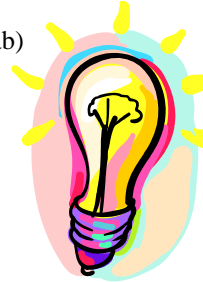


Corso di Laurea in  
Scienze cognitive e  
Processi decisionali



## Intelligenza Artificiale e analisi dei dati *Introduzione e macchine a stati finiti*

Alberto Borghese  
Università degli Studi di Milano  
Laboratorio di Sistemi Intelligenti Applicati (AIS-Lab)  
Dipartimento di Informatica  
[alberto.borghese@unimi.it](mailto:alberto.borghese@unimi.it)



A.A. 2017-2018

1/41

<http://borghese.di.unimi.it/>



## Sommario



### **Sistemi Intelligenti: cosa e perché?**

La macchina di Turing

Le macchine a stati finiti

A.A. 2017-2018

2/41



## Perchè Sistemi Intelligenti?



- Utilizzo dei calcolatori per trovare soluzioni a problemi, soluzioni che gli esseri umani non sono in grado di trovare (facilmente).
- Realizzazione di macchine che sappiano imparare da sole.
- Capacità di analizzare le situazioni e scegliere i comportamenti migliori.
- Capacità di analizzare le situazioni e scegliere i comportamenti più adatti alle situazioni (clustering).
  
- La nuova generazione di macchine mira a dare un'intelligenza alle macchine.
  
- Calcolatore come partner intelligente dell'uomo.



## Alcune macchine intelligenti



- Alcuni nomi: agenti, google engine, smart routing, smart scheduling, smart phone....
- Sistemi automatici (intelligenti) per radiografia, radioterapia e chirurgia (Cyber-Knife, da Vinci)
- Ricerca ontologica su WEB (Web2.0).
- Sistemi di visione artificiale.
- Pervasive computing.
- Agenti fissi e mobili.
- Robotica
- Scheduling

.....

.....



## Punto di vista antropocentrico



C'è più intelligenza nel risolvere un'equazione differenziale o nel bere da un bicchiere d'acqua?

Dal nostro punto di vista sicuramente nel risolvere un'equazione differenziale, ma dal punto di vista di un robot?

Sono più intelligenti gli insetti, sopravvissuti a miliardi di anni di storia o l'uomo?

Un bambino di 1 anno è in grado di riconoscere perfettamente i volti delle persone care, i sistemi di visione in grado di controllare gli accessi sono ancora in studio....

A.A. 2017-2018

5/41

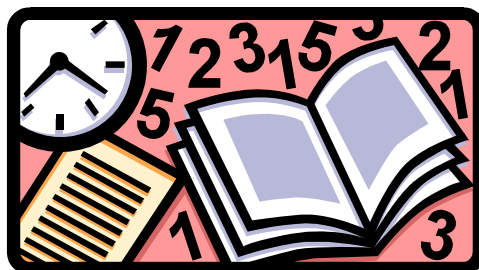
<http://borghese.di.unimi.it/>



## Quale intelligenza?



Bambini autistici di Oliver Sacks (*O. Sacks, un antropologo su Marte, Feltrinelli*).



Vincent Van Gogh (1853-1890)

<http://www.ibiblio.org/wm/paint/auth/gogh/>

<https://www.youtube.com/watch?v=91mSLGOfH2E>

<http://borghese.di.unimi.it/>



## Le intelligenze



- Linguistico-verbale (padronanza delle parole e loro utilizzo efficace), N. Chomsky).
- Logico-matematica (valutare gli oggetti scoprendone le relazioni ed i principi ad essi sottesi, H.Poincarè, A.Einstein, S.Hawkings).
- Visivo-spaziale (capacità di visualizzare e trasformare mentalmente scene tridimensionali, Michelangelo, G.Kasparov).
- Kinestesica o fisica (abilità nella percezione accurata del proprio corpo e della sua posizione, e del controllo e della coordinazione accurata del movimento, M.Jordan, K.Lewis, B.Borg).
- Naturalistica (osservazione di una parte del tutto e sua classificazione, K.Lorentz).
- Musicale (capacità di discriminazione dei suoni in modo esatto, orecchio assoluto, A.Mozart, L.van Beethoven).
- Intrapersonale (capacità di conoscere a fondo il proprio stato d'animo, le proprie aspirazioni e le pulsioni, S.Freud)
- Interpersonale (capacità di percepire gli stati d'animo altrui, M.Goethe).
- Esistenziale (Riflettere sulle domande fondamentali dell'esistenza: chi sono? Da dove vengo? Dove andiamo? (S.Kirkegaard, E.Kant).

