Experiment #	12	Student ID	2060r
Date	2024	Student Name	S. K. S Reddy

TUTORIAL SESSION 12:

Normal Form: GNF

Concept Building

Greibach Normal Form (GNF)

It is a specific type of context-free grammar that simplifies the structure of production rules to facilitate parsing and theoretical analysis. It is named after Sheila Greibach, an American computer scientist who made significant contributions to formal language theory.

Characteristics of Greibach Normal Form

A context-free grammar is in Greibach Normal Form if every production rule adheres to the following form:

$A \rightarrow a\alpha$

Here, A is a non-terminal symbol, 'a' is a terminal symbol, and α (alpha) is a (possibly empty) string of non-terminal symbols. This form ensures that each production rule begins with a terminal symbol followed by zero or more non-terminals.

Importance of Greibach Normal Form

Algorithmic Parsing: GNF is particularly useful for constructing top-down parsers, such as recursive descent parsers, because it guarantees that each production begins with a terminal symbol, making it easier to decide which rule to apply based on the next input symbol.

Theoretical Analysis: GNF simplifies the analysis of context-free grammars by providing a uniform structure for production rules. This uniformity aids in proving properties about languages and grammars, such as closure properties and decidability.

Equivalence to Other Forms: Any context-free grammar can be converted to an equivalent GNF grammar that generates the same language. This conversion preserves the language while restructuring the grammar for easier parsing.

Steps to Convert a Context-Free Grammar to GNF

Converting a context-free grammar to GNF involves several steps, often requiring intermediate transformations and the introduction of new non-terminals.

Remove Left Recursion: Ensure that the grammar is free from left recursion, as left-recursive rules cannot be converted directly to GNF.

Eliminate Non-Productive and Unreachable Symbols: Remove any non-productive symbols (symbols that do not derive any terminal strings) and unreachable symbols (symbols that cannot be reached from the start symbol).

Ensure Proper Ordering: Ensure that the non-terminals in the grammar are ordered such that when converting each rule to GNF, the right-hand side contains only non-terminals that have already been processed.

Experiment #	12	Student ID	30604
Date	20/9	Student Name	S.R.S. Reddy

Pre-Tutorial (To be completed by student before attending tutorial session)

1. Convert the given CFG to GNF:

S→AB, A→BS | b, B→ SA | a

V → T

V → TV*

Solution:

(i) No epsilon transitions

(ii) No units transitions

(iii) S → AB A → b B → a

→ ba

{S, A, B}

A → BS

SAS

S → AB

→ ba {S, A, B}

(iv) A → b, B → a are GNF

S → AB	S → bB ✓
A → BS	A → aS
B → SA	B → Sb

(v) GNF

S → AB

A → aAB | b

B → bBA | a.

pl: { A → b | aS

B → Sb | b.

S → bB }.

Experiment #	12	Student ID	24604
Date	20/9	Student Name	S.K.Sreddy

2.Transform the following grammar into GNF. S→AA | 0, A→SS | 1

Solution:

Simplification

1) Epsilon transitions

P: {S→AA|0

A→SS|1}

No Epsilon

2) No unit transition

3) Useless transitions

S→AA A→SS

S→'11' →'00'

4) GNF

S→AA

⇒ S→SSSS|SS1|1SS|11

A→SS

⇒ A→AAAA0|AA0|0AA|00.

Final:

P:{ S→0|1SS|SS1|11|SSSS

A→00|0AA|AA0|AAAA }.

Experiment #	12.	Student ID	30607
Date	20/9	Student Name	G.R.S. Reddy

3.Find GNF equivalent for the given CFG:

S→ CA | BB, B→ b | SB, C→b, A→a

Solution:

Simplification

$P:\{S \rightarrow CA \mid BB$

$B \rightarrow b \mid SB$

$C \rightarrow b$

$A \rightarrow a\}$.

1) Epsilon transitions

No epsilon

2) Unit transition

No Unit transitions

3) Usesless transitions

$S \rightarrow CA \quad B \rightarrow 'b' \quad C \rightarrow 'b' \quad A \rightarrow 'a'$

$\rightarrow 'ba'$

4) GNF.

$P': \{S \rightarrow ba \mid bb$

$B \rightarrow b \mid baB \mid bbB$

$C \rightarrow b$

$A \rightarrow a\}$.

Experiment #	12	Student ID	3041
Date	20/7	Student Name	G. P. S. Reddy

IN-TUTORIAL (To be carried out in presence of faculty in classroom)

1. Convert the following grammar into Greibach Normal Form:

$E \to E+T \mid T,\ T \to T^*F \mid F,\ F \to (E) \mid a$

Solution:

Given

$S: \{ E \to E+T \mid T$

$T \to T^*F \mid F$

$F \to (E) \mid a \}.$

Simplification

(i) No Epsilon

(ii) Unit

$E \to T$ $E \to F$

$T \to F$

$E \to T^*F$ ✓

$T \to a \mid (E)$ ✓

$E \to a \mid (E)$ ✓

$\Rightarrow E \to E+T \mid a \mid (E) \mid T^*F$

$T \to T^*F \mid a \mid (E)$

$F \to (E) \mid a \}.$

(iii) Remove useless transition.

