

Beskontekstne gramatike (Context-Free Grammar)

- Skup terminalnih simbola
{ Op, Int, Open, Close }
Svaki terminal je definisan
regularnim izrazom

Op = +|-|*|/

Int = [0-9] [0-9]*

Open = <

Close = >

- Skup neterminala
{ *Start*, *Expr* }

Start → *Expr*

Expr → *Expr* Op *Expr*

Expr → Int

Expr → Open *Expr* Close

- Skup pravila (produkcija)
 - jedan neterminal sa lijeva
 - niz terminala i
neterminala sa desne strane pravila

Generisanje

Počinjemo sa neterminalom *Start* (ili onom koji je označen kao početni neterminal)

loop until (nema neterminala u stringu)

izaberi neterminal

izaberi produkciju sa tim neterminalom sa lijeve strane

zamijeni neterminal sa desnom stranom produkcije

- Ovo je proces izvođenja (derivacije)

Primjer izvođenja

Op = +|-|*|/

Int = [0-9] [0-9]*

Open = <

Close = >

<2-1>+1

1) $Start \rightarrow Expr$

2) $Expr \rightarrow Expr Op Expr$

3) $Expr \rightarrow Int$

4) $Expr \rightarrow Open Expr Close$

Start

Expr

Expr Op Expr

Open *Expr* Close Op *Expr*

Open *Expr* Op *Expr* Close Op *Expr*

Open Int Op *Expr* Close Op *Expr*

Open Int Op *Expr* Close Op Int

Open Int Op Int Close Op Int

