

MIDTERM-25/5/13

Q1) Considering string "ab", show that the following grammar is ambiguous using parse trees. The ϵ below is epsilon.

$S \rightarrow ABAB$

$A \rightarrow aA \mid \epsilon$

$B \rightarrow bB \mid \epsilon$

(5Marks)

Q2) Consider the following grammar;

Statement \rightarrow if expression then statement else statement

Statement \rightarrow if expression then statement

You are required to provide an alternate production(s) so that it may become free from the backtracking.

(5Marks)

Q3) Considering the following grammar, calculate the first set of non-terminals S, A and B.

$S \rightarrow AB$

$A \rightarrow a \mid \epsilon$

$B \rightarrow b \mid \epsilon$

(3 Marks)

Q4) How recursive descent parser is implemented using any Object Oriented language?

(3Marks)

Q5) Considering the following grammar, calculate the follow set of non-terminals S and A.

$S \rightarrow AB$

$A \rightarrow a \mid \epsilon$

$B \rightarrow b \mid \epsilon$

(2 marks)

Q6) Write a regular expression for the language of all words that starts and ends with different letters.

(2 Marks)

uff yar cs606 ka paper tha kay pata nhi kaya tha itna ajeeb paper kuch samaj nhi ah rahie the kay question main kaya poch raha hai or hum nay answer kaya dana hai

MCQs all from book from chapter 1 to chapter20

1-Core use of Symbol table

2-Buttom-up also called LR

3-Arithmetic Expression

$E \rightarrow id \mid Q$

$Q \rightarrow ER \mid \epsilon$

$R \rightarrow +Q \mid -Q \mid *Q \mid /Q$

4-Non-terminal batana tha first S A B ka

$S \rightarrow AB$

$A \rightarrow ab$

$B \rightarrow ba$

5-Scenario given tha ios ka algorithmic view lana tha

6-Scenario give tha ios ka final elimination Left recursive lana tha

Q. In two pass assembler what is the objective of First Pass? 2 marks

Q. What is Relocation? 3 marks

5 marks

Up frontier discovery useful for top down parser.

Parser tree 3 marks

one grammar 5 marks

=====mcqs=====

LR parsers can handle _____ grammars.

Left-recursive

file-recursive

End-recursive

Start-recursive

_____ convert the relocatable machine code into absolute machine code by linking library and relocatable object files.

1. Assembler
2. **Loader/link-editor**
3. Compiler
4. Preprocessor

Consider the grammar $A \rightarrow B C D$

$B \rightarrow h B \mid \epsilon$

$C \rightarrow C g \mid g \mid C h \mid i$

$D \rightarrow A B \mid \epsilon$

Follow of B is _____ .

h

g, h, i, \$

g, i

g

Consider the grammar $A \rightarrow B C D$

$B \rightarrow h B \mid \epsilon$

$C \rightarrow C g \mid g \mid C h \mid i$

$D \rightarrow A B \mid \epsilon$

Follow of C is _____ .

g, h, i, \$

g, h, \$

h, i, \$

h, g, \$

An important component of semantic analysis is _____ .

code checking

type checking

flush checking

None of the given

In PASCAL _____ represent the inequality test.

:=

=

\leq

None of the given

Lexical Analyzer generator _____ is written in Java.

1. Flex
2. **Jlex**
3. Complex
4. None of given

_____ avoid hardware stalls and interlocks.

1. Register allocation
2. **2. Instruction scheduling**
3. Instruction selection
4. None of given

Consider the following grammar, $A \rightarrow B C D B \rightarrow h B \mid \epsilon$ $C \rightarrow C g \mid g \mid C h \mid i D \rightarrow A B \mid \epsilon$ First of A is _____ .

h, g, i

g

h

None of the given..doubted

Recursive _____ parsing is done for LL(1) grammar.

Decent

Ascent

Forward

Backward

One of the core tasks of compiler is to generate fast and compact executable code.

True

False

Left factoring of a grammar is done to save the parser from back tracking.

True

False

Responsibility of _____ is to produce fast and compact code.

