof. Zanardo del Dipartimento di Matematica "Tullio Levi Civita" dell'Università degli Studi di Padova.

# Indice

<b>1</b>	<b>Catastrofe in Cabina!</b>	<b>4</b>
<b>2</b>	<b>La soluzione del primo quesito</b>	<b>10</b>
<b>3</b>	<b>La soluzione del secondo quesito</b>	<b>11</b>
3.1	Soluzione sala 1 . . . . .	11
3.2	Soluzione sala 2 . . . . .	12
3.3	Soluzione sala 3 . . . . .	14
3.4	Soluzione sala 4 . . . . .	16
3.5	Soluzione sala 5 . . . . .	17
3.6	Soluzione sala 6 . . . . .	19
<b>4</b>	<b>La soluzione del terzo quesito</b>	<b>22</b>
<b>5</b>	<b>La soluzione del quarto quesito</b>	<b>23</b>
<b>6</b>	<b>La soluzione del quinto quesito</b>	<b>23</b>
<b>7</b>	<b>La soluzione del sesto quesito</b>	<b>23</b>

# 1 Catastrofe in Cabina!

Siete in vacanza a bordo di una navicella spaziale. Quando state sorseggiando un bicchiere di Jurancon, il petit Manseng passito, un gran millesimato del 2017, la navicella comincia a vibrare. Vi precipitate nella cabina di comando dove trovate un caos completo. I robot sono completamente disorganizzati. Dopo una rapida analisi della situazione, vi rendete conto che i robot sono stati colpiti da amnesia (tutti hanno perso la memoria), conseguenza del vento solare della settimana precedente.

Generalmente i robot hanno tutti un numero identificativo, e quello che ha il numero più piccolo è il capitano. Ora i robot, non avendo più un numero identificativo, non sanno più chi sia il capitano.

Il vostro obiettivo è scrivere un programma che permetta ai robot di eleggere un capitano.

I robot comunicano tra loro attraverso dei canali di comunicazione. Un canale di comunicazione collega sempre due robot e permette loro di comunicare secondo un protocollo che definiremo più avanti.

In una stanza c'è un numero finito di robot e di canali. Il numero di robot e la configurazione di canali non può variare nel tempo. Nella Figura 1 troviamo un esempio di configurazione di canali. In questa configurazione i cerchi rappresentano i robot e ogni segmento un canale di comunicazione.

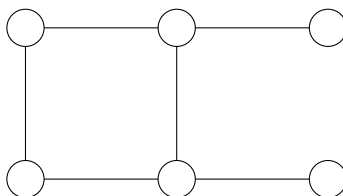


FIGURA 1: una configurazione di robot

Ci sono più tipi di robot:

1. robot con 1–vicino; sono collegati a 1 solo robot con un canale di comunicazione;
2. robot con 2–vicini; sono collegati a 2 robot con due canali di comunicazione;
3. etc etc

Tutti i robot con  $i$ –vicini sono programmati dallo stesso programma che si chiama  $P_i$ .

Per esempio, nella Figura 1, i due robot a sinistra sono di tipo 2 e sono programmati con il programma  $P_2$ . I due robot in mezzo sono di tipo 3 e sono programmati con il programma  $P_3$  e infine i due a destra sono di tipo 1 e sono programmati con il programma  $P_1$ .

Per questa stanza il vostro obiettivo è descrivere i tre programmi  $P_1$ ,  $P_2$  e  $P_3$ .

Ogni robot possiede uno stato rappresentato da un testo che potete modificare con il vostro programma. Voi potete, con il vostro programma, scegliere gli stati che desiderate. Ci sono tuttavia tre stati predefiniti: 'iniziale', 'capitano' e 'operaio'. Lo stato 'iniziale' è quello in cui si trova ogni robot alla partenza. Quando un robot si trova nello stato 'capitano', vuol dire che il robot è stato scelto come capitano. Quando un robot si trova nello stato 'operaio', vuol dire che il robot è stato scelto come operaio.

L'obiettivo del vostro programma è far sì che un solo robot sia nello stato 'capitano' e tutti gli altri siano nello stato 'operaio'.

Per arrivare a questo, voi utilizzerete i programmi  $P_i$  per cambiare lo stato dei vostri robot. All'avvio, ogni robot esegue i seguenti passi:

1. inizializza il valore del suo stato come 'iniziale';
2. numera i canali di comunicazione da 1 a  $d$ , dove  $d$  è il numero dei suoi vicini. La numerazione è arbitraria e può cambiare da una situazione di partenza ad un'altra;
3. esegue il programma  $P_d$ .

I programmi dei robot possono essere eseguiti in parallelo, ma un robot può comunicare con un solo robot alla volta.

Un programma è una tabella. Ciascuna riga descrive una evoluzione possibile dello stato di un robot in seguito a uno scambio di messaggi. I programmi  $P_1$ ,  $P_2$  e  $P_3$  della Tavola 1 sono esempi di programmi. Queste tabelle contengono cinque colonne:

- la colonna 'stato' descrive lo stato del robot;
- la colonna 'canale' dà il canale di comunicazione;
- la colonna 'emissione' contiene i messaggi che il robot invia dopo la concertazione con il suo omologo;
- la colonna 'ricezione' contiene i messaggi che il robot riceve dopo la concertazione con il suo omologo;

- la colonna 'nuovo stato' è lo stato che assumerà il robot in seguito alla comunicazione.