**Perché una persona ha più o meno intelligenza in un certo campo?**



## Intelligenza (operativa)



Intelligenza è una funzione **attiva**, che consente a breve termine di trovare **soluzioni nuove** a problemi nei domini di **interazione** con l'ambiente.

Capacità di **destreggiarsi** in **situazioni nuove** o insolite cogliendo **razionalmente** i **significati** e le **relazioni** tra gli oggetti, senza passare necessariamente per tentativi ed/od addestramento.

- 1) Destreggiarsi. Agire congruentemente con un'analisi della situazione.
- 2) Cogliere le relazioni tra oggetti o situazioni. Associazione tra situazioni simili.

Le intelligenze devono co-evolvere.



## Le 4 prospettive sui sistemi intelligenti

**Prospettiva Intelligenza simbolica.** Ragionamento automatico, rappresentazione della conoscenza.

**Prospettiva Intelligenza sub-simbolica.** «Emerging intelligence», machine learning, data mining.

**Prospettiva biologica.** Se il cervello contiene intelligenza, perché non cercare di capirlo per duplicarne i meccanismi?

**Prospettiva cibernetica.** Parte dal controllo delle macchine per arrivare al controllo intelligente. Si è sviluppata nel connessionismo e più recentemente nel filone del “machine learning”.

**Prospettiva robotica.**



## Programma di massima

- L'intelligenza simbolica
- Macchine a stati finite e alberi di decisione
- I Sistemi Fuzzy
- Apprendimento statistico
- Algoritmi genetici
- Reti neurali e apprendimento nelle macchine
- Intelligenza biologica
- Realtà virtuale ed aumentata.

**Programma e materiale del corso. Sul sito WEB:**

<https://homes.di.unimi.it/~borgese/Teaching/AI/AI.html>

Slide, syllabus, note...



## Intelligenza Artificiale (6cfu)



Lezioni			
Martedì	12.30-14.30		M205 (S. Sofia)
Giovedì	12.30-15.30		Zeta (S. Sofia)
Progetti			
	Da concordare		AIS-Lab @Celorla
Ricevimento			
Su appuntamento	alberto.borghese@unimi.it renato.mainetti@unimi.it		

Tel. (02)503.14011



## Modalità d'esame



scritto + progetto

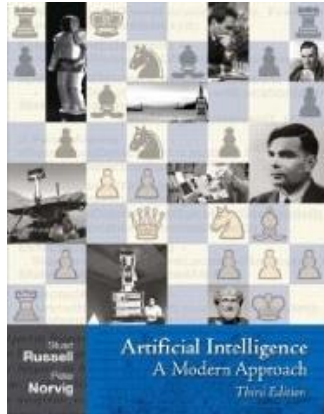
Scritto **per tutti**. E' richiesta una conoscenza di base degli argomenti del corso.

Progetto – realizzazione di algoritmi e soluzioni da sperimentare su casi reali relative ad uno degli argomenti trattati. **Progetto su una o più parti del corso.**

Per i progetti fare riferimento al Dott. Renato Mainetti  
([renato.mainetti@unimi.it](mailto:renato.mainetti@unimi.it))



## Testo principale



Stuart Russel, Peter Norvig.  
Artificial Intelligence: a Modern  
Approach, 3rd Edition Pearson. 2013.

Il materiale bibliografico è riportato a: <http://borghese.di.unimi.it/Teaching/AI/References.rtf>



## Sommario

Sistemi Intelligenti: cosa e perché?

**La macchina di Turing**

Le macchine a stati finiti



# Logica



- Principio di non-contraddizione (una proposizione vera non è falsa)
- Principio del terzo escluso (una proposizione è vera o falsa)

Connettori logici, AND, OR e NOT.

QUALUNQUE funzione logica può essere espressa combinando opportunamente tre funzioni booleane elementari.

Si dice anche che AND, OR, NOT formano un **set completo**.

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

A	Y
0	1
1	0



# Matematica e logica



Una formula della logica formale può essere soddisfatta o meno?  
Si può arrivare alle dimostrazioni automatiche? (David Hilbert, 1900).

Parallelo tra logica e matematica.

