

Richiami di nozioni base dei linguaggi formali

![[02-01.png]]

Sintassi delle espressioni regolari

![[02-02.png]] ![[02-04.png]]

Semantica delle espressioni regolari

![[02-03.png]]

Manca l'insieme vuoto.

![[02-05.png]]

Classificazione dei token dei linguaggi di programmazione

- Parole chiave (keyword)
 - `if, then, else, while, ...`
 - Si usa l'alternanza: `if | then | else | while ...`
 - Unica categoria sintattica: **KEYWORD**
 - Case insensitive (e.g., SQL)
 - `select, SELECT, SeLeCt, SElect, ...`
 - Noioso e troppo verboso: si usano impostazioni ad hoc
- Identificatori
 - `[a-zA-Z_][0-9a-zA-Z_]*`
 - `[a-zA-Z_]([0-9] | [a-zA-Z_])*`
 - **DIGIT** = `[0-9]`
 - **LETTER** = `[a-zA-Z_]`
 - `{LETTER}({DIGIT}|{LETTER})*`
- Costanti letterali (intere, floating point, stringa, ecc.)
 - Costanti intere
 - `{DIGIT}+`
 - Nota: accetta `000000`, non accetta `-1`
 - Costanti floating point

- `[+-]?[0-9]+\.[0-9]*`
 - occorre distinguere iterazione positiva dal carattere `+` ?
- Costanti carattere
 - `'[^']*'`
 - come specifichiamo la costante carattere `'` ?
- Operatori (matematici, logici, ecc.)
- Punteggiatura (parentesi, virgola, punto e virgola, ecc.)
 - `![[02-06.png]]`
- Commenti (singola linea, multi linea)
 - Commento (singola linea) del C++
 - `//[^\\n]*\\n`
 - Commento (singola linea) di SQL
 - `--[^\\n]*\\n`
 - Commento multilinea (C / C++ / Java / SQL / ...)
 - `/*([^*]|*+[^/*])**+ /`