Programma  $P_1$

stato	canale	emissione	ricezione	nuovo stato
bloccato	1	divento capitano	divento operaio	capitano
bloccato	1	divento operaio	divento capitano	operaio
bloccato	1	divento operaio	divento operaio	operaio
bloccato	1	aspetto	aspetto	bloccato

Programma  $P_2$

stato	canale	emissione	ricezione	nuovo stato
-------	--------	-----------	-----------	-------------

Programma  $P_3$

stato	canale	emissione	ricezione	nuovo stato
in attesa	3	divento capitano	divento operaio	capitano
in attesa	3	divento operaio	divento capitano	operaio
in attesa	3	aspetto	divento operaio	in attesa
in attesa	1	divento capitano	divento operaio	capitano
addormentato	3	divento operaio	divento capitano	operaio

TAVOLA 1 – I programmi  $P_1$ ,  $P_2$  e  $P_3$

I robot comunicano due a due, gli uni dopo gli altri. Quando due robot comunicano, si accordano e ciascuno sceglie una riga nel proprio programma, in modo che le due righe scelte siano compatibili. Per un robot  $r$  una riga è compatibile con quella del robot  $s$  quando

- lo stato del robot è nella colonna 'stato';
- il canale di comunicazione scelto è nella colonna 'canale';
- il messaggio inviato dal robot è nella colonna 'emissione';
- il messaggio ricevuto è nella colonna 'ricezione' e coincide con quello inviato dal robot  $s$  (cioè il messaggio nella colonna 'emissione' nella riga scelta dal robot  $s$ ).

Se i robot  $r$  e  $s$  non possono trovare delle righe compatibili, allora la comunicazione si interrompe e non succede niente. Se i robot  $r$  e  $s$  trovano più righe compatibili, allora ne scelgono una in modo arbitrario.

Consideriamo l'esecuzione del programma quando un robot  $r$  ha deciso di comunicare col robot  $s$ . Supporremo che:

- il robot  $r$  ha 3 vicini e usa il programma  $P_3$  della TAVOLA 1;
- il robot  $s$  ha 1 vicino e usa il programma  $P_1$  della Tavola 1;
- il robot  $s$  è situato sul canale 3 del robot  $r$ ;
- il robot  $r$  è situato sul canale 1 del robot  $s$ ;
- il robot  $r$  è nello stato 'in attesa';
- il robot  $s$  è nello stato 'bloccato'.

Per comunicare, il robot  $r$  va prima a selezionare tutte le righe del suo programma in cui la casella della prima colonna contiene 'in attesa' e la casella della seconda colonna contiene il canale 3. Ecco le righe che ha selezionato:

stato	canale	emissione	ricezione	nuovo stato
in attesa	3	divento capitano	divento operaio	capitano
in attesa	3	divento operaio	divento capitano	operaio
in attesa	3	aspetto	divento operaio	in attesa

Il robot  $s$ , quando tocca a lui, seleziona tutte le righe del suo programma in cui la casella della prima colonna contiene 'bloccato' e la casella della seconda colonna contiene il canale 1. Ecco le righe che ha selezionato:

stato	canale	emissione	ricezione	nuovo stato
bloccato	1	divento capitano	divento operaio	capitano
bloccato	1	divento operaio	divento capitano	operaio
bloccato	1	divento operaio	divento operaio	operaio
bloccato	1	aspetto	aspetto	bloccato

A questo punto, i due robot scelgono di comune accordo dei messaggi da scambiarsi: cioè i robot  $r$  e  $s$  scelgono ciascuno una riga, tra quelle che sono rimaste, in modo che il messaggio della casella 'ricezione' coincida con quello della casella 'emissione' dell'altro robot. Per esempio, se il robot  $r$  ha scelto una riga in cui l'emissione è 'voglio essere capitano' e la ricezione

è 'voglio essere operaio', allora il robot  $s$  dovrà scegliere una riga in cui la sua emissione sia 'voglio essere operaio', e la sua ricezione sia 'voglio essere capitano'.

Queste sono tutte le associazioni possibili:

Associazione 1

robot	stato	canale	emissione	ricezione	nuovo stato
r	in attesa	3	divento capitano	divento operaio	capitano
s	bloccato	1	divento operaio	divento capitano	operaio

Associazione 2

robot	stato	canale	emissione	ricezione	nuovo stato
r	in attesa	3	divento operaio	divento capitano	operaio
s	bloccato	1	divento capitano	divento operaio	capitano

I robot scelgono ora, in modo arbitrario, una associazione possibile. Supponiamo che abbiano scelto la seconda. Quindi il robot  $r$  passa allo stato 'operaio' e il robot  $s$  passa allo stato 'capitano'.

I programmi continuano ad essere eseguiti fino a quando ogni robot ha raggiunto lo stato 'capitano' o lo stato 'operaio'.

L'organizzatore del viaggio, che ha gettato la spugna, molto tempo dopo tuttavia propone una soluzione per la sala seguente:



Questo programma è descritto nella TAVOLA 2.