Instruction selection

Register allocation

Instruction scheduling

None of given

_____ algorithm is used in DFA minimization.

James's

Robert's

Hopcroft's

None of given

Compilers are sometimes classified as.

1. Single pass
2. Multi pass
3. Load and go
4. **4. All of the given**

Q1.linker user in compiler.2

Q2.regular expression that starts and end in diff words.2

Q3.remove non terminal from previous.3

Q4.parse table is given and we have to parse ab.3

Q5.left factoring.5

Q6.bottom up parsing .5

My paper was at 2:11pm

Mcq's 50 % from the shared

Purely conceptual paper. If you have time to kindly read and lister all the lecures carefully

aik 5 no ka ko case type question tha.

2nd 5 no ka sawal that us RE kay upar top bottom parser imposed karna thaa.

2 No. ka aik sawal bhi R.E ka tha

aik 3 Na ka

McQS 15 yahn jo pa jo mcqs han in main se hi thy

Q#1: how linker plays an important role with respect to compiler construction.(2 marks)

Q#2:define 2 pass compiler (2 marks)consider

Q#3: Draw LL(1) parsing table for the following grammar (5 marks)

$S \rightarrow aS|Ab$

$A \rightarrow XXZ|E$

$X \rightarrow ds|€$

$Y \rightarrow ds|€$

$Z \rightarrow es$

Q#4consider the following grammar (5 marks)

Statement \rightarrow if expression the statement.

Statement \rightarrow if expression the statement.

You are required provide an alternate production so that it may become from the backtracking.

Q#5:consider LL parse table(3 marks)

	a	b	\$
S	$S \rightarrow aB$	$S \rightarrow abB$	$S \rightarrow ab$
B		$B \rightarrow b$	$B \rightarrow €$

Q#6: consider the following grammar for arithmetic expressionsJ (3marks)

$E \rightarrow id|Q$

$Q \rightarrow ER|€$

$R \rightarrow +Q|-Q|*Q|/Q$

CS606
16-12-2012

Question No 1

What is the role of Automatic Code Generator with respect to compiler?

Question No2

How Linker play an important role with respects to compilers constructions?

Question No3

Consider the grammar for arithmetic expressions?

$E \rightarrow id|Q$

$Q \rightarrow ER \mid \epsilon$

$R \rightarrow +Q \mid -Q \mid *Q \mid /Q$

Show the follow set for all the non-terminal in the grammar

Question No 4

Consider the following grammar. Calculate the first set of non-terminal S, A and B

$S \rightarrow AB$

$A \rightarrow a \mid \epsilon$

$B \rightarrow b \mid \epsilon$

Question No 5

Consider the following grammar. You are required to wrote leftmost derivation of string “aebb”

$S \rightarrow aS \mid Ab$

$A \rightarrow XYZ \mid \epsilon$

$X \rightarrow cS \mid \epsilon$

$Y \rightarrow dS \mid \epsilon$

$Z \rightarrow eS$

Question No 6

Considering string “ab” shows that the following grammar is ambiguous using parse tree. The ϵ below is epsilon

$$\begin{array}{l}
 S \rightarrow ABAB \\
 A \rightarrow aA \mid \varepsilon \\
 B \rightarrow bB \mid \varepsilon
 \end{array}$$

Today my paper of cs606 at 12:00

few mcqz from moaaz file and mostly from handout.

Subjective:

Q1: given one expression and convert into Non terminals. 2 marks

Q2: top-bottom parser actions? 5 marks

Q3: given c++ code to check semantic error. 3 marks

16 mcq moaz file sy aye thy confirm hai and yeh aik dafa read kar ly 2 mcq is mein sy aye thy is ko read kary

The syntax of most programming languages is specified using Context-Free Grammars

(CFG). Context-free syntax is specified with a grammar $G=(S,N,T,P)$ where

- S is the start symbol
- N is a set of non-terminal symbols
- T is set of terminal symbols or words
- P is a set of productions or rewrite rules

MCQ's mostly from past papers..minimum 15 mcq's out of 20 were from past papers..