Espressioni matematiche come espressioni logiche

Dimostrazione di espressioni logiche → Dimostrazioni di teoremi.





## Il teorema di (in)completezza di Gödel (1931)



Esempio: sistema logico associato ai numeri interi positivi.

Non può esistere un metodo che consenta di stabilire, per qualunque enunciato dell'aritmetica che riguarda i numeri interi e positivi, se è vero o falso.

*“Qualunque sistema coerente di logica formale, deve comprendere enunciati veri di cui non è possibile dare una dimostrazione (1931)”.*

Questo ha spostato l'attenzione della logica dal concetto di **verità** al concetto di **provabilità**.



## Computabilità - Church, 1936



La tesi di Church (1936):

Ogni funzione che sia *effettivamente* calcolabile è *ricorsivamente* computabile ( $\lambda$ -calcolo  $\Rightarrow$  LISP).

*effettivamente* indica che esiste una procedura “meccanica” in grado di calcolare la funzione in un tempo **finito** (uscita in funzione dell'ingresso:  $y=f(x)$ )

*ricorsivamente* indica che esiste un insieme **finito** di operazioni aritmetiche elementari che, applicate all'ingresso e poi ai risultati successivi delle operazioni stesse, conducono, in un tempo **finito**, al valore della funzione.



## La macchina di Turing (1936)



Una formula della logica formale può essere soddisfatta o meno?  
Si può arrivare alle dimostrazioni automatiche? (David Hilbert, 1900).

Soluzione di un problema  $\Rightarrow$  Computazione  $\Rightarrow$   
Metodo di computazione = Algoritmo  $\Rightarrow$   
Macchina computazionale

### Dimostrazione di Turing (1936):

Qualsiasi funzione ricorsivamente computabile può essere calcolata in un tempo finito da una macchina manipolatrice di simboli (**macchina universale di Turing**).

Un algoritmo eseguibile dalla macchina di Turing è detto esattamente computabile



## Per conoscere meglio A. Turing.



- L'articolo originale di Turing:  
Alan. M.Turing, "On Computable Numbers, With an Application to the Entscheidungsproblem," *Proc. London Math. Soc.*, 2(42) (1936), 230-265; può essere reperito all'indirizzo:  
[http://www.thocp.net/biographies/papers/turing\\_oncomputablenumbers\\_1936.pdf](http://www.thocp.net/biographies/papers/turing_oncomputablenumbers_1936.pdf)
- Il sito ufficiale di Alan Turing è:  
◆ <http://www.turing.org.uk/turing/>.



## La macchina di Turing (razionale)



Come si fa a dare una definizione precisa di metodo?

Il metodo è un algoritmo. Passi elementari automatici.

Si può implementare una macchina che implementi questi passi elementari.

**Dato un tempo finito, la macchina di Turing è in grado di effettuare qualunque calcolo che possa essere eseguito da un moderno calcolatore digitale, di qualunque potenza esso sia. (esempio: programma ha un'uscita (risultato) o rimane in loop?).**

La macchina di Turing realizza un algoritmo, computabile.



## Conseguenze

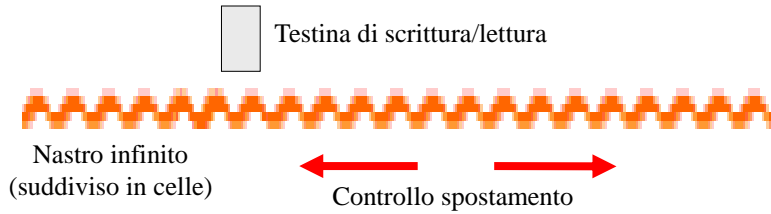


Un calcolatore digitale tradizionale, purché abbia il programma giusto, una memoria abbastanza grande e tempo a sufficienza, può calcolare qualsiasi funzione tra ingresso e uscita governata da regole (che siano meccaniche e deterministiche). Può cioè fornire in ogni caso un'uscita adeguata in funzione degli input forniti dall'ambiente e dello stato (che riassume la storia degli input dell'ambiente).

Il problema diventa **definire il programma**. Definizione simbolica (algoritmo), o definizione in linguaggio macchina (sub-simbolica).