Programma  $P_1$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento capitano	divento operaio	capitano
iniziale	1	divento operaio	divento capitano	operaio
iniziale	1	divento operaio	sono capitano	operaio
iniziale	1	divento operaio	sono operaio	operaio

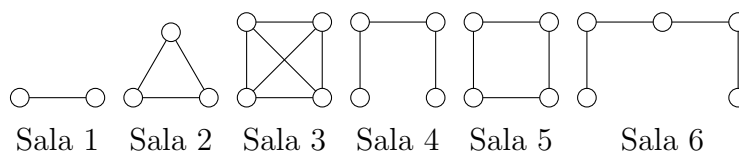


Programma  $P_2$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento capitano	divento operaio	capitano
iniziale	1	divento operaio	divento capitano	operaio
iniziale	2	divento capitano	divento operaio	capitano
iniziale	2	divento operaio	divento capitano	operaio
capitano	1	sono capitano	divento operaio	capitano
capitano	2	sono capitano	divento operaio	capitano
operaio	1	sono operaio	divento operaio	capitano
operaio	2	sono operaio	divento operaio	capitano

TAVOLA 2 – I programmi  $P_1$  e  $P_2$  dell'organizzatore

1. Il programma dell'organizzatore è valido?
2. Quali programmi si possono scrivere affinché i robot delle sale seguenti eleggano un capitano?



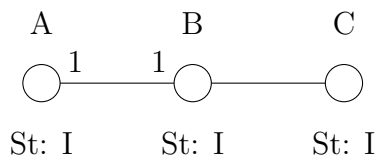
3. Ci sono delle sale su cui non sapete tutto: per esempio, voi non conoscete il numero dei vertici-robot, ma sapete che tutti i robot sono collegati tra loro. Potete scrivere un programma che determini un capitano?
4. C'è una sala di cui non conoscete né il numero dei vertici-robot, né la configurazione dei canali di comunicazione. Potete scrivere dei programmi che determinino un capitano?
5. C'è una sala dove conoscete il numero dei vertici-robot ma non la configurazione dei canali di comunicazione. Scrivete un programma che determini un capitano.
6. Esistono dei programmi informatici che in entrata (input) hanno sale e restituiscono (output) un programma per l'elezione di un capitano?

## 2 La soluzione del primo quesito

Nella risoluzione dei quesiti abbiamo inteso che un programma è funzionante se è funzionante in ogni situazione, indipendentemente dalle scelte, anche casuali, che i robot possono compiere.

Osserviamo che in questa configurazione è presente un solo robot di tipo due (quello centrale) e due robot di tipo uno e che la configurazione dei robot è simmetrica rispetto al robot centrale. Poiché un robot può comunicare contemporaneamente con un solo altro robot, il primo robot a comunicare sarà il robot centrale e, vista la simmetria del problema, non è rilevante, in questo caso, con quale altro robot comunicherà (se quello di destra o quello di sinistra). Supponiamo che decida di comunicare con quello di sinistra<sup>1</sup>.

Schematizziamo la situazione:



I due robot scelgono i messaggi da scambiarsi e le associazioni possibili sono:

Associazione 1

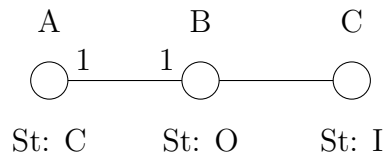
robot	stato	canale	emissione	ricezione	nuovo stato
A	iniziale	1	divento operaio	divento capitano	operaio
B	iniziale	1	divento capitano	divento operaio	capitano

Associazione 2

robot	stato	canale	emissione	ricezione	nuovo stato
A	iniziale	1	divento capitano	divento operaio	capitano
B	iniziale	1	divento operaio	divento capitano	operaio

Poiché esistono più associazioni, i robot scelgono arbitrariamente i messaggi da scambiarsi. Supponiamo venga scelta la seconda associazione. La situazione che si presenta ora è la seguente:

<sup>1</sup>Notiamo inoltre che il programma  $P_2$  contiene righe replicate per ciascun canale, per cui la scelta del canale è irrilevante.

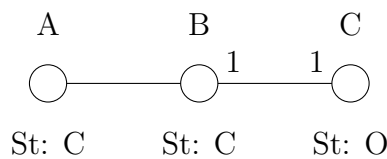


Osserviamo anzitutto che il robot  $A$  non può più comunicare (nel programma  $P_1$  non sono comprese righe con stati diversi da 'in attesa'). Pertanto ora possono comunicare tra loro solo i robot  $B$  e  $C$ . La comunicazione a questo punto può procedere secondo un'unica associazione possibile, la seguente:

Associazione

robot	stato	canale	emissione	ricezione	nuovo stato
B	operaio	1	sono operaio	divento operaio	capitano
C	iniziale	1	divento operaio	sono operaio	operaio

Al termine della comunicazione la situazione che si presenta è la seguente:

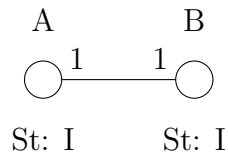


In questa condizione né il robot  $A$ , né il robot  $B$  possono comunicare e la configurazione presenta due capitani. Il programma termina e non si è trovato un unico capitano.

### 3 La soluzione del secondo quesito

#### 3.1 Soluzione sala 1

In questo caso abbiamo due robot di tipo 1 ed un unico canale.

