Computabilità

C.d.S. magistrale in Matematica

Prof. Domenico Cantone

A.A. 2015/16

PROGRAMMA

TEORIA DELLA CALCOLABILITA'

FUNZIONI CALCOLABILI

- ALGORITMI, PROCEDURE EFFETTIVE
- IL MODELLO URM (UNLIMITED REGISTER MACHINE)
- FUNZIONI URM-CALCOLABILI
- PREDICATI E PROBLEMI DECIDIBILI
- CALCOLABILITA' SU ALTRI DOMINI

GENERAZIONE DI FUNZIONI CALCOLABILI

- FUNZIONI CALCOLABILI DI BASE
- UNIONE DI PROGRAMMI
- SOSTITUZIONE
- RICORSIONE
- MINIMALIZZAZIONE

TESI DI CHURCH

- ALTRI APPROCCI ALLA CALCOLABILITA'
- FUNZIONI PARZIALMENTE RICORSIVE
- FUNZIONI PRIMITIVE RICORSIVE
- MACCHINE DI TURING
- (SISTEMI DI POST E MARKOV)
- TESI DI CHURCH-TURING

ENUMERAZIONE DELLE FUNZIONI CALCOLABILI E I PROGRAMMI UNIVERSALI

- IL METODO DIAGONALE
- IL TEOREMA S-M-n
- FUNZIONI E PROGRAMMI UNIVERSALI
- DUE APPLICAZIONI DEL PROGRAMMA UNIVERSALE
- PROBLEMI INDECIDIBILI
- TEOREMA DI RICORSIONE

LIBRO DI TESTO:

N.J. CUTLAND: "COMPUTABILITY", CAMBRIDGE UNIVERSITY PRESS, 1986.

TEORIA DELLA CALCOLABILITÀ

CENNI STORICI:

NASCE CON I LAVORI DI ALAN TURING E ALONZO CHURCH CHE INTORNO AL 1936 HANNO DATO UNA SOLUZIONE NEGATIVA AL PROBLEMA DELLA DECISIONE PER LA LOGICA DEL I ORDINE CONSIDERATO DA DAVID HILBERT TRA I PROBLEMI PIÙ IMPORTANTI DEL SUO TEMPO

- PER RISOLVERE *IN POSITIVO* UN PROBLEMA ALGORITMICO E' SUFFICIENTE ESIBIRE UNA PROCEDURA IN UN FORMALISMO ACCETTABILE.

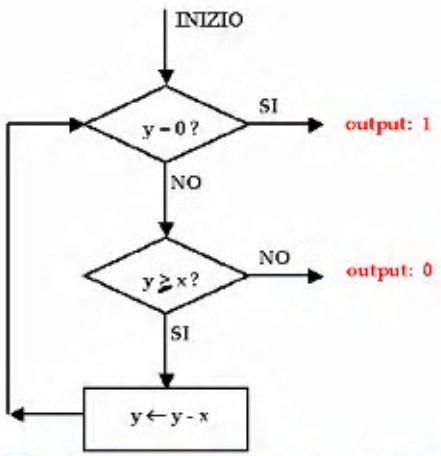
ES.
- ALGORITMO DI EUCLIDE
- ALGORITMI PER LE 4 OPERAZIONI SUI NUMERI INTERI
- ALGORITMO DI FATTORIZZAZIONE
- CRIVELLO DI ERATOSTENE
- TEST DI DIVISIBILITA'

...

ESEMPIO: TEST DI DIVISIBILITA': y E' DIVISIBILE PER x?

FORMALISMO: DIAGRAMMI DI FLUSSO

$\exists z: y = xz$?



- SI PUÒ DIMOSTRARE CHE:
  - IL PROCESSO DI CALCOLO TERMINA SEMPRE
  - SE L'OUTPUT E' 1 E L'ASSEGNAMENTO $y \leftarrow y - x$ E' ESEGUITO z VOLTE, ALLORA VALE $y = z \cdot x$
  - SE L'OUTPUT E' 0 ALLORA y NON E' DIVISIBILE PER x

