

Daniel Alejandro Osornio López

0244685@up.edu.mx

Jorge Barba Perez

0237328@up.edu.mx

Algoritmo de fuerza bruta

En los archivos de la entrega le incluimos un ejecutable fuerza.exe ya compilado para Windows, así es más fácil que lo pruebe.

```
Windows PowerShell
PS D:\Users\0244685\Downloads> .\fuerza.exe .\gramatica.txt b*b
INFO:: Leyendo gramatica de la ruta ".\gramatica.txt"
INFO:: Gramatica con representacion Grammar { nt_sorted: [2, 4, 0, 1, 3, 5], non_terminal: ["S", "E", "E'", "T", "T'", "F"], number_rules: [1, 1, 2, 1, 2, 2], starting_ptr: [0, 1, 2, 4, 5, 7], rhs: ["E", "TE'", "+TE'", "", "FT'", "*FT'", "", "b", "(E)"] }
INFO:: Evaluando "b*b#"
INFO:: Grammar {
  S -> E
  E -> TE'
  E' -> +TE'|ε
  T -> FT'
  T' -> *FT'|ε
  F -> b|(E)
}

(Q, 1, ε, S#)
(Caso 1 ) |- (Q, 1, {S1}, E#)
(Caso 1 ) |- (Q, 1, {S1}{E1}, TE'#)
(Caso 1 ) |- (Q, 1, {S1}{E1}{T1}, FT'E'#)
(Caso 1 ) |- (Q, 1, {S1}{E1}{T1}{F1}, bT'E'#)
(Caso 2 ) |- (Q, 2, {S1}{E1}{T1}{F1}b, T'E'#)
(Caso 1 ) |- (Q, 2, {S1}{E1}{T1}{F1}b{T'1}, *FT'E'#)
(Caso 2 ) |- (Q, 3, {S1}{E1}{T1}{F1}b{T'1}*, FT'E'#)
(Caso 1 ) |- (Q, 3, {S1}{E1}{T1}{F1}b{T'1}*{F1}, bT'E'#)
(Caso 2 ) |- (Q, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b, T'E'#)
(Caso 1 ) |- (Q, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'1}, *FT'E'#)
(Caso 4 ) |- (B, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'1}, *FT'E'#)
(Caso 6a) |- (Q, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'2}, E'#)
(Caso 1 ) |- (Q, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'2}{E'1}, +TE'#)
(Caso 4 ) |- (B, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'2}{E'1}, +TE'#)
(Caso 6a) |- (Q, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'2}{E'2}, #)
(Caso 3 ) |- (T, 4, {S1}{E1}{T1}{F1}b{T'1}*{F1}b{T'2}{E'2}, ε)

La gramatica acepto la entrada "b*b#"

```

Archivo de entrada

El archivo de entrada espera reglas de la forma:

```
NOMBRE[:] VAR1 [| VAR2 [| VAR3 [...]]]
```

- Es decir, el nombre, después opcionalmente : o un espacio para indicar el fin del nombre, después una o más producciones, separadas por |.
- Los espacios antes y después de las producciones son descartados, más no los espacios dentro de la producción, de forma que:
 - 'a B' \neq 'aB'
 - 'aB|cC' = ' aB | cC '.
- El nombre de los no terminales pueden ser de longitud mayor a 1. Por ejemplo, S' y S'' son nombres válidos.