$E \to a \quad T \to a \quad F \to a$

No useless

(iv) Final GNF

$S: \{ E \to FT'E'$

$E' \to +FT'E' \mid$

$T \to FT'$

$T' \to *FT'$

$F \to (FT'E')) \mid a \}.$

Experiment #	12	Student ID	30681
Date	20/7	Student Name	S. B. S. Reddy

2.Convert the following CFG to GNF S → AB, A→ BS | a, B → SA | b.

Solution:

P: {S → AB

A → BS | a

B → SA | b }.

Simplification of CFG

(i) No nullable values

∴ No Epsilon transitions

(ii) Unit transitions

∴ No Unit transitions

(iii) Remove Useless transitions

S → AB A → a B → b

→ ab

∴ No useless transitions

(iv) Final GNF

P: {S → aSA | ab | BSSA | BSb

A → a | SAS | bS

B → b | SA }.

Experiment #	12	Student ID	21062
Date	20/9	Student Name	S.P.S.P

4. A grammar G is defined with rules $S \rightarrow XA \mid BB, B \rightarrow b \mid SB, X \rightarrow b, A \rightarrow a$. Write the productions obtained after normalized GNF of G.

Solution:

$V \rightarrow T$

$V \rightarrow TV^*$

P: { $S \rightarrow XA \mid BB$

$B \rightarrow b \mid SB$

$X \rightarrow b$

$A \rightarrow a$ }.

Simplification

1) Epsilon transitions

No Nullable Values

2) Unit Values

No Unit Values

3) Useless transitions

$S \rightarrow XA \quad B \rightarrow b \quad X \rightarrow \quad A \rightarrow a$

$\rightarrow$ 'ba'

4) GNF

Already GNF

$X \rightarrow b$ ✓

$A \rightarrow a$ ✓

$B \rightarrow b$ ✓

Final GNF

P1 : { ~~$S \rightarrow bSA \mid ba \mid BSA \mid bSA \mid a$~~

$S \rightarrow ba \mid bB \mid SBB$.

$B \rightarrow b \mid baB \mid bBB \mid aBB \mid BBB$

$X \rightarrow b$

$A \rightarrow a$ }.

Experiment #	12	Student ID	36604
Date	2019	Student Name	S.12.S.Reddy

Post-Tutorial (To be carried out by student after attending tutorial session)

1. **Convert the following CFG to GNF: S→ AB, A→ BSB, A→ a, B→ b**

Solution:

P: {S → AB

A → BS | B

A → a

B → b}

Simplification

1) Epsilon transition

No nullable values.

2) Unit productions

No Unit productions

3) Useless

S → AB A → BSB B → b.

→ 'ab' → bABb

→ babb

4) GNF

Already GNF Find GNF

A → a

B → b

P': {S → ba | b | SB | B

B → b | baB | bBB

X → a

A → a }

Experiment #	12	Student ID	30601
Date	20/7	Student Name	S.R.S. Reddy

2. Convert the following CFG to GNF : S→ A, A→ aBa | a, B → bAb | b

Solution:

P: {S → A
A → aBa | a
B → bAb | b }

Simplification

1) No epsilon

2) No Unit

3) Useless

S → A A → a
→ a' B → b

1) GNF

Already GNF

A → a
B → b

Final

P1 : {S → aBa | a
A → aBa | a
B → baBab | bab | b }.

3. Write the steps for removing null productions and unreachable symbols? Explain with an example of your own.

Solution:

1. Removing null productions

- Identify nullable non-terminals.
- For each nullable non-terminal, add new productions
- Remove the null productions.

2) Remove Unreachable Symbols

- Identify reachable symbols by starting from the start symbol and marking symbols used in productions

Experiment #	12	Student ID	30609
Date	20/9	Student Name	S.V.S.Reddy

Viva Questions

1. What is Greibach Normal Form (GNF) in the context of context-free grammars, and what are its characteristics?

Answer:

GNF is a special form of context-free grammars where every production rule starts with a terminal followed by zero or more terminals.

2. What are some practical applications of Greibach Normal Form in computer science?

Answer:

1. Top-Down parsing
2. Theoritical Analysis
3. Proving Grammer Correctness

Experiment #	12	Student ID	30 629
Date	20/9	Student Name	S.R.S.Reddy.

3. Can you describe the process of converting a context-free grammar into Greibach Normal Form?

Answer:

To convert a CNF to GNF

1) Start by removing left recursion

2) ensure every production starts with a terminal

$V \rightarrow T$

$V \rightarrow TV^*$

3) Replacing non-terminals with proper terminal first rules

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured: 50 out of 50 Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation: 21/9/24