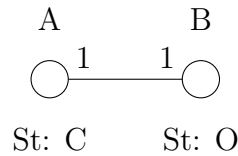


Si consideri il seguente programma:

Programma  $P_1$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento capitano	divento operaio	capitano
iniziale	1	divento operaio	divento capitano	operaio

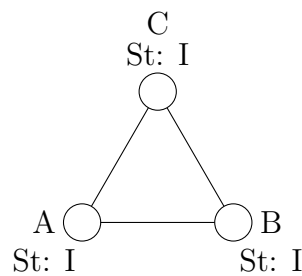
In questo caso è possibile una sola associazione (che coincide con il programma stesso, indipendentemente da quale robot sceglie la prima o la seconda riga del programma) e dopo la sua esecuzione si ottiene la seguente configurazione (o quella speculare con gli stati invertiti):



In questo caso abbiamo un unico 'capitano' ed il secondo robot è un 'operaio', il programma termina ed il problema è risolto.

### 3.2 Soluzione sala 2

Osserviamo che in questo caso (così come nel caso precedente e nel caso successivo) siamo in presenza di un grafo connesso. In questo caso siamo in presenza di tre robot di tipo 2. Rappresentiamo con una figura la situazione iniziale:



Osserviamo che la configurazione dei robot è simmetrica per cui non è vincolante sapere quale dei robot interagirà per primo (e lo potranno fare solo due dei tre robot in quanto ogni robot non può comunicare contemporaneamente con più di un altro robot). Supponiamo pertanto che i due robot che comunicheranno per primi siano il robot  $A$  ed il robot  $B$ .

Consideriamo il seguente programma  $P_2$ .

Programma  $P_2$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento operaio	divento capitano	operaio
iniziale	1	divento capitano	divento operaio	capitano
iniziale	2	divento operaio	divento capitano	operaio
iniziale	2	divento capitano	divento operaio	capitano
iniziale	1	divento operaio	sono capitano	operaio
iniziale	2	divento operaio	sono capitano	operaio
capitano	1	sono capitano	divento operaio	capitano
capitano	2	sono capitano	divento operaio	capitano

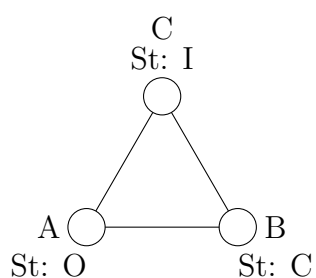
Osserviamo che il programma ha una riga identica replicata per ogni canale, per cui in realtà non è importante sapere attraverso quale canale comunicano i robot.

Il programma prevede una sola associazione possibile, la seguente (la scelta di quale dei due computer esegue una delle due righe di codice è ininfluente):

Associazione

robot	stato	canale	emissione	ricezione	nuovo stato
A	iniziale	1	divento operaio	divento capitano	operaio
B	iniziale	1	divento capitano	divento operaio	capitano

A questo punto lo stato del robot  $A$  passa a 'operaio' e lo stato del computer  $B$  passa a 'capitano'.

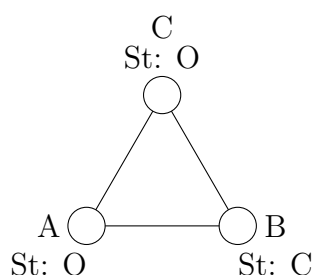


Il programma non consente al robot con stato 'operaio' di comunicare con gli altri robot. Pertanto la successiva comunicazione potrà avvenire solo tra i robot  $B$  e  $C$ . Anche in questo caso vi è una sola associazione possibile:

## Associazione

robot	stato	canale	emissione	ricezione	nuovo stato
B	capitano	1	sono capitano	divento operaio	capitano
C	iniziale	1	divento operaio	sono capitano	operaio

All'esecuzione del programma la situazione è la seguente:



Vi è pertanto un unico robot con lo stato 'capitano' ed i rimanenti robot hanno lo stato impostato su 'operaio'. Il programma termina ed il problema è risolto in quanto è stato individuato il capitano.

### 3.3 Soluzione sala 3

Facciamo anzitutto alcune osservazioni<sup>2</sup>:

1. la configurazione dei robot è rappresentata da un grafo connesso, in questo modo tutti i robot possono comunicare con qualsiasi altro robot pertanto è indifferente sapere all'inizio quali robot comunicano tra di loro;
2. è necessario un unico programma di tipo  $P_3$ ;
3. saranno presenti tante righe di codice replicate quanti sono i canali di comunicazione (in questo caso 3);
4. il programma è costruito in modo tale che i robot con stato 'operaio' siano esclusi dalla comunicazione con gli altri robot.

Alla prima esecuzione del programma almeno uno dei robot passa allo stato di 'capitano' e un altro passa allo stato di 'operaio'. Il programma

---

<sup>2</sup>Le seguenti osservazioni guideranno anche la soluzione della domanda successiva del problema.

gestisce anche il caso in cui, per qualche motivo, i due robot rimanenti non dovessero comunicare tra di loro contemporaneamente ai primi due robot.