# Una macchina di Turing



Alfabeto

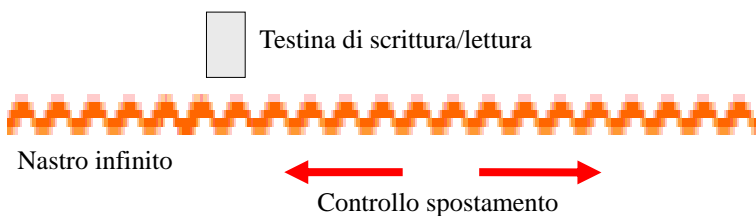
$A [a_0, a_1, .. a_n]$

Non ci sono limiti alla lunghezza della parola. Un simbolo per cella.  
Può **cancellare** o **leggere** un simbolo alla volta.

Fu sviluppata durante la guerra per decodificare le trasmissioni.



# Stato della macchina e direzione spostamento



Alfabeto

$A [a_0, a_1, .. a_n]$

Stato

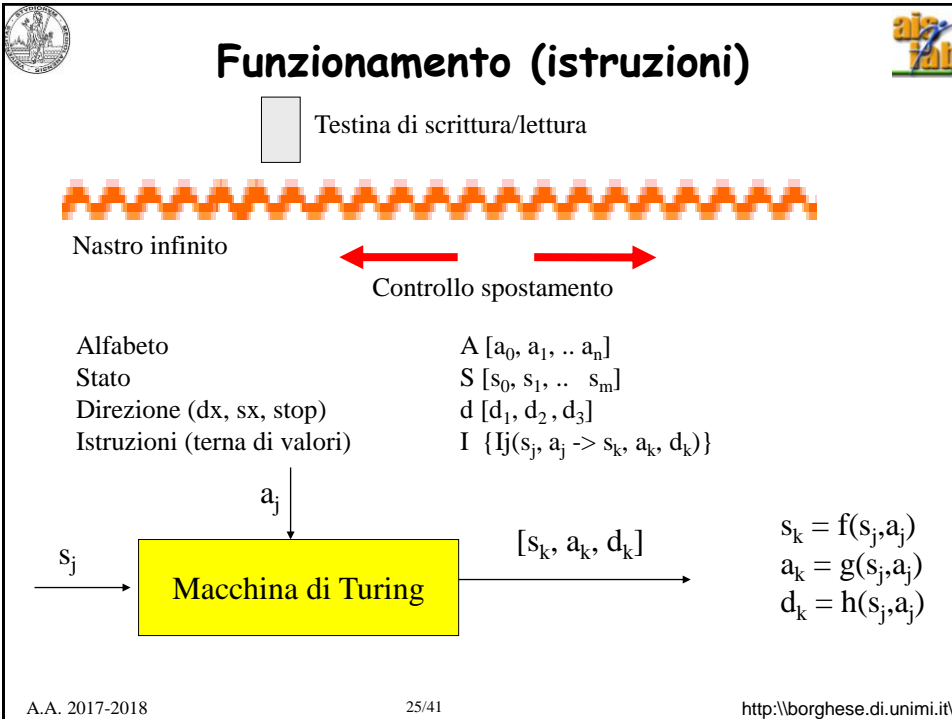
$S [s_0, s_1, .. s_m]$

Direzione (dx, sx, stop)

$d [d_1, d_2, d_3]$

Stato della macchina. Ciascuno stato viene descritto mediante gli effetti che ha l'azione ha sulla macchina stessa.

Esempio: carattere stampato quando viene premuto Shift, NumLock, Alt, nulla...).



## La macchina di Turing universale

Computabilità teorica e sperimentale coincidono (definizione intuitiva di computabilità).

Ponte tra teoria e tecnologia.

Ponte tra matematica e logica (NB i calcolatori sono basati su logica booleana, ma eseguono operazioni matematiche...).

Equivalenza tra una macchina M ed una macchina astratta (interpreti).

A.A. 2017-2018 26/41 http://borghese.di.unimi.it/



## La macchina di Turing::riassunto



La macchina di Turing universale consente di rappresentare un qualunque algoritmo e una qualunque funzione computabile.

E' basata su operazioni elementari su un alfabeto di simboli, molto simile al funzionamento di un calcolatore elettronico.

La macchina di Turing si pone quindi come soluzione operativa di un qualsiasi problema (risolvibile).

Cosa vuole dire che una macchina di Turing è in grado, mediante un algoritmo, di risolvere un qualsiasi problema. Che relazione ha ciò con l'intelligenza? E' uno strumento "intelligente"?



## Sommario



Sistemi Intelligenti: cosa e perché?