Primjer izvođenja

Op = +|-|*|/

Int = [0-9] [0-9]*

Open = <

Close = >

1) *Start* → *Expr*

2) *Expr* → *Expr Op Expr*

3) *Expr* → Int

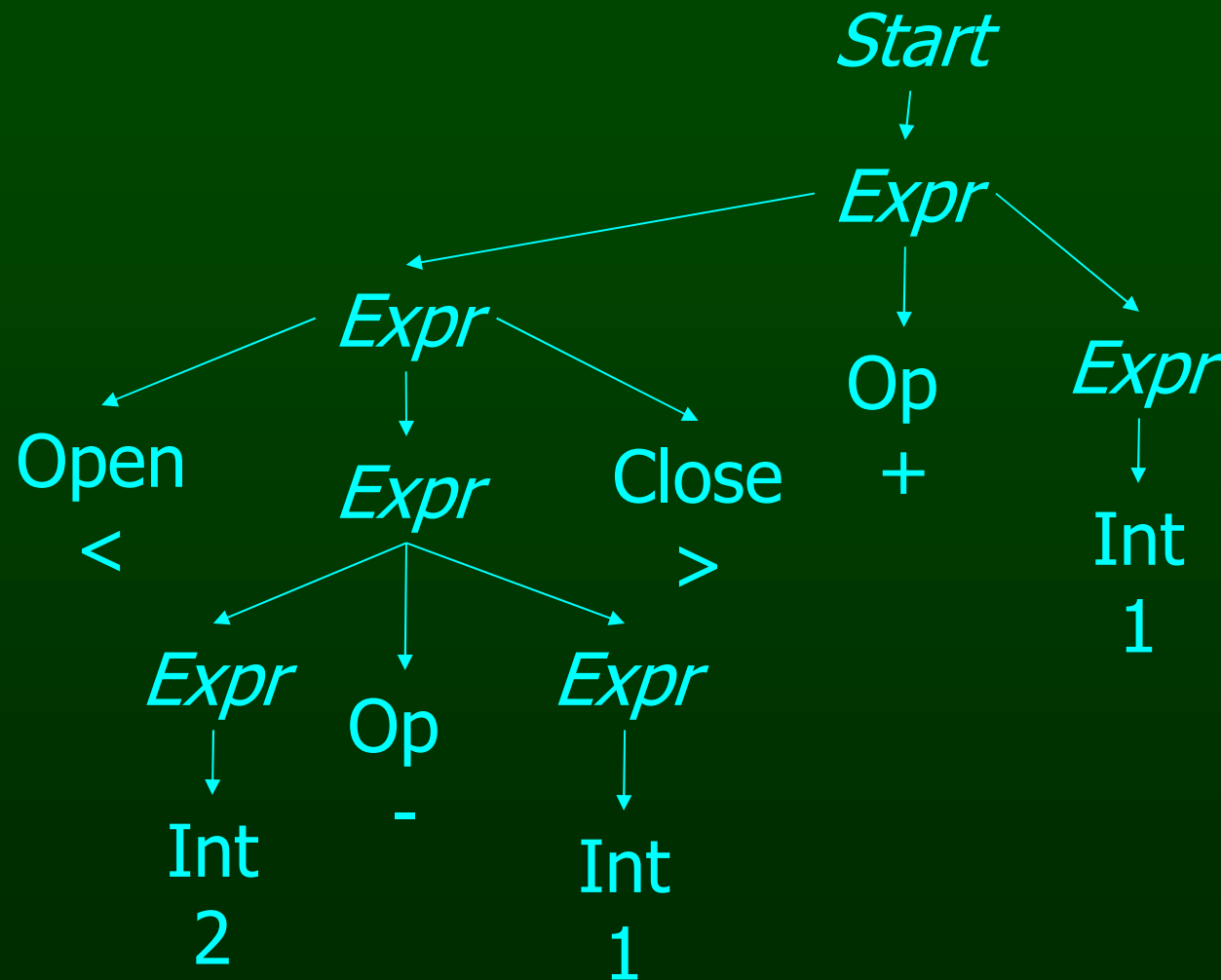
4) *Expr* → Open *Expr* Close

Open Int Op Int Close Op Int
< Int Op Int Close Op Int
< [0-9][0-9]* Op Int Close Op Int
< 2 Op Int Close Op Int
< 2 +|-|*|/ Int Close Op Int
< 2 - Int Close Op Int
< 2 - [0-9][0-9]* Close Op Int
< 2 - 1 Close Op Int
< 2 - 1 > Op Int
< 2 - 1 > +|-|*|/ Int
< 2 - 1 > + Int
< 2 - 1 > + [0-9][0-9]*
< 2 - 1 > + 1

Drvo parsiranja

- unutrašnji čvorovi: neterminali
- listovi: terminali
- grane: individualna izvođenja

Drvo parsiranja za <2-1>+1



Primjer

$Start \rightarrow S$

$S \rightarrow (L)$

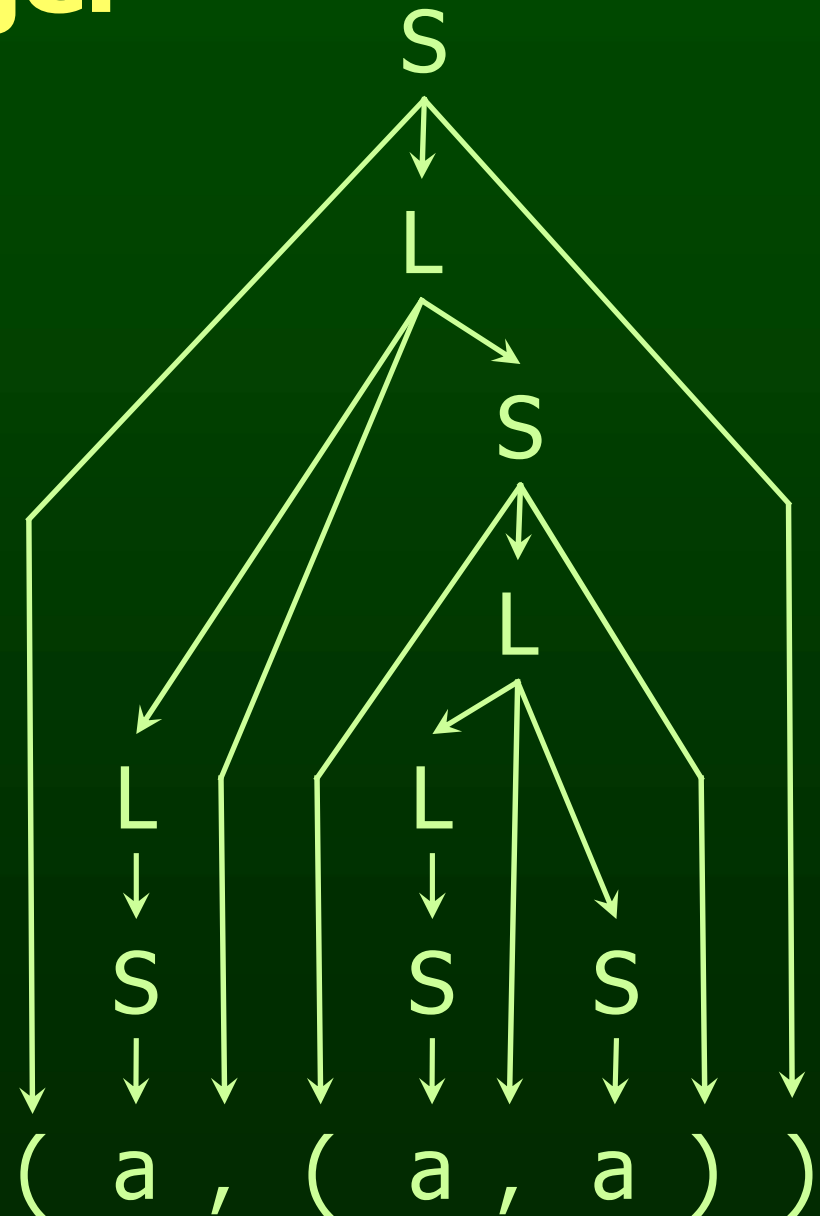
$S \rightarrow a$

$L \rightarrow L , S$

$L \rightarrow S$

Drvo parsiranja za

$(a, (a, a))$



Definisanje KSG

KSG je struktura $\{S, P, N, T\}$

S = startno stanje

P = produkcije (pravila)

N = neterminali

T = terminali

$V = N \cup T$

$p \in P$ ima oblik $X \rightarrow W_1 W_2 \dots W_n$

$X \in N$

$W_i \in V, i=1, \dots, n$

KSG primjer 1

Palindromi

npr. "abba"

