

Compiler Design UNIT-2 PYQ Analysis **(RGPV)**

Syntax Analysis & Syntax Directed **Translation**

Most Repeated Questions Analysis (2017-2020)

Maine uploaded PYQ papers analyze kiye aur UNIT-2 se related sabhi questions extract kiye.
Ye unit RGPV me bohot important hai because:

- Parsing numericals aate hain
 - Grammar based questions repeat hote hain
 - LR parser almost har saal pucha gaya hai
-

UNIT-2 Topics

Syntax Analysis

- CFG
 - Top Down Parsing
 - Bottom Up Parsing
 - Recursive Descent Parsing
 - Predictive Parsing
 - Left Recursion
 - Left Factoring
 - Operator Precedence Parsing
 - LR Parser (SLR/LALR/CLR)
-

Syntax Directed Translation

- Syntax Trees
 - S-Attributed Definition
 - L-Attributed Definition
 - Top Down Translation
 - Bottom Up Evaluation
-

MOST REPEATED QUESTIONS

Question	Frequency	Importance
Difference between Top Down & Bottom Up Parsing	★★★★★	VERY HIGH
Explain LR Parser	★★★★★	VERY HIGH
Explain Operator Precedence Parser	★★★★★	HIGH
Remove Left Recursion	★★★★★	HIGH
Explain Recursive Descent Parsing	★★★★★	HIGH
Explain Predictive Parsing	★★★★★	HIGH
Construct Parse Tree	★★★	HIGH
Explain CFG	★★★	HIGH
Explain Syntax Tree	★★★	HIGH

YEAR-WISE PYQ ANALYSIS

DECEMBER 2020 PAPER

Q2(a)

Illustrate the steps in parsing the input:

$x = y + z - 5;$

by an LR parser using a predictive constructed LR parsing table.

 Related Topic:

- LR Parser
- Bottom Up Parsing

 VERY IMPORTANT

Q2(b)

Distinguish between top-down parsing and bottom-up parsing.

What is the largest class of grammars that can be parsed by each of them?

 Related Topic:

- Top Down Parsing
- Bottom Up Parsing

 MOST REPEATED QUESTION

Q7(a)

What are the fundamental differences between parse tree and abstract syntax tree?

 Related Topic:

- Parse Tree
- Syntax Tree

★ IMPORTANT

✓ Q7(b)

How does an Operator Precedence parser work?

Use a pre-constructed operator precedence table to guide parsing of:

$a+b-20$

📌 Related Topic:

- Operator Precedence Parsing

★ VERY IMPORTANT

🔥 **JUNE 2020 PAPER**

✓ Q2

Prove that Grammar is CLR but not LALR.

Grammar:

$S \rightarrow Aa/bAc/Bc/bBa$

$A \rightarrow d$

$B \rightarrow d$

📌 Related Topic:

- CLR Parser
- LALR Parser
- LR Parsing

★ VERY IMPORTANT

✓ Q3

Distinguish between top-down parsing and bottom-up parsing.

What is the largest class of grammars that can be parsed by each of them?

📌 Related Topic:

- Top Down Parsing
- Bottom Up Parsing

★ REPEATED AGAIN

✓ Q5

How does an Operator Precedence parser work?

Use a pre-constructed operator precedence table to guide parsing of:

$a+b-20$

📌 Related Topic:

- Operator Precedence Parsing

★ REPEATED QUESTION

DECEMBER 2017 PAPER

Q2(b)

Explain left recursion and show how it is eliminated.

Describe the algorithm used for eliminating left recursion.

 Related Topic:

- Left Recursion
- Grammar Transformation

 VERY VERY IMPORTANT

Q3(a)

Show that the following grammar is LR(1) but not LALR(1).

 Related Topic:

- LR Parser
- LALR Parser

 IMPORTANT

Q3(b)

What are the causes of backtracking in top down parser? Explain with example.

 Related Topic:

- Top Down Parsing

- Recursive Descent Parsing

★ IMPORTANT

✓ Q4

Assuming suitable syntax directed definition, construct a syntax tree for the expression:

4-4*e

📌 Related Topic:

- Syntax Tree
- Syntax Directed Definition

★ IMPORTANT

🔥 FINAL ANALYSIS (MOST IMPORTANT QUESTIONS)

★ TOP 7 MOST IMPORTANT QUESTIONS

1 Difference between Top Down Parsing and Bottom Up Parsing

🔥 Asked repeatedly

🔥 Highest probability

2 Explain LR Parser

🔥 Most important parser topic

3 Explain Operator Precedence Parsing

🔥 Numerical based repeated question

4 Explain Left Recursion and its elimination

🔥 Very frequent

5 Explain Recursive Descent Parsing

🔥 Important theory question

6 Explain Predictive Parsing

🔥 High chance

7 Construct Syntax Tree

🔥 Frequently asked

 **MOST IMPORTANT SMALL TOPICS**

Topic	Chance
Parse Tree	HIGH
Abstract Syntax Tree	HIGH
CFG	HIGH
Backtracking	HIGH
FIRST & FOLLOW	HIGH
SLR vs LALR vs CLR	VERY HIGH
Grammar Transformation	HIGH



MOST REPEATED PARSER

QUESTIONS

Parser Type	Importance
Top Down Parser	★★★★★
Bottom Up Parser	★★★★★
LR Parser	★★★★★
Operator Precedence Parser	★★★★
Recursive Descent Parser	★★★★
Predictive Parser	★★★★



PREDICTED QUESTIONS FOR

UPCOMING EXAM



VERY HIGH CHANCE

1. Explain Top Down and Bottom Up Parsing.
2. Explain LR Parser with example.
3. Explain Operator Precedence Parser.

4. Explain Predictive Parsing.
 5. Remove Left Recursion with algorithm.
-

★ HIGH CHANCE

6. Explain Recursive Descent Parsing.
 7. Construct Parse Tree and Syntax Tree.
 8. Difference between Parse Tree and Syntax Tree.
 9. Explain CFG with example.
-

★ MEDIUM CHANCE

10. Explain S-Attributed and L-Attributed Definitions.
 11. Explain Top Down Translation.
 12. Explain Bottom Up Evaluation.
-

⚡ SMART STUDY STRATEGY FOR

UNIT-2

If Exam Is Tomorrow 😊

MUST STUDY FIRST:

- ✓ Top Down vs Bottom Up Parsing
 - ✓ LR Parser
 - ✓ Operator Precedence Parser
 - ✓ Left Recursion
-

THEN STUDY:

- ✓ Predictive Parsing
 - ✓ Recursive Descent Parsing
 - ✓ Syntax Tree
-

LAST:

- ✓ SDD
 - ✓ L-Attributed
 - ✓ S-Attributed
-



ONE NIGHT REVISION PRIORITY

1. Top Down vs Bottom Up Parsing
2. LR Parser
3. Operator Precedence Parsing
4. Left Recursion
5. Predictive Parsing
6. Recursive Descent Parsing
7. Syntax Tree
8. Parse Tree
9. CFG