La macchina di Turing

**Le macchine a stati finiti**



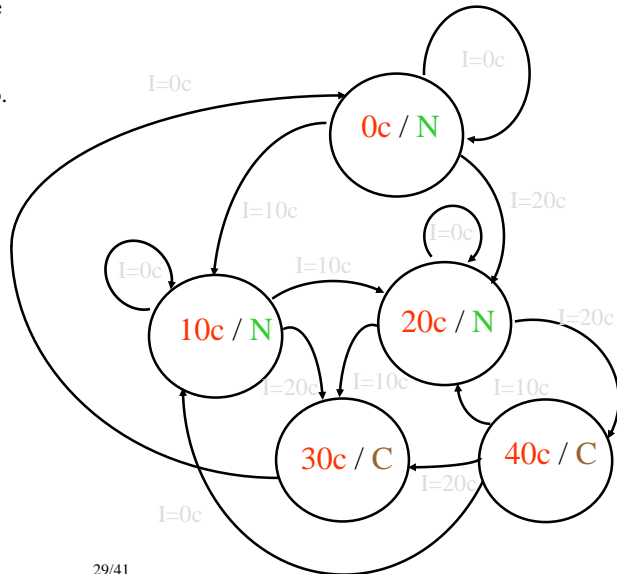
## STG di una macchina venditrice di bibite



Voglio costruire una macchinetta che eroga caffè quando l'utente ha inserito 30c.

Accetta 10 o 20c e non dà resto.

N = No Caffè  
C = Caffè.



A.A. 2017-2018

29/41



## STT della vendor machine



		I		Y
X	0c	10c	20c	No caffè
0c	0c	10c	20c	No caffè
10c	10c	20c	30c	No caffè
20c	20c	30c	40c	No caffè
30c	00c	10c	20c	Caffè
40c	10c	20c	30c	Caffè

Il controllore controlla ogni 0.5s l'ingresso e ogni 0.5s aggiorna lo stato e l'uscita.

A.A. 2016-2017

30/41

<http://borghese.di.unimi.it/>



## Macchina a Stati Finiti (di Moore)



La Macchina di Moore è definita, in teoria degli automi, dalla sestupla:

$$\langle X, I, Y, f(\cdot), g(\cdot), X_{\text{ini}} \rangle$$

**X:** insieme degli stati (in numero finito).

**I:** insieme di ingresso: tutti i simboli che si possono presentare in ingresso.

**Y:** insieme di uscita: tutti i simboli che si possono generare in uscita.

**f(.):** funzione stato prossimo:  $X' = f(X, I)$ . Definisce l'evoluzione della macchina nel tempo. L'evoluzione è deterministica.

**g(.):** funzione di uscita:  $Y = g(X)$  nelle macchine di Moore.

**Stato iniziale:**  $X_{\text{ini}}$ . Per il buon funzionamento della macchina è previsto uno stato iniziale, al quale la macchina può essere portata mediante un comando di reset.

Cosa rappresenta lo stato?