$T = \{a, b\}$ $N = \{E\}$ $S = E$

$P = \{ E \rightarrow aEa \mid bEb \mid \varepsilon \}$

Derivacija

1. $E \Rightarrow aEa$

2. $E \Rightarrow abEba$

3. $E \Rightarrow ab\varepsilon ba$

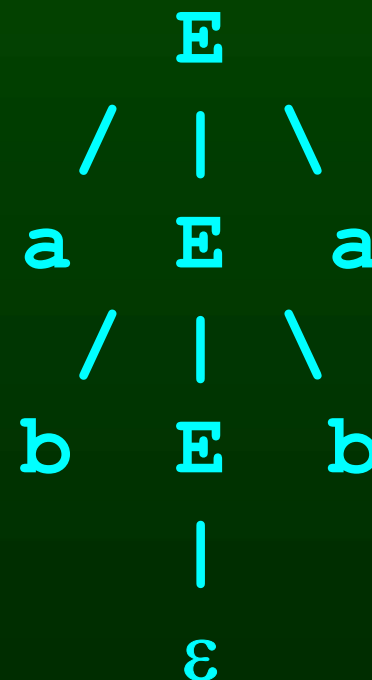
4. $E \Rightarrow abba$

Pravilo

$E \rightarrow aEa$

$E \rightarrow bEb$

$E \rightarrow \varepsilon$

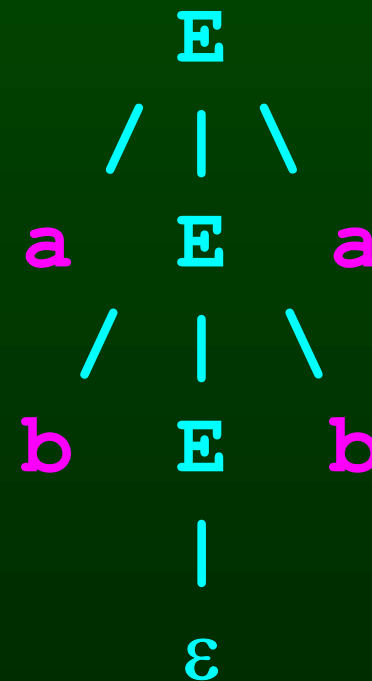


KSG primjer 1

$T = \{a, b\}$ $N = \{ E \}$ $S = E$

$P = \{ E \rightarrow aEa \mid bEb \mid \varepsilon \}$

Prikaz rečenice – obilazak stabla

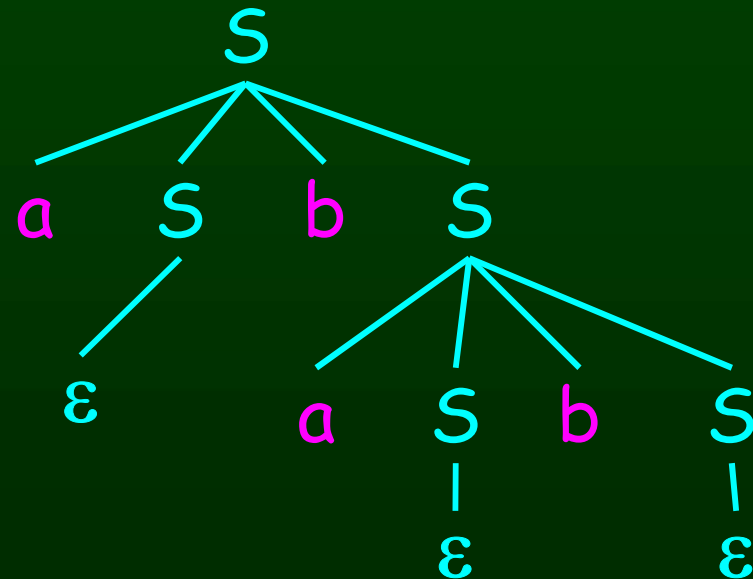
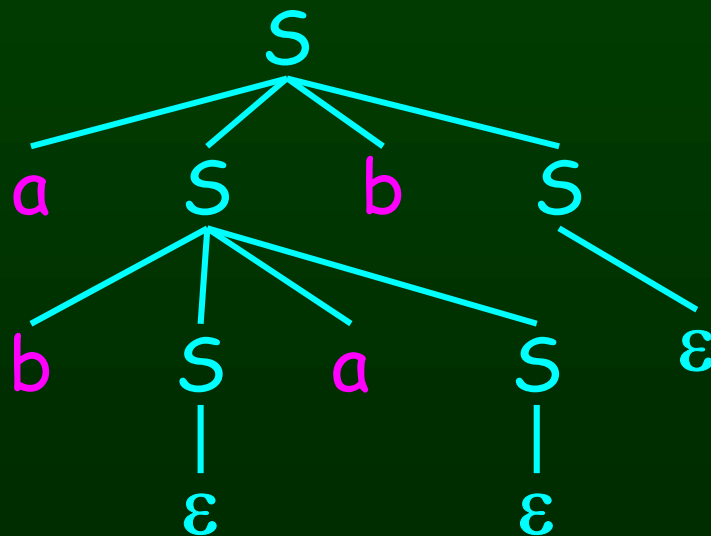


KSG primjer 2

$T = \{a, b\}$ $N = \{S\}$

$S = S, S = \{ S \rightarrow \varepsilon \mid aSbS \mid bSaS \}$

Stringovi sa jednakim brojem slova a i b.