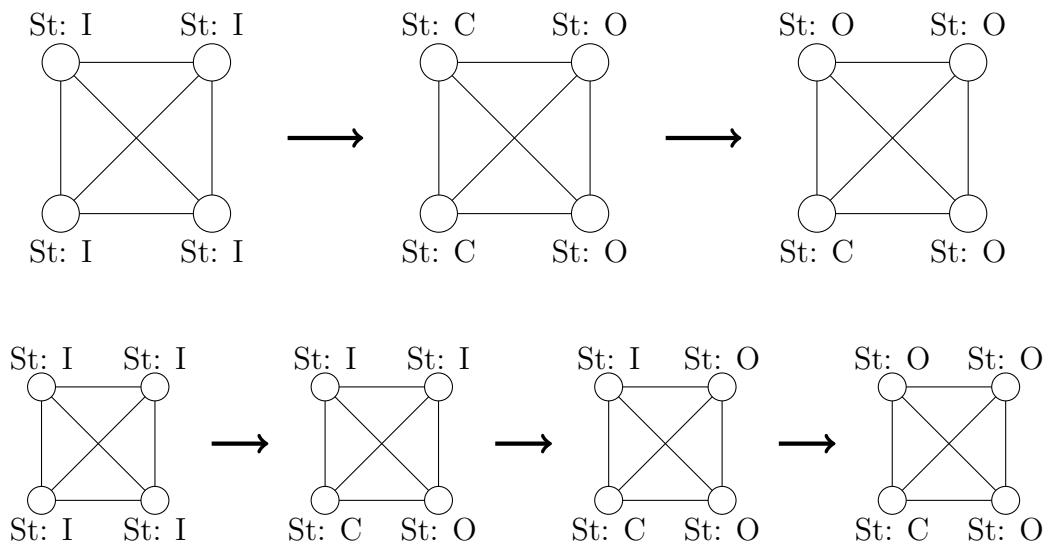
Una volta raggiunto lo stato 'operaio' il robot viene escluso dalla esecuzione del programma, per cui possono comunicare tra di loro solo robot con stato 'capitano od 'iniziale'.

La tabella contenente il programma da eseguire è la seguente:

Programma  $P_3$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento capitano	divento operaio	capitano
iniziale	1	divento operaio	divento capitano	operaio
iniziale	2	divento capitano	divento operaio	capitano
iniziale	2	divento operaio	divento capitano	operaio
iniziale	3	divento capitano	divento operaio	capitano
iniziale	3	divento operaio	divento capitano	operaio
iniziale	1	divento operaio	sono capitano	operaio
iniziale	2	divento operaio	sono capitano	operaio
iniziale	3	divento operaio	sono capitano	operaio
capitano	1	divento operaio	sono capitano	operaio
capitano	2	divento operaio	sono capitano	operaio
capitano	3	divento operaio	sono capitano	operaio
capitano	1	sono capitano	divento operaio	capitano
capitano	2	sono capitano	divento operaio	capitano
capitano	3	sono capitano	divento operaio	capitano

Rappresentiamo le varie situazioni che si possono presentare:



### 3.4 Soluzione sala 4

La sala presenta due robot di tipo 1 e due robot di tipo 2. Il programma che consente di risolvere la sala è il seguente:

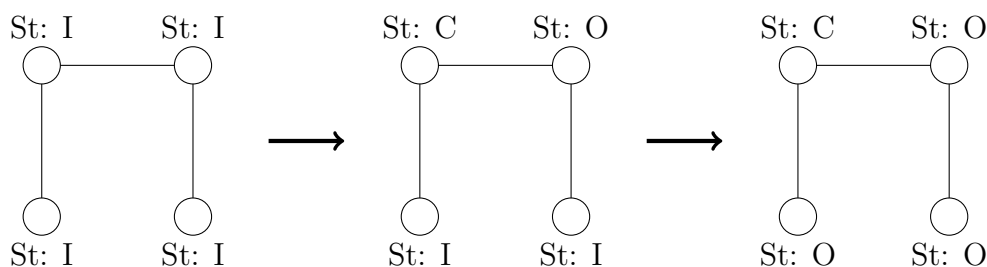
Programma  $P_1$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento operaio	sono operaio	operaio
iniziale	1	divento operaio	sono capitano	operaio

Programma  $P_2$

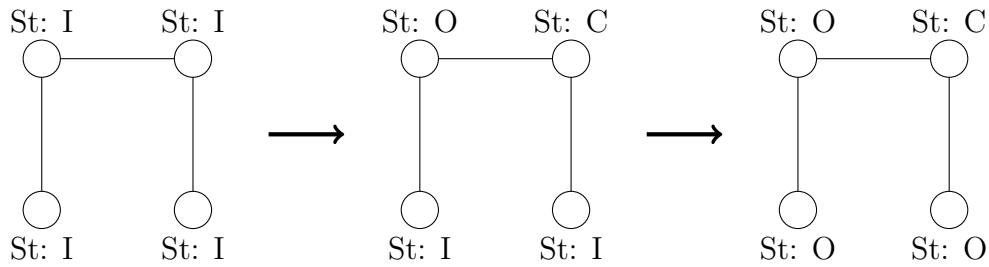
stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento operaio	divento capitano	operaio
iniziale	1	divento capitano	divento operaio	capitano
iniziale	2	divento operaio	divento capitano	operaio
iniziale	2	divento capitano	divento operaio	capitano
operaio	1	sono operaio	divento operaio	operaio
operaio	2	sono operaio	divento operaio	operaio
capitano	1	sono capitano	divento operaio	capitano
capitano	2	sono capitano	divento operaio	capitano