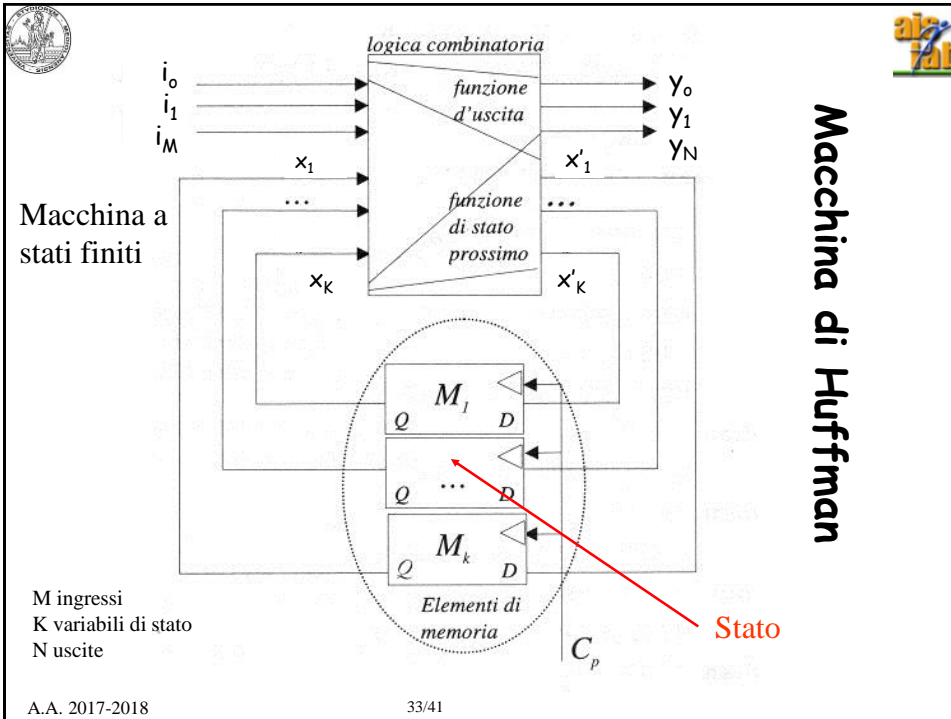
## Descrizione di una macchina di Moore



**STG: State Transition Graph** (Diagramma degli stati o Grafo delle transizioni). Ad ogni nodo è associato uno stato. Un arco orientato da uno stato  $x_i$  ad uno stato  $x_j$ , contrassegnato da un simbolo (di ingresso)  $\alpha$ , rappresenta una transizione (passaggio di stato) che si verifica quando la macchina, essendo nello stato  $x_i$ , riceve come ingresso il simbolo  $\alpha$ .

**STT: State Transition Table** (Tabella degli Stati). Per ogni coppia, (Stato presente – Ingresso), si definisce l'Uscita e lo Stato Prossimo. La forma è tabellare e ricorda le tabelle della verità da cui è derivata.





## Controllore di un semaforo

2 strade: nord-sud, NS, ed est-ovest, EO, che devono essere controllate da un semaforo. Il sistema di controllo dà via libera alternativamente alla direttrice NS o EO.

Il sistema di controllo pilota un semaforo il quale accenderà alternativamente il verde sulla direttrice NS o sulla direttrice EO (quando il semaforo non è verde, per semplicità supponiamo che sia rosso).

Il semaforo può commutare ogni 30 secondi (clock con frequenza = ?).

Supponiamo che esista una video-camera in grado di “leggere”, per ogni direttrice, se esiste almeno un’auto in attesa, oppure un’auto che si accinga ad attraversare (le due condizioni sono trattate allo stesso modo).

Il semaforo deve cambiare colore (da rosso a verde quando esiste un’auto in attesa sulla sua direttrice).

Se ci sono due auto in attesa sulle due direttrici, viene posto a verde il semaforo rosso e a rosso il semaforo verde (viene dato via libera all’altra direttrice).

Supponiamo che all’accensione venga dato via libera alla direttrice NS.

A.A. 2017-2018 34/41



## Stato, Input, Output del semaforo



**Ingresso:** {Nulla, Auto<sub>NS</sub>, Auto<sub>EO</sub>, Auto<sub>Both</sub>}

**Uscita:** {Luce<sub>NS</sub>, Luce<sub>EO</sub>}

**Stato:** ?

**f(X,I) = ?**

**G(X) = ?**

A.A. 2017-2018

35/41



## STG del semaforo



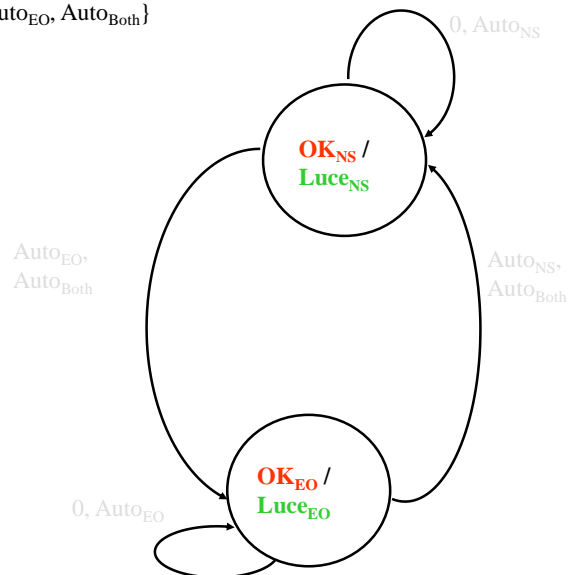
**Ingresso:** {Nulla, Auto<sub>NS</sub>, Auto<sub>EO</sub>, Auto<sub>Both</sub>}

