

**Computer Engineering Department, S.V.N.I.T., Surat.
End Semester Examinations, May 2015**

**B. Tech.-III (CO) – 6th semester
Systems Software (CO304)**

Dated: 6th May, 2015

Time: 1530hrs to 1830hrs

Max Marks: 100

Instructions:

1. Write your B Tech Admission No/Roll No and other details clearly on the answer books while write your B.Tech. Admission No on the question paper, too.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q.1.

- (a) Explain the followings w.r.t. Lex and YACC: yyparse(), yywrap(), yyval, yytext
- (b) Write an algorithm with example to construct DFA from NFA.
- (c) Fill the blanks B1 to B3 in the given LL(1) parsing table for the following grammar:

	a	b	\$
S	B1	B2	$S \rightarrow \epsilon$
A	$A \rightarrow S$	$A \rightarrow S$	Error
B	$B \rightarrow S$	$B \rightarrow S$	B3

$S \rightarrow aAbB \mid bAaB \mid \epsilon$
 $A \rightarrow S$
 $B \rightarrow S$

- (d) Calculate **First** and **Follow** of each non terminals for the following grammars:

- (i) $S \rightarrow aSe$
 $S \rightarrow B$
 $B \rightarrow bBe$
 $B \rightarrow C$
 $C \rightarrow cCe$
 $C \rightarrow d$
- (ii) $start \rightarrow expr$
 $expr \rightarrow id := expr$
 $expr \rightarrow term \ term_tail$
 $term_tail \rightarrow + \ term \ term_tail \mid emptystring$
 $term \rightarrow factor \ factor_tail$
 $factor_tail \rightarrow * \ factor \ factor_tail \mid emptystring$
 $factor \rightarrow (\ expr \) \mid id$

- (e) Is the following grammar LL(1)?, LR(0)?, SLR(1)?, CLR(1)?, LALR(1)? Justify your answer for each parser.

$S \rightarrow Aa \mid bAc \mid Bc \mid bBa$
 $A \rightarrow d$
 $B \rightarrow d$

- (f) Which of the following grammars are not LR (0)? Justify your answer.

- (i) $S \rightarrow StList\$$
 $StList \rightarrow StList ; Stmt$
 $StList \rightarrow Stmt$
 $Stmt \rightarrow null$

- (iii) $S \rightarrow StList$
 $StList \rightarrow StList ; StList$
 $StList \rightarrow Stmt$
 $Stmt \rightarrow null$
 $= \text{terminal}$

- (ii) $S \rightarrow StList\$$
 $StList \rightarrow Stmt ; StList$
 $StList \rightarrow Stmt$
 $Stmt \rightarrow null$

- (iv) $S \rightarrow StList\$$
 $StList \rightarrow null \ StTail$
 $StTail \rightarrow \lambda$
 $StTail \rightarrow ; \ StList$

- (g) Define an operator grammar. Write an operator precedence parsing algorithm.

- (h) With an appropriate code explain following: local and global sub expression elimination, propagation, code motion, frequency and strength reduction

- (i) For the following code snippet, generate Intermediate code and draw control flow diagram using basic blocks. 05

```
int fib (int m)
{ int f0 = 0, f1 = 1, f2, i;
  if (m <= 1)
    return m;
  else
    for (i=2; i<=m; i++)
    { f2 = f0 + f1;
      f0 = f1;
      f1 = f2;
    }
  return f2;
}
```

Q.2.

- (a) What is program relocation? Explain program relocation algorithm with example. 06
- (b) Enlist and explain task involved in macro expansion 07
- (c) Explain following terms: Cross assembler, Absolute loader, Bootstrap loader, execution of program 08
- (d) Following is an assembly language program for finding factorial of a given number N with Mnemonic code details. Write an equivalent machine language program. 07

	START 101	Mnemonics CODE
	READ N	STOP 00
	MOVER BREG, ONE	ADD 01
	MOVEM BREG, TERM	MULT 03
AGAIN	MULT BREG, TERM	MOVER 04
	MOVER CREG, TERM	MOVEM 05
	ADD CREG, ONE	COMP 06
	MOVEM CREG, TERM	BC 07
	COMP CREG, N	READ 09
	BC LE, AGAIN	PRINT 10
	MOVEM BREG, RESULT	LE 02
	PRINT RESULT	START 01
	STOP	END 02
N	DS 1	Ordinal number of
RESULT	DS 1	BREG and CREG is 2 & 3
ONE	DC '1'	respectively
TERM	DS 1	
	END	

- (e) Explain with example: How data structure of macro preprocessor is generated? 08
- (f) Describe following data structures with example: OPTAB, SYMTAB, LITAB and POOLTAB. 06

"You may never know what results come of your action, but if you do nothing there will be no result." - M. K. Gandhi

different code segments

11300043

Computer Engineering Department, SVNIT, Surat.
End Semester Examination, April 2016
B.Tech. 3rd year (COED) - 6th Semester
Systems Software (CO304)

Date: 26/4/2016
Instructions:

Time - 12:00 to 15:00

Total Marks - 100

1. Write Your B.Tech Admission No/Roll No and other details clearly on the answer books while write your B.Tech admission no. on the question paper, too.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q-1 ANSWER THE FOLLOWING

1 Consider following grammar :

58
10

1. $S \rightarrow S a S | \epsilon$
2. $S \rightarrow S | S a | \epsilon$
3. $S \rightarrow X c | Y$
 $Y \rightarrow a$
 $X \rightarrow a$
4. $P \rightarrow S$
 $S \rightarrow a | b B$
 $B \rightarrow e | S e | S B$

- a) For each grammar no. 1 & 2 is the grammar ambiguous? Why? Provide a brief explanation.
- b) For each grammar no. 3 & 4, is the grammar LL(1)? Derive appropriate tables to justify your answer.