KSG primjer 3

Start \rightarrow *Expr*

Expr \rightarrow *Expr* + *Term*

Expr \rightarrow *Expr* - *Term*

Expr \rightarrow *Term*

Term \rightarrow *Term* * Int

Term \rightarrow *Term* / Int

Term \rightarrow Int

- Skup tokena je { +, -, *, /, Int }, gdje Int = [0-9][0-9]*
- Svaki Int n token predstavljamo samo sa n

Primjer parsiranja

Parse
Tree

Start

Preostali dio ulaza

2-2*2

Rečenična forma

Start

Trenutna pozicija u drvetu

Primjer parsiranja

Parse
Tree

Start



Expr



Trenutna pozicija u drvetu

Preostali dio ulaza

2-2*2

Rečenična forma

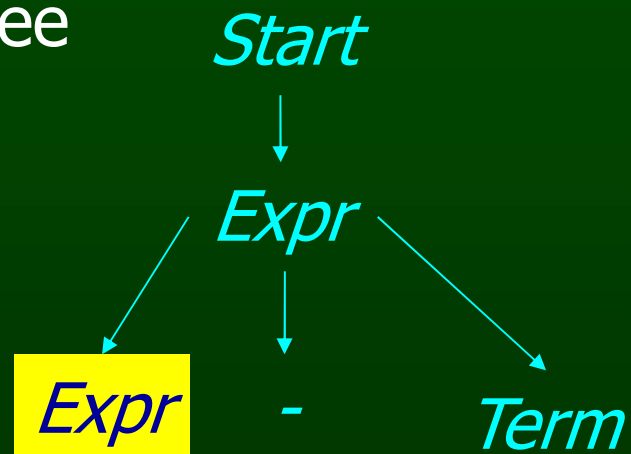
Expr

Primijenjeno pravilo

Start → *Expr*

Primjer parsiranja

Parse
Tree



Preostali dio ulaza

2-2*2

Rečenična forma

$Expr - Term$

$Expr \rightarrow Expr + Term$

$Expr \rightarrow Expr - Term$

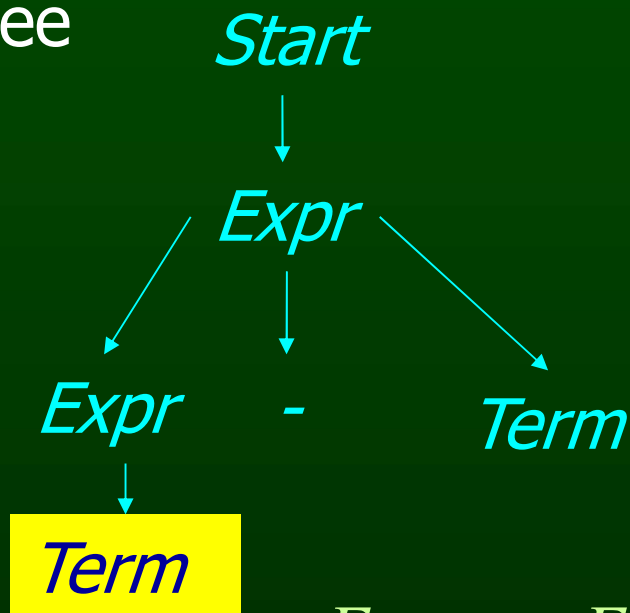
$Expr \rightarrow Term$