Osserviamo anzitutto che i due robot di tipo 1 non possono comunicare per primi. Pertanto dovranno comunicare per primi necessariamente i due robot di tipo 2. Il programma è strutturato in modo tale che uno dei due robot di tipo 2 passi allo stato 'operaio' mentre l'altro passi allo stato 'capitano'. Il programma non prevede la possibilità che due robot con stato 'operaio' e 'capitano' possano comunicare tra di loro pertanto alla seconda esecuzione del programma dovranno comunicare necessariamente i due robot di tipo 1. Il programma viene eseguito una seconda volta e termina in quanto la configurazione dei robot presenta un solo 'capitano' ed i rimanenti robot si trovano nello stato 'operaio'.



Soluzione Sala 4: primo caso.

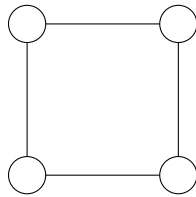




Soluzione Sala 4: secondo caso.

### 3.5 Soluzione sala 5

La sala 5 è composta da 4 robot  $P_2$  disposti come i vertici di un quadrato i cui lati sono i canali di comunicazione. Vogliamo dimostrare che risulta impossibile scrivere un programma che garantisca sempre la soluzione di questa sala perciò questa sala è *irrisolvibile*.



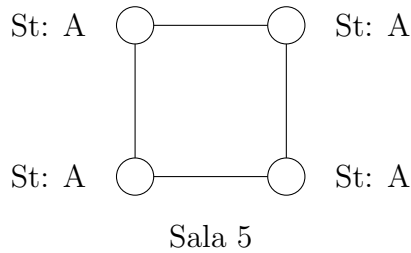
Sala 5

La dimostrazione di ciò procede per casi. Innanzitutto notiamo che la stanza allo stato iniziale è perfettamente simmetrica, con ciò intendiamo in particolare che qualsiasi coppia di robot in grado di comunicare noi scegliamo, gli altri due robot possono ugualmente comunicare. Infatti i robot hanno tutti lo stesso stato, ovvero 'iniziale' e seguono lo stesso programma  $P_2$ . Di conseguenza se stabiliamo una certa interazione tra due robot anche gli altri due potranno seguirla.

Ricordiamo che è sufficiente un solo caso in cui un programma non funzioni per definire un programma *non funzionante*. La sola disposizione finale possibile è un vertice nello stato 'capitano' e gli altri tre vertici nello stato 'operaio'. Partendo dalla situazione iniziale e assumendo che ogni comunicazione avvenga parallelamente<sup>3</sup> tra entrambe le coppie di robot abbiamo due possibilità di scelta per l'interazione tra due robot nello stato 'iniziale' del nostro ipotetico programma.

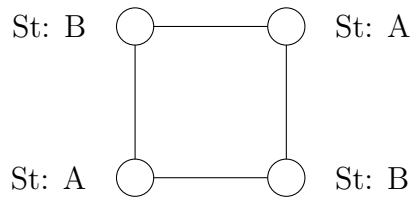
<sup>3</sup>Vogliamo dimostrare precisamente che è possibile che un qualunque programma *non funzioni* ovvero che la probabilità che un qualunque programma non funzioni sia non nulla.

Prima possibilità: l'interazione iniziale trasforma due robot nello stato 'iniziale' in due robot con lo stesso stato 'A' (i nomi dati agli stati sono puramente convenzioni). A questo punto, dopo che entrambe le coppie di robot hanno eseguito questa interazione, ci ritroviamo in una situazione analoga allo stato iniziale ovvero quattro robot di uguale stato.



Sala 5

Seconda possibilità: l'interazione trasforma due robot nello stato 'iniziale' in due robot con stati tra loro diversi 'A' e 'B'. A questo punto, siccome entrambe le coppie eseguono l'interazione e all'interno di ogni coppia i due robot sono identici, uno dei nuovi stati possibili è composto da due robot 'A' e due robot 'B' disposti in modo tale da avere robot con stati uguali non adiacenti (possiamo dire agli estremi di quelle che sarebbero le diagonali del quadrato) così che ogni robot sia collegato ai due robot di stato diverso dal proprio.



Sala 5

Nel primo caso ci ritroviamo a una situazione analoga a quella di partenza semplicemente con nomi diversi per lo stato. Perciò le possibilità per un'ulteriore interazione non sarebbero diverse da quelle iniziali se non per lo stato dei robot di partenza: ci troveremo o ancora una volta con quattro robot dello stesso stato o con la possibilità che avvenga la situazione rappresentata nel secondo caso. La prima possibilità non cambia quindi il possibile sviluppo del programma e risulterebbe quindi una scelta ininfluyente.

Nel secondo caso la situazione è ancora totalmente simmetrica, qualsiasi coppia di robot che possono comunicare, l'altra sarà indistinguibile. Dunque l'unica comunicazione possibile è tra un robot di stato 'A' e uno di stato 'B'

per cui si ripresentano le due scelte possibili per la scrittura del programma di quello iniziale.