QUINDI IL TEST DI DIVISIBILITA' E' COMPUTABILE

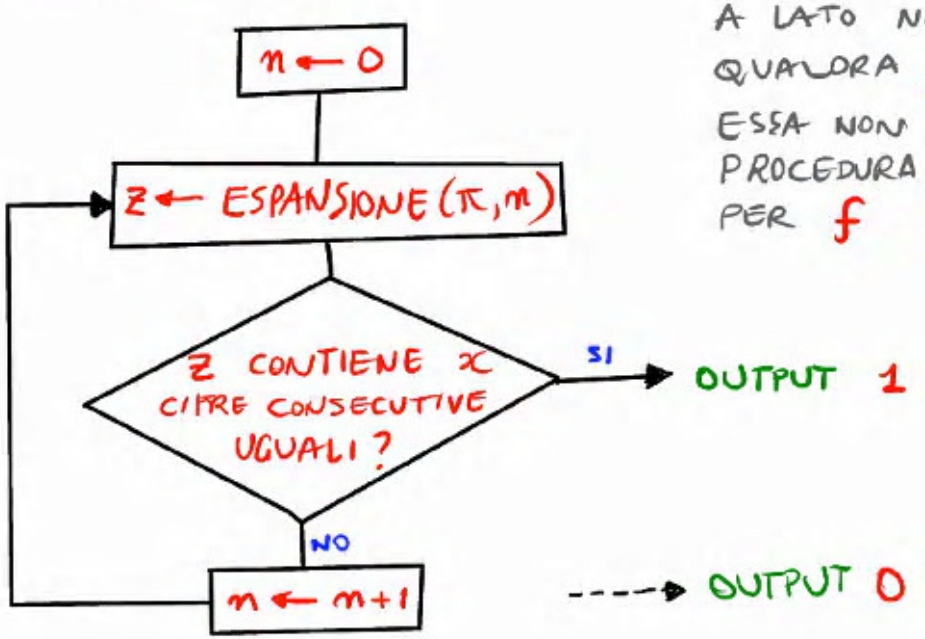
ESEMPIO

SI CONSIDERI LA FUNZIONE

$$f(x) = \begin{cases} 1 & \text{SE L'ESPANSIONE DECIMALE DI } \pi \text{ CONTIENE} \\ & \text{ESATTAMENTE } x \text{ CIFRE CONSECUTIVE EGUALI} \\ 0 & \text{ALTRIMENTI} \end{cases}$$

PUR DISPONENDO DI UNA PROCEDURA FINITA CHE PER OGNI n CALCOLA L'ESPANSIONE DI π CON n CIFRE DECIMALI, ALLO STATO ATTUALE DELLE CONOSCENZE NON SI DISPONE DI UN PROCESSO DI CALCOLO PER f.

ESEMPIO (CONTINUA)



- POICHÈ LA PROCEDURA A LATO NON SI FERMA QUALORA $f(x) = 0$, ESSA NON È UNA PROCEDURA DI CALCOLO PER f

- PER RISOLVERE IN NEGATIVO UN PROBLEMA ALGORITMICO E' NECESSARIO AVERE UN'IDEA CHIARA DI TUTTI I POSSIBILI ALGORITMI E QUINDI FARE VEDERE CHE CIASCUNO DI ESSI NON E' IN GRADO DI RISOLVERE IL PROBLEMA IN ESAME

- OCCORRE QUINDI FORMALIZZARE IL CONCETTO INTUITIVO DI ALGORITMO:

  SEQUENZA DI ISTRUZIONI CIASCUNA DELLE QUALI PUÒ ESSERE ESEGUITA IN MANIERA NON AMBIGUA ED EFFETTIVA DA UN'OPPORTUNA MACCHINA

- IL PROBLEMA DIVENTA ALLORA QUELLO DI FORMALIZZARE UN OPPORTUNO MODELLO DI CALCOLO UNIVERSALE (MACCHINA)

- SONO STATI PROPOSTI PARECCHI MODELLI:
  - MACCHINE DI TURING (TURING, 1936)
  - λ-CALCOLO (CHURCH, 1936)
  - FUNZIONI PARZIALI RICORSIVE (GÖDEL-KLEENE, 1936)
  - SISTEMI DEDUTTIVI CANONICI (POST, 1945)
  - SISTEMI DI MARKOV (MARKOV, 1951)
  - MACCHINE A REGISTRI ILLIMITATI (URM) (SHEPERDSON & STURGIS, 1936)

