**Università degli studi di Bergamo**

**Facoltà di Ingegneria**

**Corso di Laurea Magistrale in Ingegneria Informatica**



**Linguaggi formali e compilatori**

**(COD.CORSO: 38070)**

**Anno accademico: 2018/2019**

**Settore scientifico-disciplinare: ING-INF/05**

**Dipartimento: Ingegneria gestionale, dell'informazione e della produzione**

**Titolo:**

**Riconoscitore di grammatiche LR(1)**

**Autori:**

**Luca Filice Matr: 1026838**

**Matteo Gusmini Matr: 1035426**

**Davide Presciani Matr: 1035189**

Indice

[Introduzione 3](#_Toc2678563)

[Funzionamento 4](#_Toc2678564)

[Conclusioni 5](#_Toc2678565)

# Introduzione

In informatica e in linguistica, una grammatica libera dal contesto è una grammatica formale in cui ogni regola sintattica è espressa sotto forma di derivazione di un simbolo a sinistra a partire da uno o più simboli a destra. Ciò può essere espresso con due simbolismi equivalenti:

1. S := α
2. S → α

dove S è un simbolo non terminale e α è una sequenza di simboli terminali e non terminali.

L'espressione "libera dal contesto" si riferisce al fatto che il simbolo non terminale S può sempre essere sostituito da α, indipendentemente dai simboli che lo precedono o lo seguono. Un linguaggio formale si dice libero dal contesto se esiste una grammatica libera dal contesto che lo genera.

Le grammatiche libere dal contesto sono abbastanza potenti da descrivere la sintassi della maggior parte dei linguaggi di programmazione; al tempo stesso, sono abbastanza semplici da consentire un parsing molto efficiente.

La notazione formale di Backus-Naur (BNF) è la sintassi più comunemente usata per descrivere grammatiche libere dal contesto.

Un parser LR è un parser di tipo Bottom-up per grammatiche libere da contesto, usate molto di frequente nei compilatori dei linguaggi di programmazione. Un Parser LR legge il proprio input partendo da sinistra verso destra, producendo una derivazione destra. A volte questo parser viene anche indicato col nome “Parser LR(k)” dove k si riferisce al numero di simboli letti (ma non "consumati") utilizzati per prendere le decisioni di parsing.

# Funzionamento

# Conclusioni