1. Ejemplo de entrada:

```
S E
E TE'
E' +TE'|
T FT'
T' *FT'|
F b|(E)
```

1. Otro ejemplo de entrada válida

```
E: aE | bE | cS
S: eS | bB | cB
B: bE | cE | c | g | h
```

Ejemplos

1. `cargo run --release -- gramatica2.txt 'ccc'`

```
ae@march ~/s/l/fuerza (main)> cargo run --release -- gramatica2.txt 'ccc'
    Finished release [optimized] target(s) in 0.00s
    Running `target/release/fuerza gramatica2.txt ccc`
INFO:: Leyendo gramatica de la ruta "gramatica2.txt"
INFO:: Gramatica con representacion Grammar { nt_sorted: [0, 1, 2], non_terminal: ["E", "S", "B"], number_rules: [3, 3, 5], starting_ptr: [0, 3, 6], rhs: ["aE", "bE", "cS", "eS", "bB", "cB", "bE", "cE", "c", "g", "h"] }
INFO:: Evaluando "ccc#"
INFO:: Grammar {
  E → aE|bE|cS
  S → eS|bB|cB
  B → bE|cE|c|g|h
}

(Q, 1, ε, E#)
(Caso 1 ) ⊢ (Q, 1, {E1}, aE#)
(Caso 4 ) ⊢ (B, 1, {E1}, aE#)
(Caso 6a) ⊢ (Q, 1, {E2}, bE#)
(Caso 4 ) ⊢ (B, 1, {E2}, bE#)
(Caso 6a) ⊢ (Q, 1, {E3}, cS#)
(Caso 2 ) ⊢ (Q, 2, {E3}c, S#)
(Caso 1 ) ⊢ (Q, 2, {E3}c{S1}, eS#)
(Caso 4 ) ⊢ (B, 2, {E3}c{S1}, eS#)
(Caso 6a) ⊢ (Q, 2, {E3}c{S2}, bB#)
(Caso 4 ) ⊢ (B, 2, {E3}c{S2}, bB#)
(Caso 6a) ⊢ (Q, 2, {E3}c{S3}, cB#)
(Caso 2 ) ⊢ (Q, 3, {E3}c{S3}c, B#)
(Caso 1 ) ⊢ (Q, 3, {E3}c{S3}c{B1}, bE#)
(Caso 4 ) ⊢ (B, 3, {E3}c{S3}c{B1}, bE#)
(Caso 6a) ⊢ (Q, 3, {E3}c{S3}c{B2}, cE#)
(Caso 2 ) ⊢ (Q, 4, {E3}c{S3}c{B2}c, E#)
(Caso 1 ) ⊢ (Q, 4, {E3}c{S3}c{B2}c{E1}, aE#)
(Caso 4 ) ⊢ (B, 4, {E3}c{S3}c{B2}c{E1}, aE#)
(Caso 6a) ⊢ (Q, 4, {E3}c{S3}c{B2}c{E2}, bE#)
(Caso 4 ) ⊢ (B, 4, {E3}c{S3}c{B2}c{E2}, bE#)
(Caso 6a) ⊢ (Q, 4, {E3}c{S3}c{B2}c{E3}, cS#)
(Caso 4 ) ⊢ (B, 4, {E3}c{S3}c{B2}c{E3}, cS#)
(Caso 6c) ⊢ (B, 4, {E3}c{S3}c{B2}c, E#)
(Caso 5 ) ⊢ (B, 3, {E3}c{S3}c{B2}, cE#)
(Caso 6a) ⊢ (Q, 3, {E3}c{S3}c{B3}, c#)
(Caso 2 ) ⊢ (Q, 4, {E3}c{S3}c{B3}c, #)
(Caso 3 ) ⊢ (T, 4, {E3}c{S3}c{B3}c, e)

La gramatica acepto la entrada "ccc#"
```

2. `cargo run --release -- gramatica2.txt 'ccc'`

```

ae@march ~/s/l/fuerza (main)> cargo run --release -- gramatica.txt 'b+b'
    Finished release [optimized] target(s) in 0.00s
    Running `target/release/fuerza gramatica.txt b+b`
INFO:: Leyendo gramatica de la ruta "gramatica.txt"
INFO:: Gramatica con representacion Grammar { nt_sorted: [2, 4, 0, 1, 3, 5], non_terminal: ["S", "E", "E'", "T", "T'", "F"], number_rules: [1, 1, 2, 1, 2, 2], starting_ptr: [0, 1, 2, 4, 5, 7], rhs: ["E", "TE'", "+TE'", "", "FT'", "*FT'", "", "b", "(E)"] }
INFO:: Evaluando "b+b#"
INFO:: Grammar {
  S → E
  E → TE'
  E' → +TE'|ε
  T → FT'
  T' → *FT'|ε
  F → b|(E)
}

(Q, 1, ε, S#)
(Caso 1 ) ⊢ (Q, 1, {S1}, E#)
(Caso 1 ) ⊢ (Q, 1, {S1}{E1}, TE'#)
(Caso 1 ) ⊢ (Q, 1, {S1}{E1}{T1}, FT'E'#)
(Caso 1 ) ⊢ (Q, 1, {S1}{E1}{T1}{F1}, bT'E'#)
(Caso 2 ) ⊢ (Q, 2, {S1}{E1}{T1}{F1}b, T'E'#)
(Caso 1 ) ⊢ (Q, 2, {S1}{E1}{T1}{F1}b{T'1}, *FT'E'#)
(Caso 4 ) ⊢ (B, 2, {S1}{E1}{T1}{F1}b{T'1}, *FT'E'#)
(Caso 6a) ⊢ (Q, 2, {S1}{E1}{T1}{F1}b{T'2}, E'#)
(Caso 1 ) ⊢ (Q, 2, {S1}{E1}{T1}{F1}b{T'2}{E'1}, +TE'#)
(Caso 2 ) ⊢ (Q, 3, {S1}{E1}{T1}{F1}b{T'2}{E'1}+, TE'#)
(Caso 1 ) ⊢ (Q, 3, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}, FT'E'#)
(Caso 1 ) ⊢ (Q, 3, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}, bT'E'#)
(Caso 2 ) ⊢ (Q, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b, T'E'#)
(Caso 1 ) ⊢ (Q, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'1}, *FT'E'#)
(Caso 4 ) ⊢ (B, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'1}, *FT'E'#)
(Caso 6a) ⊢ (Q, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'2}, E'#)
(Caso 1 ) ⊢ (Q, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'2}{E'1}, +TE'#)
(Caso 4 ) ⊢ (B, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'2}{E'1}, +TE'#)
(Caso 6a) ⊢ (Q, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'2}{E'2}, #)
(Caso 3 ) ⊢ (T, 4, {S1}{E1}{T1}{F1}b{T'2}{E'1}+{T1}{F1}b{T'2}{E'2}, ε)
La gramatica acepto la entrada "b+b#"

```