- CIASCUNO DI TALI SISTEMI DA' LUOGO AD UNA DEFINIZIONE DI FUNZIONI CALCOLABILI (OVVERO DI PROBLEMI RISOLVIBILI IN MANIERA EFFETTIVA)
- E' INTERESSANTE OSSERVARE CHE TUTTI I SUDDETTI MODELLI DI CALCOLO DEFINISCONO LA MEDESIMA CLASSE DI FUNZIONI CALCOLABILI

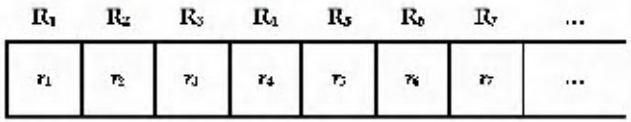
TESI DI CHURCH-TURING

UNA FUNZIONE E' CALCOLABILE SE E SOLO SE E' CALCOLABILE IN UNO DEI SUDDETTI MODELLI DI CALCOLO.

UNLIMITED REGISTER MACHINES (URM)

(MACCHINE A REGISTRI ILLIMITATI)

- SI TRATTA DI UNA VARIANTE DELLE MACCHINE DI SHEPERDSON & STURGIS [1963]
- UNA URM E' DOTATA DI UN ARRAY INFINITO DI REGISTRI $R_1, R_2, R_3, \ldots$ CIASCUNO DEI QUALI CONTIENE UN NUMERO NATURALE (INTERO NON NEGATIVO) $r_1, r_2, r_3, \ldots$



- LO STATO DI UNA URM E' IL CONTENUTO DEI SUOI REGISTRI: $\vec{r} \in \mathbb{N}^{\infty}$

- LO STATO DI UNA URM PUÒ ESSERE MODIFICATO DALL'ESECUZIONE DI UN URM-PROGRAMMA
- UN URM-PROGRAMMA È UNA SEQUENZA FINITA $I_1, I_2, \dots, I_s$ DI ISTRUZIONI DI UNO DEI SEGUENTI QUATTRO TIPI:
  - RESET
  - INCREMENTO
  - (ASSEGNAMENTO)
  - SALTO CONDIZIONATO

(s È LA LUNGHEZZA DEL PROGRAMMA)

- UNA CONFIGURAZIONE ISTANTANEA È UNA COPPIA $(k, \vec{r}) \in \mathbb{N}^+ \times \mathbb{N}^\infty$ CON
  - k INTERO POSITIVO, DETTO CONTATORE DI PROGRAMMA
  - $\vec{r}$ STATO

(INFORMALMENTE, k RAPPRESENTA L'INDICE DELLA ISTRUZIONE CHE STA PER ESSERE ESEGUITA QUANDO LO STATO DEI REGISTRI È $\vec{r}$)

ESEMPIO

$(5, (1,0,0,\dots))$ È UNA CONFIGURAZIONE ISTANTANEA

ISTRUZIONI DI RESET

$Z(n) \quad (n \in \mathbb{N}^+) \qquad [R_n \leftarrow 0]$

SEMANTICA: $(k, \vec{r}) \xrightarrow{Z(n)} (k+1, \vec{r'})$ CON $r'_i = \begin{cases} r_i, & i \neq n \\ 0, & i = n \end{cases}$

ESEMPIO

R_1	R_2	R_3	R_4	R_5	R_6	...
9	6	5	23	7	0	...

$\downarrow Z(3)$

R_1	R_2	R_3	R_4	R_5	R_6	...
9	6	0	23	7	0	...

ISTRUZIONI DI INCREMENTO

$S(n) \quad (n \in \mathbb{N}^+)$ $\quad [R_n \leftarrow R_n + 1]$

SEMANTICA: $(k, \vec{r}) \xrightarrow{S(n)} (k+1, \vec{r'})$ CON $r'_i = \begin{cases} r_i, & i \neq n \\ r_n + 1, & i = n \end{cases}$

ESEMPIO

$k = 5$

R_1	R_2	R_3	R_4	R_5	R_6	...
9	6	0	23	7	0	...

$S(5)$

$k = 6$

R'_1	R_2	R_3	R_4	R_5	R_6	...
9	6	0	23	8	0	...

ISTRUZIONI DI ASSEGNAMENTO

$T(m,n)$ $(m,n \in \mathbb{N}^+)$ $[\ R_m \leftarrow R_m\ ]$

SEMANTICA: $(k, \vec{r}) \xrightarrow{T(m,n)} (k+1, \vec{r'})$ CON $r'_i = \begin{cases} r_i, & i \neq m \\ r_n, & i = m \end{cases}$

ESEMPIO

$k=5$

R_1	R_2	R_3	R_4	R_5	R_6	...
9	6	0	23	8	0	...

