

# Algoritmi e Strutture Dati

Giovanni Norbedo

2024-2025

## Indice

Algoritmo	1
Modello di calcolo	1
Random-Acces Machine (RAM)	1
Istruzioni RAM	1
Assegnamento	2
Relazione di ordine	2

## Algoritmo

E' una sequenza di passi "ben definiti" per trasformare in un tempo finito un insieme di dati in input in un insieme di dati in output.

## Modello di calcolo

E' uno strumento formale per eseguire delle computazioni.

## Random-Acces Machine (RAM)

- memoria infinita (**registri**)
- ogni registro contiene un numero naturale
- **input** (sola lettura) e **output** (sola scrittura)
- il programma è una sequenza di istruzioni
- il **program counter** (PC) è un registro che indica la prossima istruzione da eseguire
- ad ogni istruzione eseguita il PC viene automaticamente incrementato

## Istruzioni RAM

- $CLR(r)$  azzerà il registro  $R_r$
- $INC(r)$  incrementa il registro  $R_r$
- $*r$  ottiene il contenuto del registro  $r$   
(se  $*5 = 8$  allora  $**5 = 8$  è il contenuto del registro 8)
- $JE(*r, s, j)$  se  $*r = s$  allora  $PC = j$
- HTL termina l'esecuzione del programma

## Assegnamento

```

1:    CLR(r)    # R_r = 0
2:    INC(r)    # R_r = 1
3:    INC(r)    # R_r = 2
...
v + 1: INC(r)   # R_r = v

R_r <- v

```

## Relazione di ordine

Verificare se  $*s < *r$

```

1:  JE(*r, *s, 9)    # se *r = *s allora PC = 9
2:  R_0 <- *r        # R_0 = *r
3:  R_1 <- *s        # R_1 = *s
4:  INC(0)           # R_0 = *r + 1
5:  JE(*0, *s, 9)    # se *r + 1 = *s allora PC = 9
6:  INC(1)           # R_1 = *s + 1
7:  JE(*1, *r, 11)   # se *s + 1 = *r allora PC = 11
8:  JE(0, 0, 4)      # PC = 4
9:  R_0 <- 0         # R_0 = 0
10: JE(0, 0, 12)     # PC = 12
11: R_0 <- 1         # R_0 = 1
12: ...             # risultato in R_0

```

$\leq, <, =, >, \geq$

### Esempio con due numeri

```

r = 5
s = 8

```

```

1:  JE(*r, *s, 9)    # 5 != 8 -> PC = 2
2:  R_0 <- *r        # R_0 = 5
3:  R_1 <- *s        # R_1 = 8
4:  INC(0)           # R_0 = 6
5:  JE(*0, *s, 9)    # 6 != 8 -> PC = 6
6:  INC(1)           # R_1 = 9
7:  JE(*1, *r, 11)   # 9 != 5 -> PC = 8
8:  JE(0, 0, 4)      # PC = 4

4:  INC(0)           # R_0 = 7
5:  JE(*0, *s, 9)    # 7 != 8 -> PC = 6
6:  INC(1)           # R_1 = 10
7:  JE(*1, *r, 11)   # 10 != 5 -> PC = 8
8:  JE(0, 0, 4)      # PC = 4

4:  INC(0)           # R_0 = 8
5:  JE(*0, *s, 9)    # 8 = 8 -> PC = 9

9:  R_0 <- 0         # R_0 = 0
10: JE(0, 0, 12)     # PC = 12

12: ...             # risultato in R_0 (falso)

```

### Esempio con due numeri

```

r = 8

```

s = 5

```
1:  JE(*r, *s, 9)      # 8 != 5 -> PC = 2
2:  R_0 <- *r          # R_0 = 8
3:  R_1 <- *s          # R_1 = 5
4:  INC(0)             # R_0 = 9
5:  JE(*0, *s, 9)      # 9 != 5 -> PC = 6
6:  INC(1)             # R_1 = 6
7:  JE(*1, *r, 11)     # 6 != 8 -> PC = 8
8:  JE(0, 0, 4)        # PC = 4

4:  INC(0)             # R_0 = 10
5:  JE(*0, *s, 9)      # 10 != 5 -> PC = 6
6:  INC(1)             # R_1 = 7
7:  JE(*1, *r, 11)     # 7 != 8 -> PC = 8
8:  JE(0, 0, 4)        # PC = 4

4:  INC(0)             # R_0 = 11
5:  JE(*0, *s, 9)      # 11 != 5 -> PC = 6
6:  INC(1)             # R_1 = 8
7:  JE(*1, *r, 11)     # 8 = 8 -> PC = 11

11: R_0 <- 1           # R_0 = 1
12: ...               # risultato in R_0 (vero)
```