**Uscita:** {Luce<sub>EO</sub>, Luce<sub>NS</sub>}

**Stato:** {OK<sub>NS</sub>, OK<sub>EO</sub>}

**f(X,I) = ?**

**G(X) = ?**



A.A. 2017-2018

36/41

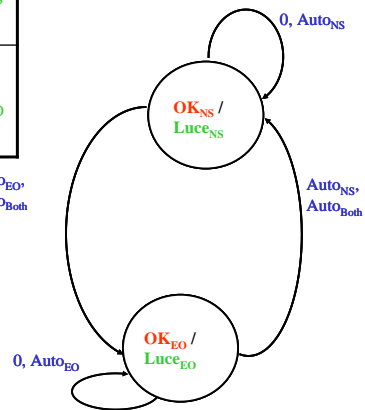


## STT del semaforo



I X	Input				Uscita
	Nulla	Auto <sub>NS</sub>	Auto <sub>EO</sub>	Auto <sub>Both</sub>	
OK <sub>NS</sub>	OK <sub>NS</sub>	OK <sub>NS</sub>	OK <sub>EO</sub>	OK <sub>EO</sub>	Luce <sub>NS</sub>
OK <sub>EO</sub>	OK <sub>EO</sub>	OK <sub>NS</sub>	OK <sub>EO</sub>	OK <sub>NS</sub>	Luce <sub>EO</sub>

Auto<sub>EO</sub>,  
Auto<sub>Both</sub>



A.A. 2017-2018

37/41



## STT del semaforo binaria



I X	Input				Uscita
	Nulla = 00	Auto <sub>NS</sub> = 01	Auto <sub>EO</sub> = 10	Auto <sub>Both</sub> = 11	
OK <sub>NS</sub> = 0	OK <sub>NS</sub> = 0	OK <sub>NS</sub> = 0	OK <sub>EO</sub> = 1	OK <sub>EO</sub> = 1	Luce <sub>NS</sub> = 1
OK <sub>EO</sub> = 1	OK <sub>EO</sub> = 1	OK <sub>NS</sub> = 0	OK <sub>EO</sub> = 1	OK <sub>NS</sub> = 0	Luce <sub>EO</sub> = 0

**Ingresso:** {Nulla, Auto<sub>NS</sub>, Auto<sub>EO</sub>, Auto<sub>Both</sub>} = {00, 01, 10, 11}

**Uscita:** {Luce<sub>EO</sub>, Luce<sub>NS</sub>} = {0, 1}

**Stato:** {OK<sub>NS</sub>, OK<sub>EO</sub>} = {0, 1}

**f(X,I) = ?**

**G(X) = ?**

A.A. 2017-2018

38/41



## I passi della progettazione di una MSF



Il committente fornisce le specifiche di funzionamento.

Definizione delle variabili di Input, Stato e Output. Definizione degli insiemi di simboli che possono essere assunti dalle variabili di Input e di Output.

Costruzione dello STG => Definizione dell'insieme di simboli che possono essere assunti dallo stato.

Costruzione della STT => Definizione implicita delle funzioni stato prossimo ed uscita.

Codifica della STT => Definizione del numero di bit per Input, Stato e Output.

STT Codificata => Circuiti combinatori che sintetizzano le funzioni  $f(X,I)$  e  $g(X)$ .



## Esercizi



■ Costruire una macchina a stati finiti (di Moore), in grado di individuare all'interno di una parola di 0 e 1 le seguenti configurazioni: 1010 e 1110. Le configurazioni si possono concatenare (e.g. 101010 da' uscita vera, al secondo e terzo 0). Stato iniziale 00.

■ Costruire una macchina a stati finiti (di Moore), con due ingressi,  $x_1$  e  $x_2$ , che fornisce 1 quando negli ultimi 3 istanti si è verificata la seguente configurazione:

	$t = -2$	$t = -1$	$t = 0$
$x_1$	0	X	1
$x_2$	x	1	0

Stato iniziale  $x_1 = 0$   $x_2 = 0$ .

■ Costruire un venditore di bibite che distribuisce una bibita quando si raggiungono i 35 cents inseriti. Non dà resto.

■ Costruire una macchina che analizza il testo da sinistra verso destra e fornisce 1 in uscita quando riconosce la stringa ABB all'interno del testo.



## Sommario



Sistemi Intelligenti: cosa e perché?

La macchina di Turing

Le macchine a stati finiti