$T(5,1)$

$k=6$

R_1	R_2	R_3	R_4	R_5	R_6	...
8	6	0	23	8	0	...

ISTRUZIONI DI SALTO CONDIZIONATO

$J(m,n,q)$ $(m,n,q \in \mathbb{N}^+)$ [if $R_m = R_n$ then goto q]

SEMANTICA: $(k, \vec{r}) \xrightarrow{J(m,n,q)} (k', \vec{r})$ CON $k' = \begin{cases} q, & r_m = r_n \\ k+1, & r_m \neq r_n \end{cases}$

ESEMPIO

$k=5$

R_1	R_2	R_3	R_4	R_5	R_6	...
8	6	0	23	8	0	...

$J(5,1,9)$

$k'=9$

R_1	R_2	R_3	R_4	R_5	R_6	...
8	6	0	23	8	0	...

COMPUTAZIONI

- DATI
  - $P = I_1, I_2, \ldots, I_s$ (PROGRAMMA DI LUNGHEZZA s)
  - $\vec{r}$ (STATO DEI REGISTRI)

LA COMPUTAZIONE $P(\vec{r})$ DI P A PARTIRE DALLO STATO INIZIALE $\vec{r}$ È LA SEQUENZA (FINITA O INFINITA) DI CONFIGURAZIONI ISTANTANEE

$$(k_1, \vec{r^{(1)}}), (k_2, \vec{r^{(2)}}), (k_3, \vec{r^{(3)}}), \ldots$$

TALE CHE

COMPUTAZIONI (CONTINUA)

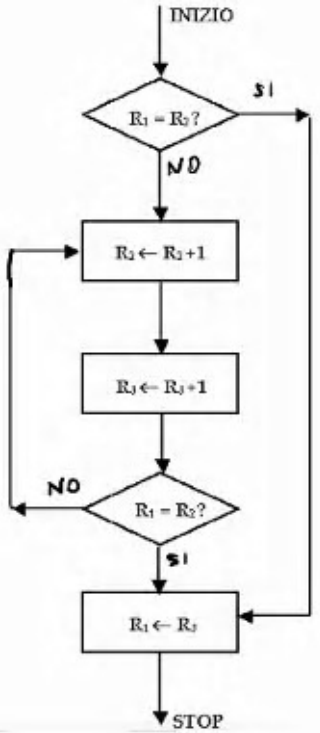
- $k_1 = 1$, $\overrightarrow{r^{(1)}} = \vec{r}$

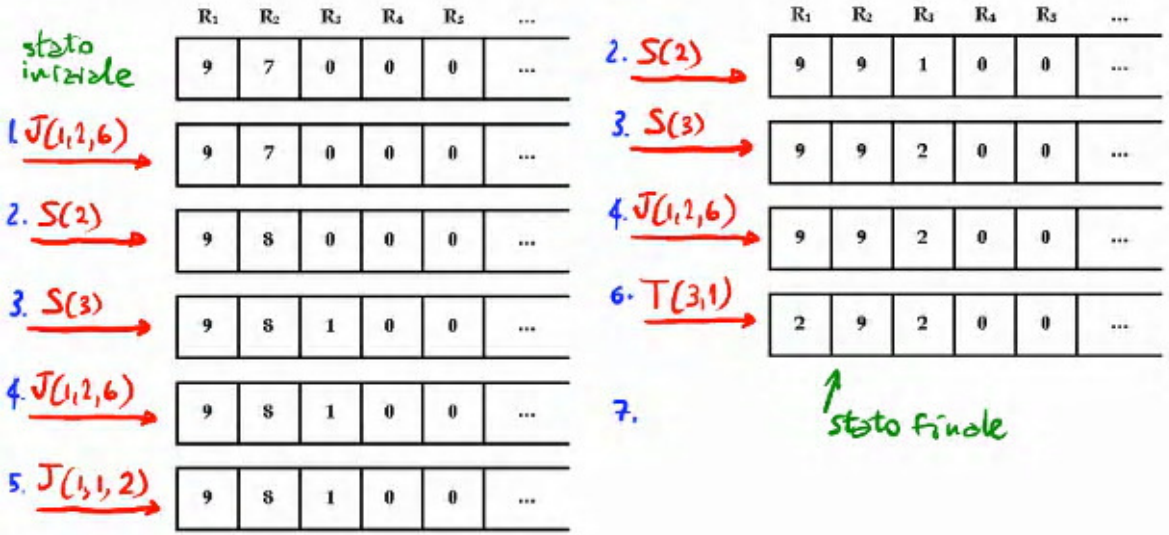
  (CIOÈ LA COMPUTAZIONE INIZIA CON LA PRIMA ISTRUZIONE DI P E CON LO STATO INIZIALE $\vec{r}$)