Primijenjeno pravilo

$Expr \rightarrow Expr - Term$

Primjer parsiranja

Parse
Tree



Preostali dio ulaza

2-2*2

Rečnična forma

Term - Term

Expr \rightarrow *Expr* + *Term* Primijenjeno pravilo

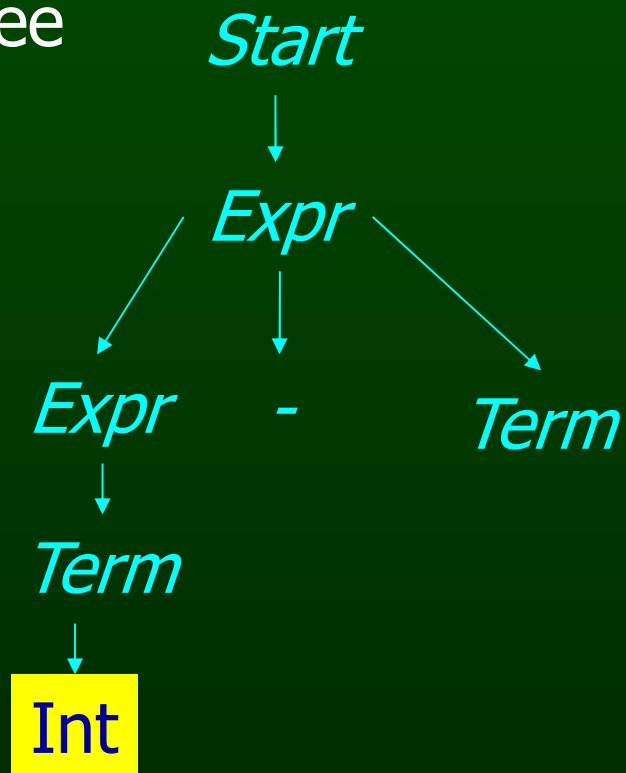
Expr \rightarrow *Expr* - *Term*

Expr \rightarrow *Term*

Expr \rightarrow *Term*

Primjer parsiranja

Parse
Tree



Preostali dio ulaza

2-2*2

Rečenična forma

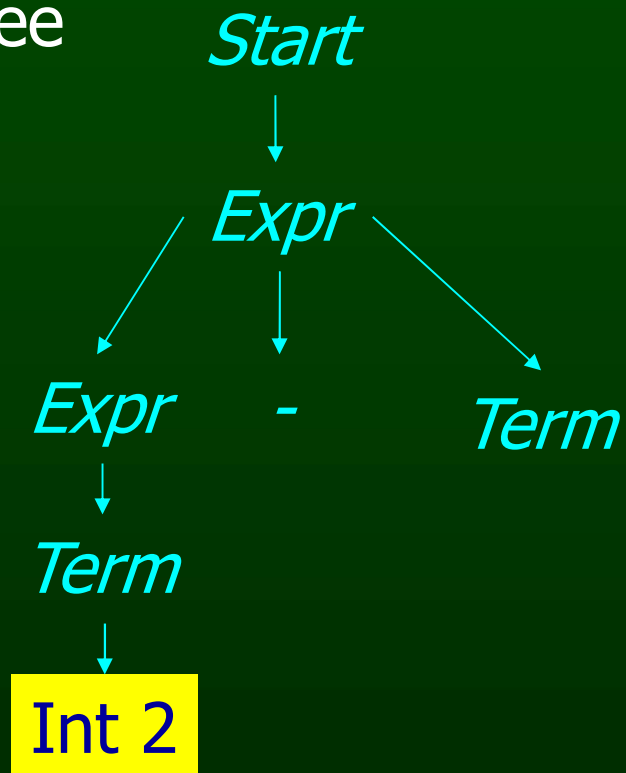
$Int - Term$

Primijenjeno pravilo

$Term \rightarrow Int$

Primjer parsiranja

Parse
Tree



Match
Input
Token!

Preostali dio ulaza

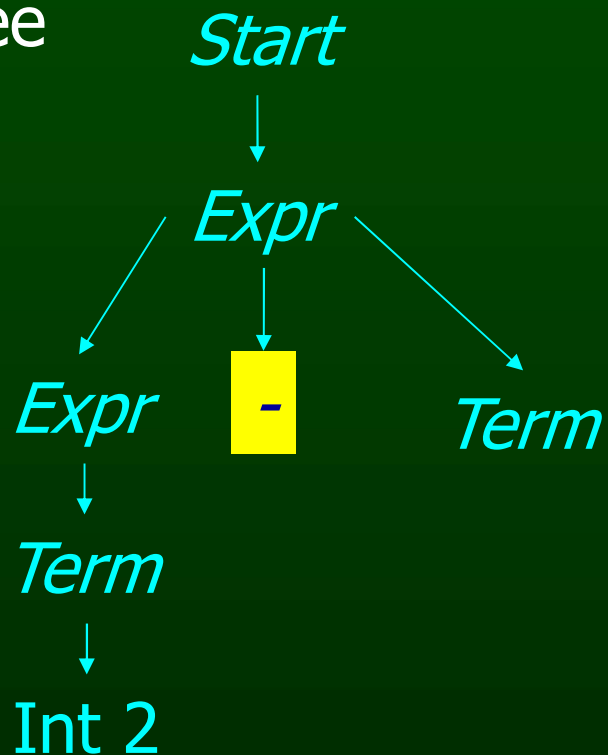
2-2*2

Rečenična forma

2 - Term

Primjer parsiranja

Parse
Tree



Match
Input
Token!

Preostali dio ulaza

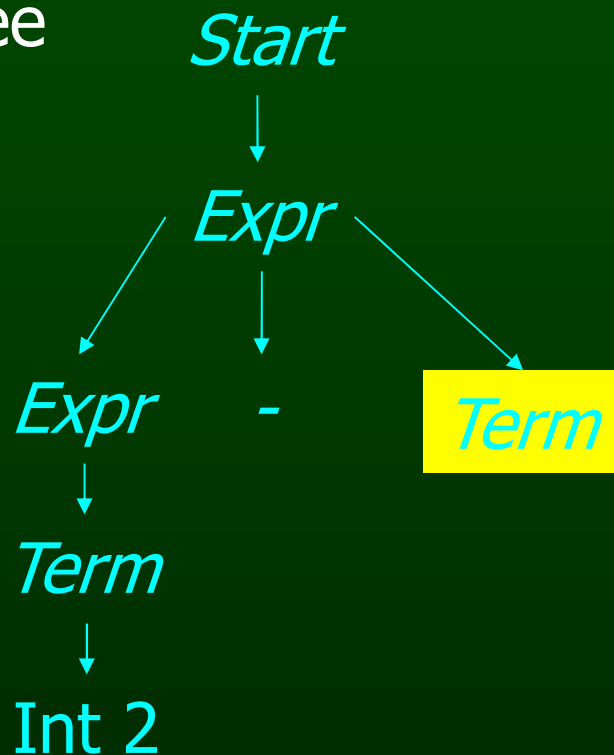
-2*2

Rečenična forma

2 - Term

Primjer parsiranja

Parse
Tree



Match
Input
Token!