Se scegliamo di trasformare i due robot in due robot con lo stesso stato ci si riconduce alla situazione del primo caso che come abbiamo già detto risulta ininfluente. Se scegliamo di trasformare i due robot in due robot con stati diversi (non importa quali ma solo che siano diversi tra loro) si può ripresentare la situazione del secondo caso, con solo robot di stati diversi collegati tra di loro.

In tutte le situazioni discusse la struttura è rimasta totalmente simmetrica e perciò i casi presentati possono sempre verificarsi portando il sistema ad oscillare o ripetere i due casi totalmente simmetrici. Nessuno di questi due casi è una soluzione accettabile ed essi si possono sempre verificare. Avendo esaminato le due uniche scelte quando si scrive un programma per questa stanza possiamo concludere che nessun programma potrà risolvere sempre questa sala senza ricadere in uno dei casi problematici. La sala è dunque *irrisolvibile*.

### 3.6 Soluzione sala 6

È possibile risolvere la sala utilizzando un programma simile a quello utilizzato nella sala 4.

- a) I robot di tipo  $P_1$  non possono comunicare con un robot di tipo  $P_2$  che si trovi allo stato 'iniziale' (pertanto alla prima esecuzione del programma non possono comunicare con altri robot);
- b) alla prima esecuzione del programma potranno comunicare tra di loro solo il robot centrale ed uno (ed uno solo) dei due robot che si trovano al suo fianco: uno passerà allo stato 'operaio', l'altro allo stato 'capitano';
- c) i robot allo stato 'capitano' ed 'operaio' non possono comunicare tra di loro;
- d) due robot allo stato 'operaio' non possono comunicare tra di loro;
- e) se un robot  $P_2$  allo stato 'capitano' comunica con un robot  $P_1$  allo stato 'iniziale', il robot allo stato 'capitano' rimane nello stato 'capitano' mentre il robot nello stato 'iniziale' passa allo stato 'operaio';
- f) se un robot  $P_2$  allo stato 'capitano' comunica con un robot  $P_1$  allo stato 'iniziale', il robot allo stato 'capitano' rimane nello stato 'capitano' mentre il robot nello stato 'iniziale' passa allo stato 'operaio';

- g) se un robot  $P_2$  allo stato 'operaio' comunica con un robot  $P_1$  allo stato 'iniziale', il robot allo stato 'operaio' rimane nello stato 'operaio' mentre il robot nello stato 'iniziale' passa allo stato 'operaio';
- h) se un robot  $P_2$  allo stato 'operaio' comunica con un robot  $P_1$  allo stato 'iniziale', il robot allo stato 'operaio' rimane nello stato 'operaio' mentre il robot nello stato 'iniziale' passa allo stato 'operaio'.

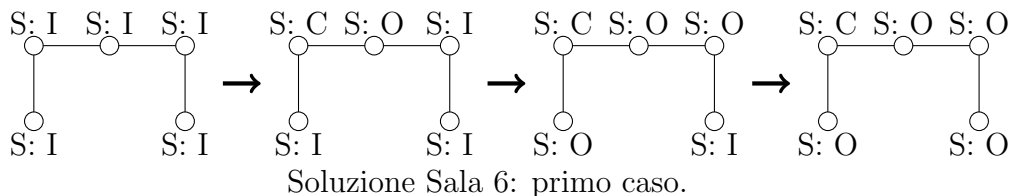
Di seguito riportiamo il codice del programma e le varie situazioni che si potrebbero verificare<sup>4</sup>.

Programma  $P_1$

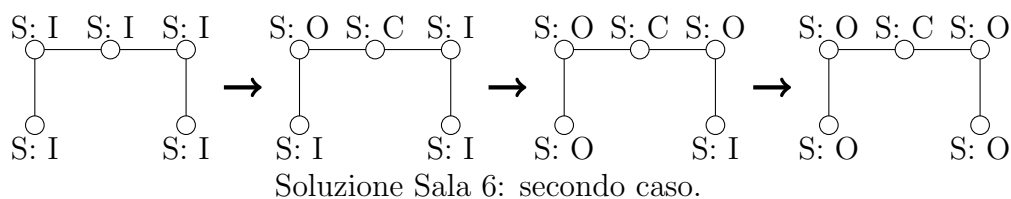
stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento operaio	sono operaio	operaio
iniziale	1	divento operaio	sono capitano	operaio

Programma  $P_2$

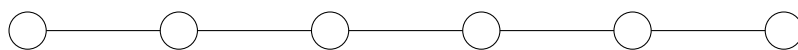
stato	canale	emissione	ricezione	nuovo stato
iniziale	1	divento operaio	divento capitano	operaio
iniziale	1	divento capitano	divento operaio	capitano
iniziale	2	divento operaio	divento capitano	operaio
iniziale	2	divento capitano	divento operaio	capitano
iniziale	1	divento operaio	sono operaio	operaio
iniziale	2	divento operaio	sono operaio	operaio
iniziale	1	divento operaio	sono capitano	operaio
iniziale	2	divento operaio	sono capitano	operaio
operaio	1	sono operaio	divento operaio	operaio
operaio	2	sono operaio	divento operaio	operaio
capitano	1	sono capitano	divento operaio	capitano
capitano	2	sono capitano	divento operaio	capitano



<sup>4</sup>Riportiamo in particolare una delle due situazioni simmetriche che si potrebbero verificare.