- SE $(k_i, \overrightarrow{r^{(i)}})$ È L'i-ESIMA CONFIGURAZIONE ISTANTANEA DELLA COMPUTAZIONE $P(\vec{r})$ ALLORA
  - SE $k_i \leq s$, $(k_i, \overrightarrow{r^{(i)}})$ HA SUCCESSORE IN $P(\vec{r})$ E SI HA $(k_i, \overrightarrow{r^{(i)}}) \xrightarrow{I_{k_i}} (k_{i+1}, \overrightarrow{r^{(i+1)}})$
  - SE $k_i > s$, $(k_i, \overrightarrow{r^{(i)}})$ NON HA SUCCESSORE IN $P(\vec{r})$ (CIOÈ $(k_i, \overrightarrow{r^{(i)}})$ È LA CONFIGURAZIONE FINALE DI $P(\vec{r})$)

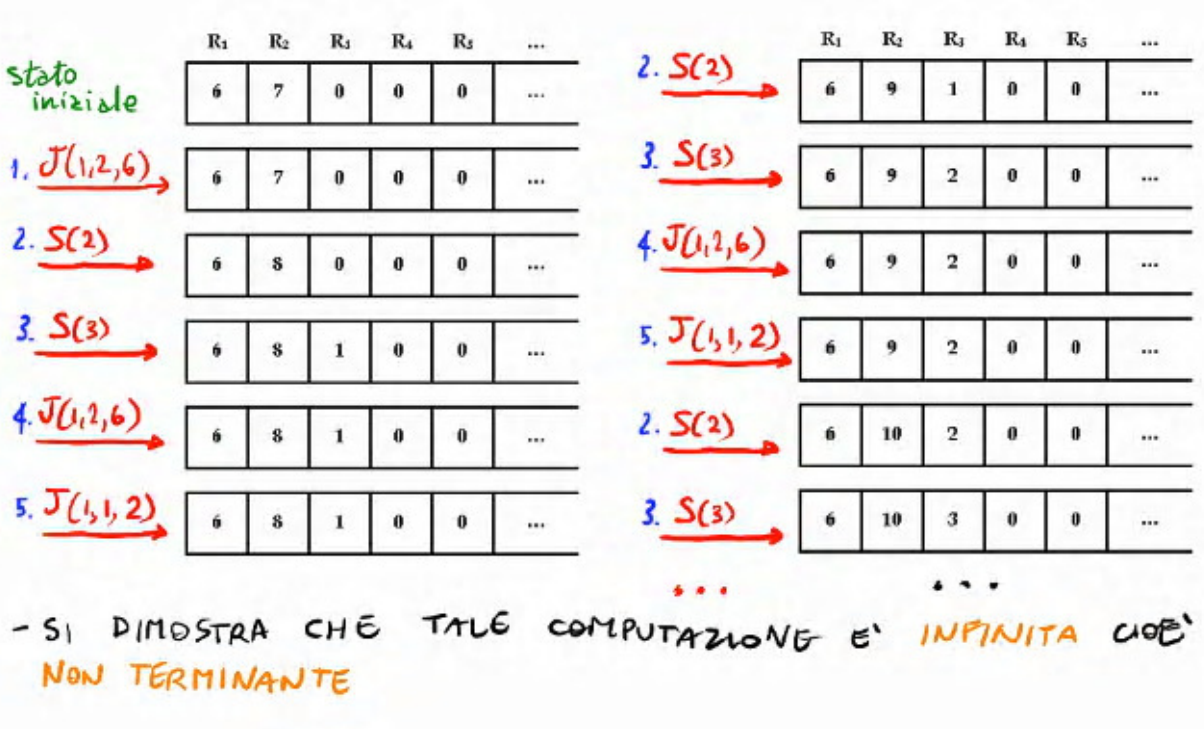
ESEMPIO

I_1 : J(1,2,6)		if $R_1 = R_2$ then goto 6
I_2 : S(2)	[2]	$R_2 \leftarrow R_2 + 1$
I_3 : S(3)		$R_3 \leftarrow R_3 + 1$
I_4 : J(1,2,6)		if $R_1 = R_2$ then goto 6
I_5 : J(1,1,2)		if $R_1 = R_1$ then goto 2
I_6 : T(3,1)	[6]	$R_1 \leftarrow R_3$





-SI TRATTA DI UNA COMPUTAZIONE FINITA O TERMINANTE



- SI DIMOSTRA CHE TALE COMPUTAZIONE E' INFINITA CIOE' NON TERMINANTE

ALCUNE NOTAZIONI UTILI

- SIA P UN PROGRAMMA ED $\vec{r}$ UNO STATO (INIZIALE) DEI REGISTRI.
- SE LA COMPUTAZIONE $P(\vec{r})$ E' TERMINANTE SCRIVEREMO $P(\vec{r})\downarrow$
- SE LA COMPUTAZIONE $P(\vec{r})$ E' NON TERMINANTE SCRIVEREMO $P(\vec{r})\uparrow$

- CON LA NOTAZIONE $P(a_1, a_2, \ldots, a_n)$ INDICHEREMO LA COMPUTAZIONE $P(\vec{r})$ DOVE