Preostali dio ulaza

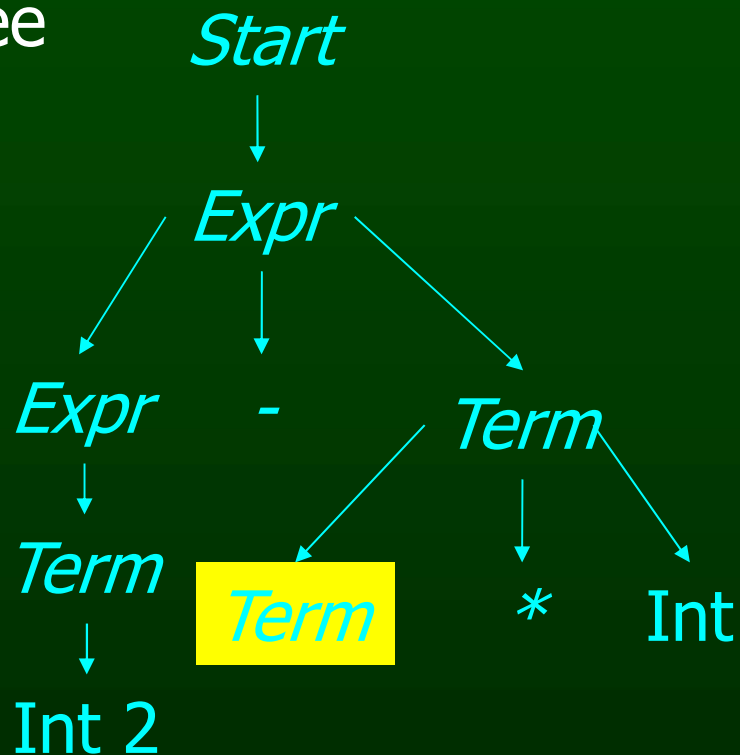
2*2

Rečenična forma

2 - Term

Primjer parsiranja

Parse
Tree



Preostali dio ulaza

2*2

Rečenična forma

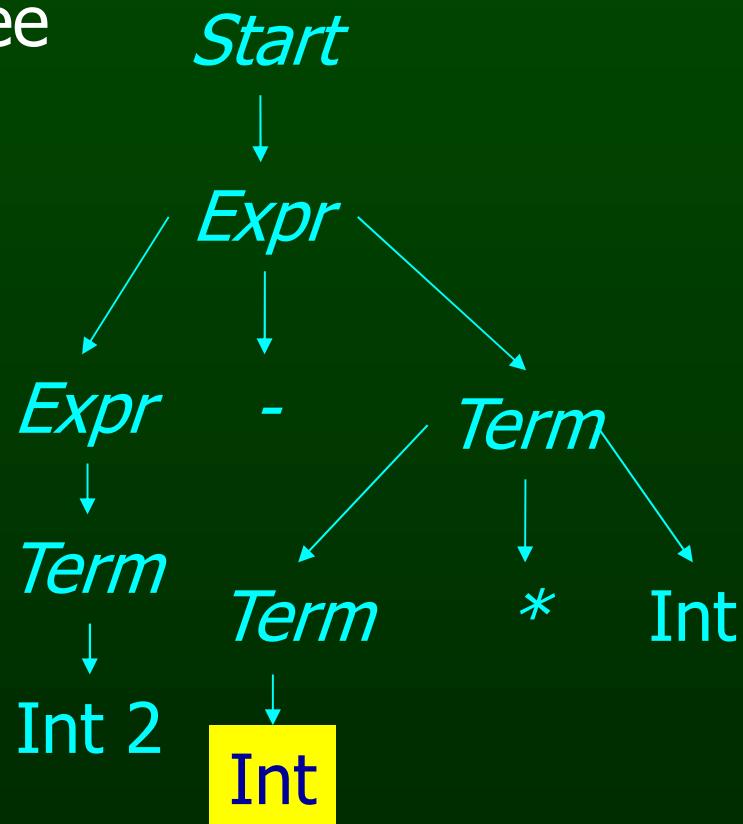
2 - *Term**Int

Primijenjeno pravilo

Term → *Term* * Int

Primjer parsiranja

Parse
Tree



Preostali dio ulaza

$2*2$

Rečenična forma

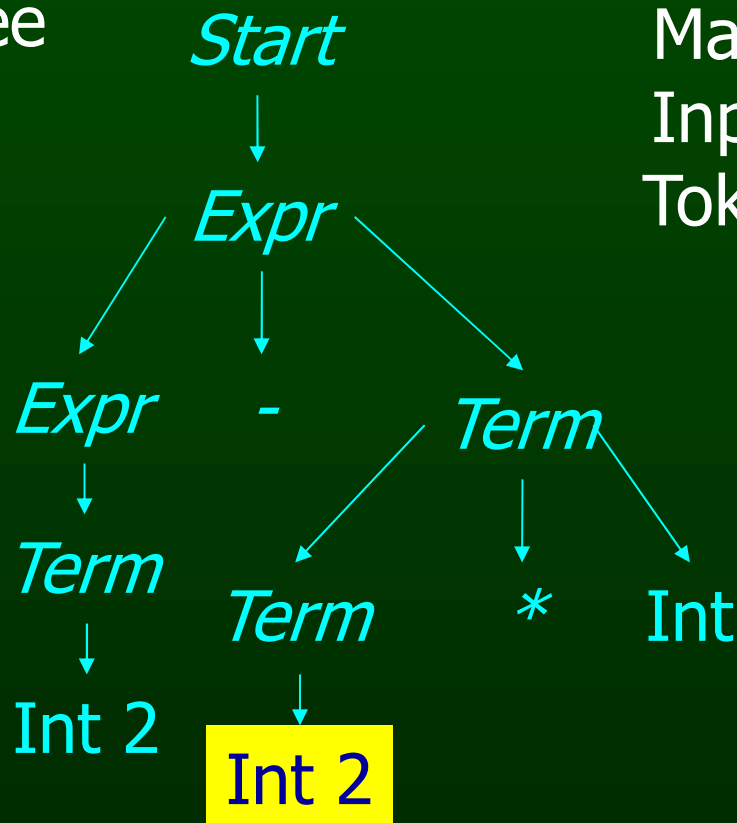
$2 - \text{Int} * \text{Int}$