Subjective:

Q: given one expression and convert into Non terminals. 2 marks

Q: given c++ code to check semantic error.

Q: Considering the following grammar, calculate the follow set of non-terminals S and A.

$$S \rightarrow AB$$

$$A \rightarrow a \mid \varepsilon$$

$$B \rightarrow b \mid \varepsilon$$

Overall paper was easy..read the file shared aboved and moaz file for mcq's..

All the best.. :)

Q: Define Non-deterministic finite automata 2 marks

Q: how recursive descent parser is implemented using any Object Oriented Language

Q: tASK to develop a parser which has the parsing strategy wherein one first looks at the highest level of the parse tree and works down the parse tree by using an algorithm's idea for this task....

Q: calculate first set of Non terminal x, y, z and Follow set of NT x, y

$$x \rightarrow YZa \mid f$$

$$Y \rightarrow bY \mid d$$

$$Z \rightarrow cZ \mid e$$

Assalamualaikum. CS606 paper held today was comprised of easy MCQs and following subjective questions...

1. Phases of Frontend of a compiler. (2 marks)
2. Role of automated code generator. (2 marks)
3. Code was given to find semantic errors. (3 marks)
4. Convert following expression to non-left recursive. (3 marks)

$A \rightarrow Aa \mid b$

5. DFA (from lecture 8 & 9) was given. It was required to "make a transition table (only)" after minimizing DFA using Hopcroft's algorithm. (5 marks)
6. Consider string "ab". Show that the following grammar is ambiguous using parse trees. The ϵ below is epsilon. (5 marks)
 - $S \rightarrow ABAB$
 - $A \rightarrow aA \mid \epsilon$
 - $B \rightarrow bB \mid \epsilon$

Assalam o ALaikum

Mcqs from Maoz file 100%

Subjective:

Write a regular expression for the language of all words that starts and ends with different letters (2)

Phases Of Front End(2)

Ambiguous Grammer main sy bhi tha aik qstn (3)

Lec 8 k last main aik table hai wo bhi dyia hoa tha (5)

Lec 7 main Hopcroft Algorithm ka bh aik qstn tha (5)

Overall Paper bhoot easy tha

Good Luck For all of U

uff yar cs606 ka paper tha kay pata nhi kaya tha itna ajeeb paper kuch samaj nhi ah rahie the kay question main kaya poch raha hai or hum nay answer kaya dana hai

MCQs all from book from chapter 1 to chapter20

1-Core use of Symbol table

2-Buttom-up also called LR

3-Arithmetic Expression

$E \rightarrow id \mid Q$

$Q \rightarrow ER \mid \epsilon$

$R \rightarrow +Q \mid -Q \mid *Q \mid /Q$

4-Non-terminal batana tha first S A B ka

$S \rightarrow AB$

$A \rightarrow ab$

$B \rightarrow ba$

5-Scenario given tha ios ka algorithmic view lana tha

6-Scenario give tha ios ka final elimination Left recursive lana tha

Assalam o ALaikum

What does the compiler need to consider when applying optimization?

Define NFA?

In Main memory, what type of information is stored in data segement?

Identify whether the grammer is LR(1) and LL(1)?

$S \rightarrow Aa \mid Bb$

$A \rightarrow c$

$B \rightarrow c$

Characteristics of Live variables?

Algorithm to generate LALR(1) DFA?

Which information we get from 3rd section of YACC File?

How can compiler transformation improve program?

LL Parse table ?

INT and BOOI main sy bhi tha 5 marks ka ?

For each instruction, show which variable are live variable are live imidiately after instruction execution (5marks)

1. A=7

2. B=A+2

3. C=A+B

4.D=C+B

5.B=C+B

cs606

tree was given and preorder was asked?

what is non determininstic fa?

productions were given and canonical collection set was asked to built using productions.

action table and goto table were given we were asked to show in stack and input buffer

when we use garbage collection for memory management?

if on right hand side of production there are k symbols then how many positions are there for place holder?

production was given to determine whether this is of LR(1) or LL(1) ?

code was given and we have to find basic blocks?

use of address descriptor in code generation algorithm of 5 marks