$$r_i = \begin{cases} a_i & \text{SE } i \leq n \\ 0 & \text{ALTRIMENTI} \end{cases}$$

CONVENZIONI DI INPUT E OUTPUT

- SIA $P(a_1, a_2, \dots, a_n)\downarrow$.
  L'OUTPUT DI P SU INPUT $a_1, a_2, \dots, a_n$ È IL CONTENUTO DEL REGISTRO R_1 NELLO STATO FINALE DELLA COMPUTAZIONE $P(a_1, a_2, \dots, a_n)$.
- SCRIVEREMO $P(a_1, a_2, \dots, a_n)\downarrow b$ PER INDICARE CHE b È L'OUTPUT DELLA COMPUTAZIONE $P(a_1, \dots, a_n)$

- SIA $f : A \to B$ UNA FUNZIONE PARZIALE (TALE CIOE' CHE $Dom(f) \subseteq A$)
- SE f E' DEFINITA SU a SCRIVEREMO $f(a)\downarrow$ ALTRIMENTI SCRIVEREMO $f(a)\uparrow$
- SIANO $f, g : A \to B$ FUNZIONI PARZIALI, SCRIVEREMO $f \simeq g$ PER INDICARE CHE

1) PER OGNI $a \in A$: $f(a)\downarrow$ SE E SOLO SE $g(a)\downarrow$
2) PER OGNI $a \in A$: SE $f(a)\downarrow$ ALLORA $f(a) = g(a)$

- SIA P UN PROGRAMMA.
- PER OGNI $n \geq 1$ PONIAMO

$$f_P^{(n)}(a_1, a_2, \ldots, a_n) = \begin{cases} b & \text{SE } P(a_1, a_2, \ldots, a_n) \downarrow b \\ \uparrow & \text{ALTRIMENTI} \end{cases}$$

- LA FUNZIONE (PARZIALE) $f_P^{(n)} : \mathbb{N}^n \to \mathbb{N}$ E' LA FUNZIONE n-ARIA CALCOLATA DA P
- UNA FUNZIONE PARZIALE $g : \mathbb{N}^n \to \mathbb{N}$ SI DICE URM-CALCOLABILE SE ESISTE UN PROGRAMMA URM P TALE CHE $g \simeq f_P^{(n)}$