Primijenjeno pravilo

$\text{Term} \rightarrow \text{Int}$

Primjer parsiranja

Parse
Tree



Match
Input
Token!

Preostali dio ulaza

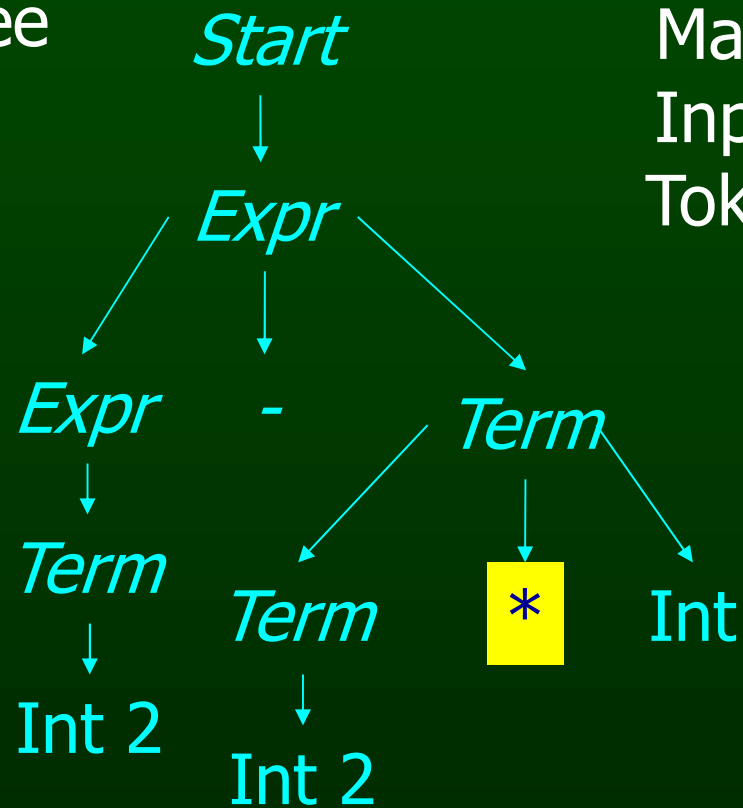
2*2

Rečenična forma

2 - 2 * Int

Primjer parsiranja

Parse
Tree



Match
Input
Token!

Preostali dio ulaza

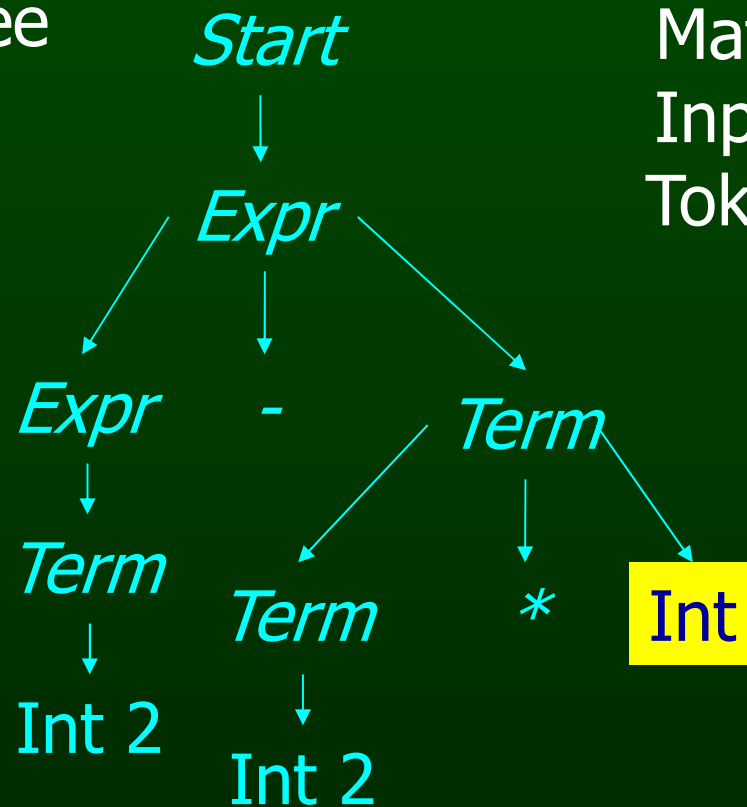
*2

Rečenična forma

2 - 2 * Int

Primjer parsiranja

Parse
Tree



Match
Input
Token!