A conclusione del quesito proponiamo una soluzione per tutte le sale di tipo *stringa*. Definiamo una stringa come una configurazione in cui sono presenti esclusivamente robot con due canali di comunicazione, in numero naturale  $n \neq 0$ , eccetto per due robot con un solo canale di comunicazione, che fungono da estremi della stringa. Il grafico che ne risulta non è chiuso e non presenta alcuna diramazione.



Un esempio di stringa.

Costruiamo il programma nel seguente modo:

Programma  $P_1$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	A	B	operaio

Programma  $P_2$

stato	canale	emissione	ricezione	nuovo stato
iniziale	1	B	A	capitano
capitano	1	A	B	operaio
capitano	1	B	A	capitano
iniziale	2	B	A	capitano
capitano	2	A	B	operaio
capitano	2	B	A	capitano

Possiamo osservare che il programma agisce nominando immediatamente un 'capitano' ad uno o entrambi gli estremi, per poi muovere questo stato verso l'estremo opposto, fino ad incontrare un altro robot nello stato 'capitano' o un robot nello stato 'iniziale' di un robot  $P_1$ .

Partendo da una situazione in cui tutti i robot sono allo stato 'iniziale', l'unica interazione possibile è tra un robot con un solo canale di comunicazione, che utilizza dunque il programma  $P_1$ , e un robot adiacente, di tipo  $P_2$ .

Questo può avvenire anche contemporaneamente ai due estremi. Il primo passa allo stato 'operaio' e, non essendoci possibilità per robot nello stato 'operaio' di tipo  $P_1$  di comunicare, rimarrà tale fino alla fine. Il secondo passa allo stato 'capitano'. Ora, un robot  $P_2$  nello stato 'capitano' che comunichi con un robot  $P_2$  nello stato 'iniziale' passa allo stato 'operaio' ed esce dalla comunicazione, mentre il robot nello stato 'iniziale' passa allo stato 'capitano'. Questo procedimento può essere ripetuto più volte, spostando il robot nello stato 'capitano' verso l'altro estremo della stringa.

Ora possono verificarsi due casi, a seconda che il programma sia stato o meno avviato contemporaneamente all'altro estremo. In caso positivo, due robot nello stato 'capitano' di tipo  $P_2$  si trovano a comunicare forzatamente e uno dei due passa allo stato 'operaio' ed il secondo rimane nello stato 'capitano' e pertanto la configurazione è risolta. In caso negativo, la comunicazione avviene tra un robot nello stato 'capitano' di tipo  $P_2$  e un robot nello stato 'iniziale' di tipo  $P_1$  in modo che il primo rimanga 'capitano' e il secondo passi allo stato 'operaio'. La configurazione è pertanto risolta.

La soluzione è valida per qualsiasi stringa, a prescindere dalla lunghezza, in quanto il procedimento è ricorsivo.

## 4 La soluzione del terzo quesito

La soluzione del quesito si basa su quanto esposto nella risoluzione della Sala 3 (vedi pag. 14). È sufficiente un unico programma  $P_n$  (composto da  $5n$  righe) in quanto il grafo che rappresenta la configurazione dei robot è connesso.

Programma  $P_n$

stato	canale	emissione	ricezione	nuovo stato
iniziale	$1, \dots, n$	divento capitano	divento operaio	capitano
iniziale	$1, \dots, n$	divento operaio	divento capitano	operaio
iniziale	$1, \dots, n$	divento operaio	sono capitano	operaio
capitano	$1, \dots, n$	divento operaio	sono capitano	operaio
capitano	$1, \dots, n$	sono capitano	divento operaio	capitano

Ad ogni esecuzione del programma:

- a) i robot con stato iniziale 'iniziale' passano necessariamente allo stato 'capitano' o 'operaio' se comunicano con altro robot allo stato 'iniziale', altrimenti passano allo stato 'operaio' se comunicano con robot allo stato 'capitano';

- b) i robot con stato 'operaio' escono dalla comunicazione;
- c) due robot con stato 'capitano', se comunicano tra loro, passano uno allo stato 'capitano' e l'altro allo stato 'operaio'.

Notiamo inoltre che, essendo il grafo connesso, un robot con stato 'iniziale' o 'operaio' è sempre in grado di comunicare con altro robot con stato 'iniziale' o 'operaio'.

Dopo al più un numero finito di ripetizioni del programma, si ottiene un solo robot con stato 'capitano' ed i rimanenti robot hanno stato 'operaio'.

## **5 La soluzione del quarto quesito**

Abbiamo osservato (vedi pag. 17) che la 'Sala 5' non è risolvibile. Il quesito pertanto non è risolvibile in quanto una configurazione di robot come nella 'Sala 5' non consente di determinare un programma funzionante.

## **6 La soluzione del quinto quesito**

Analogamente al caso precedente non è possibile determinare sempre un programma che risolva la Sala, in quanto una configurazione di robot come il tipo della 'Sala 5' non è risolvibile.

## **7 La soluzione del sesto quesito**