- INDICHEREMO CON $\mathcal{C}$ LA COLLEZIONE DI TUTTE LE FUNZIONI URM-CALCOLABILI (DI QUALUNQUE ARIETÀ)
- INDICHEREMO CON $\mathcal{C}_n$ LA COLLEZIONE DI TUTTE LE FUNZIONI n-ARIE URM-CALCOLABILI
- PERTANTO VALE

$$\mathcal{C} = \bigcup_{n=1}^{\infty} \mathcal{C}_n$$

ESEMPI

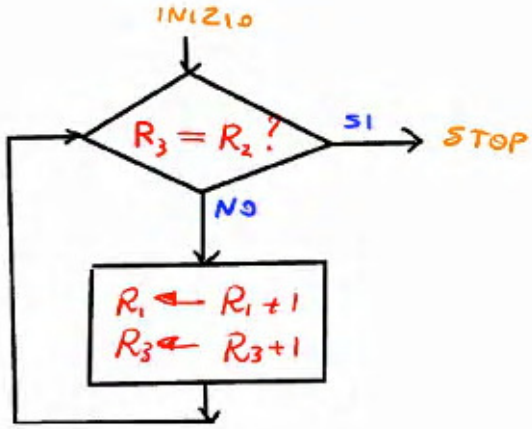
$\lambda xy.\, x+y$

I_1 $J(3,2,5)$

I_2 $S(1)$

I_3 $S(3)$

I_4 $J(1,1,1)$



- E' FACILE VERIFICARE CHE IL PROGRAMMA

$I_1 : J(1,2,6)$
$I_2 : S(2)$
$I_3 : S(3)$
$I_4 : J(1,2,6)$
$I_5 : J(1,1,2)$
$I_6 : T(3,1)$

CALCOLA LA FUNZIONE BINARIA

$$f(x,y) = \begin{cases} x - y & \text{SE } x \geq y \\ \uparrow & \text{ALTRIMENTI} \end{cases}$$

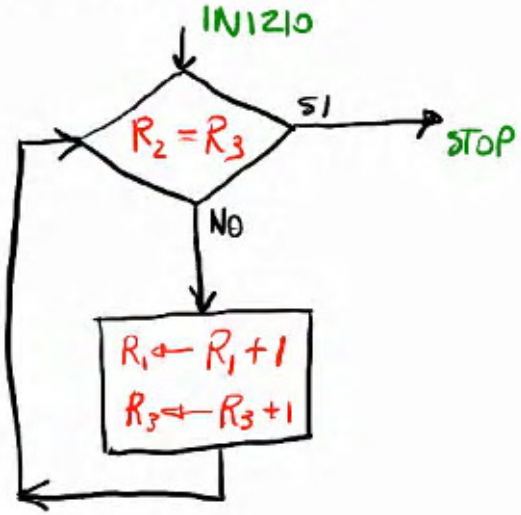
ALCUNI ESEMPI

- SI CONSIDERI LA FUNZIONE $x+y$
- IL SEGUENTE PROGRAMMA CALCOLA $x+y$

1: $J(2,3,5)$
2: $S(1)$
3: $S(3)$
4: $J(1,1,1)$

STATO TIPICO

R_1	R_2	R_3	R_4	R_5	
$x+k$	y	k	0	0	...

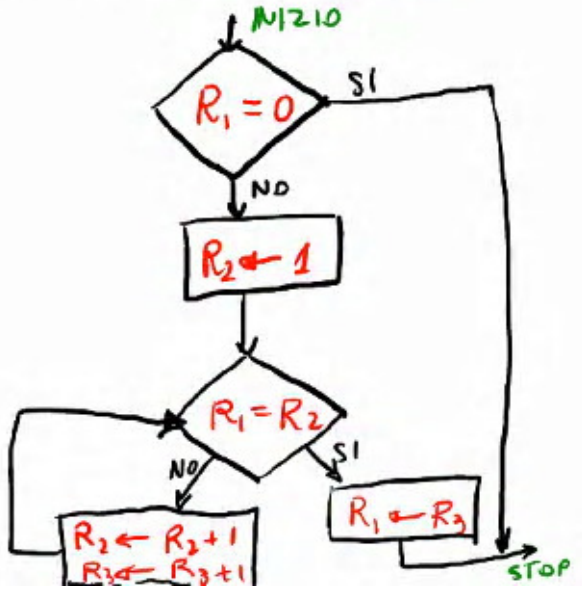


- SI CONSIDERI LA FUNZIONE $x \dot{-} 1 = \begin{cases} x-1 & \text{SE } x \geq 1 \\ 0 & \text{ALTRIMENTI} \end{cases}$
- IL SEGUENTE PROGRAMMA CALCOLA $x \dot{-} 1$

1: J(1,2,8)
2: S(2)
3: J(1,2,7)
4: S(2)
5: S(3)
6: J(1,1,3)
7: T(3,1)