Preostali dio ulaza

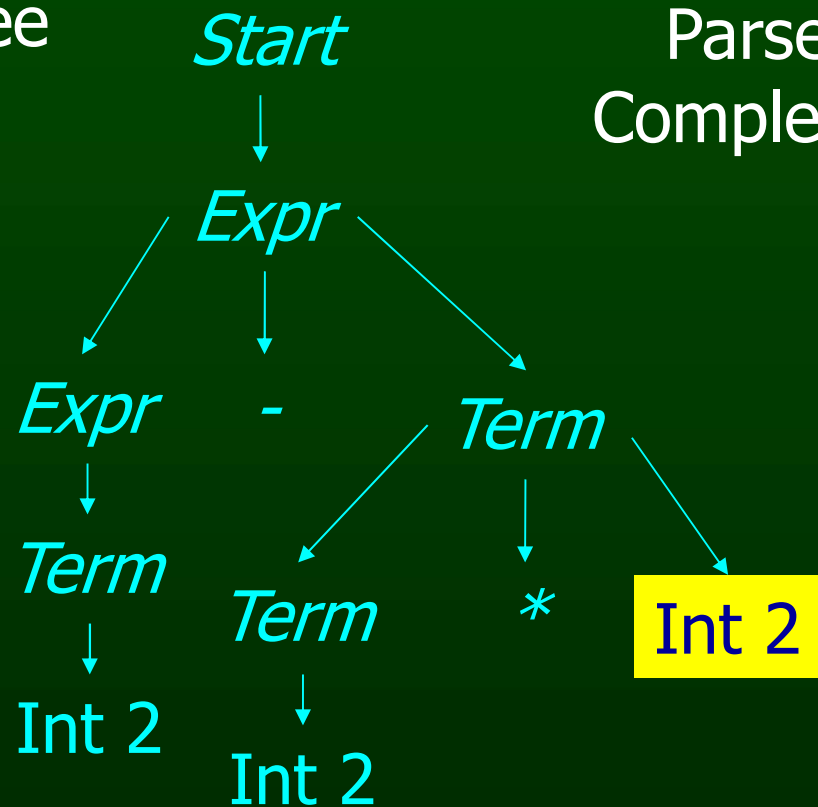
2

Rečenična forma

2 - 2 * Int

Primjer parsiranja

Parse
Tree



Parse
Complete!

Preostali dio ulaza

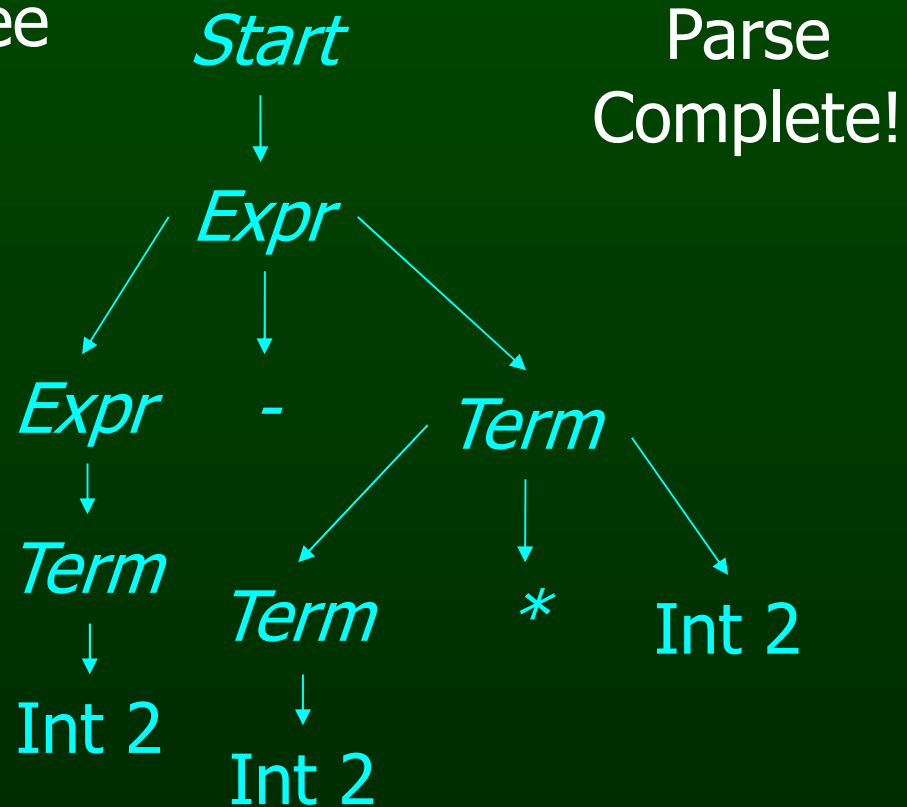
2

Rečenična forma

2 - 2 * 2

Primjer parsiranja

Parse
Tree



Preostali dio ulaza

Rečenična forma

2 - 2 * 2