STATO TIPICO

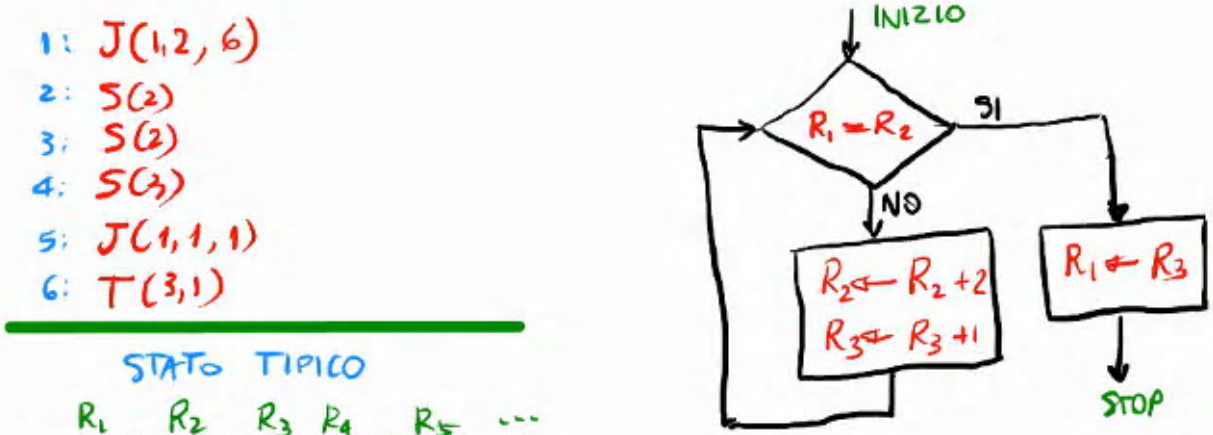
R_1	R_2	R_3	R_4	R_5	
x	$k+1$	k	0	0	...



- SI CONSIDERI LA FUNZIONE $f(x) = \begin{cases} \frac{1}{2}x & \text{SE } x \text{ E' PARI} \\ \uparrow & \text{ALTRIMENTI} \end{cases}$

- IL SEGUENTE PROGRAMMA CALCOLA f

1: J(1,2,6)
2: S(2)
3: S(2)
4: S(3)
5: J(1,1,1)
6: T(3,1)



STATO TIPICO

R_1	R_2	R_3	R_4	R_5	...
x	$2k$	k	0	0	...

ESERCIZI

1. SI DIMOSTRI CHE LE SEGUENTI FUNZIONI SONO CALCOLABILI ESIBENDO OPPORTUNI PROGRAMMI URM

(a) $$f(x) = \begin{cases} 0 & \text{SE } x = 0 \\ 1 & \text{SE } x \neq 0 \end{cases}$$

(b) $$f(x) = 5$$

(c) $$f(x,y) = \begin{cases} 0 & \text{SE } x = y \\ 1 & \text{SE } x \neq y \end{cases}$$

(d) $$f(x,y) = \begin{cases} 0 & \text{SE } x \leq y \\ 1 & \text{SE } x > y \end{cases}$$

(e) $f(x) = \begin{cases} 1/3\, x & \text{SE } x \text{ È UN MULTIPLO DI 3} \\ \uparrow & \text{ALTRIMENTI} \end{cases}$

(f) $f(x) = \left\lfloor \frac{2x}{3} \right\rfloor$

2. SIA P UN PROGRAMMA URM CHE NON CONTIENE ALCUNA ISTRUZIONE DEL TIPO $J(m,n,p)$. SI DIMOSTRI CHE ESISTE $m \in \mathbb{N}$ TALE CHE:

- $f_P^{(1)}(x) = m \quad \forall x \in \mathbb{N}$, OPPURE
- $f_P^{(1)}(x) = x + m \quad \forall x \in \mathbb{N}$,

3. DATO UN PROGRAMMA URM P SI DIMOSTRI CHE ESISTE UN ALTRO PROGRAMMA URM P' TALE CHE

- P' NON CONTIENE ALCUNA ISTRUZIONE DEL TIPO $T(m, n)$
- $f_P^{(n)}(\vec{x}) = f_{P'}^{(n)}(\vec{x})$, $\forall \vec{x} \in \mathbb{N}^n$