- 2**
- a) Explain panic mode and phrase level error recovery with example.
 - b) The table with error routine is shown below. Routine e_1 when called, pushes an imaginary id into the input; and routine e_2 when called, removes all the remaining symbols from the input. Trace the behavior of the parser for the input id + * id \$

8

	id	+	*	\$
E	$E \rightarrow TE_1$	e_1	e_1	e_1
T	$T \rightarrow FT_1$	e_1	e_1	e_1
F	$F \rightarrow id$	e_1	e_1	e_1
E_1	$E_1 \rightarrow \epsilon$	$E_1 \rightarrow +TE_1$	$E_1 \rightarrow \epsilon$	$E_1 \rightarrow \epsilon$
T_1	$T_1 \rightarrow \epsilon$	$T_1 \rightarrow \epsilon$	$T_1 \rightarrow *FT_1$	$T_1 \rightarrow \epsilon$
id	pop			
+		pop		
*			pop	
\$	e_2	e_2	e_2	accept

3 What is Handle ? What is Handle Purning ?

8

For Following grammar in Shift Reduce Parsing indicate right most Sentential Form, Reducing Production and Handle to derive complete parsing Action.

Input String : id+id*id

Grammar :

$E \rightarrow E + T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid id$

*Handwritten: id id * id + id +*

4 Consider an input string $id + id * id + id$. 8

- Derive an operator precedence table to parse this string.
- Explain the significance for operator precedence function. Write an algorithm for the same.
- Derive the precedence function table and its appropriate graph representation.
- Parse the input string using the precedence function you derived.

5 What is the significance of three address code, triples, quadruples and indirect triples in code optimization? Explain them for the expression $X = (a+b) * -c/d$ 8

6 Consider a program below: 8

- | | | |
|-----------------|-----------------|------------------|
| 1. $a := 1$ | 6. $d := b * d$ | 11. $b := a + b$ |
| 2. $b := 2$ | 7. $goto(11)$ | 12. $e := c - a$ |
| 3. $c := a + b$ | 8. $d := a + b$ | 13. $goto(3)$ |
| 4. $d := c - a$ | 9. $e := e + 1$ | 14. $a := b * d$ |
| 5. $goto(8)$ | 10. $goto(6)$ | 15. $b := a - d$ |

- Write an algorithm to identify the basic blocks and build a control flow graph for the code.
- What code optimization technique can be used to optimize this code? Describe all suitable optimization method for the above program and its outcome of the optimized block.

7 Briefly explain and differentiate LR, SLR, CLR and LALR parsing techniques. 8

Q-2 ANSWER THE FOLLOWING

1 Explain the four component of the object module and Create object module of the following. If linked origin of the program P is 900 then what is the linked origin of program Q (10)

PROGRAM (P)

START 150

ENTRY ONE, TWO

EXTERN THREE, FOUR

150 READ FIVE

151 MOVER AREG, FIVE

152 ADD AREG, SIX

153 MOVER BREG, THREE

154 MULT BREG, SEVEN

155 SUB BREG, FOUR

156 STOP

FIVE DS 1

SIX DC '6'

SEVEN DC '7'

ONE DS 1

TWO DS 1

END

Handwritten: 56, 150 - 156, 906

PROGRAM (Q)

START 100

ENTRY THREE, FOUR

100 MOVER AREG, EIGHT

101 ADD AREG, NINE

102 STOP

THREE DC '3'

FOUR DC '4'

EIGHT DC '8'

NINE DC '9'

END

Handwritten: 100 - 102

id +

- 2 Generate Data structure of macro preprocessor with suitable example and explain which data structure constructed/used on which steps of macro. 8
- 3 What are the main features to facilitate alteration of flow control during expansion? Explain with suitable example 7
- 4 Write an algorithm of program linking. 6
- 5 Write a program to allocate sequential block of memory containing a value 56, 24, 16, 5 using expansion time loops facility 5
- 6 Enlist task involve in macro expansion and explain any three of it. 6

OR

- How single pass assembler resolves problem of forward reference with the help of TIL. Explain it with suitable example.

irp ang < 56, 24, 16, 5

Dated: 2nd May 2017

Engineering Department, S.V.N.I.T. Surat.
Semester Examination, May 2017
B Tech (CO) - IIIrd Year semester-VI
Course: System Software (CO304)

Time: 15:30 to 18:30

Max Marks: 100

Instructions:

1. Write your B Tech Admission No/Roll No and other details clearly on the answer books while write your B Tech Admission No on the question paper, too.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q1 Answer The Following Questions.(Any Three)

1. Explain the activities involved in Pass-I and Pass-II of Toy Compiler giving example. 15
2. Consider the grammar G (Front End and Back End or Analysis and Synthesis)
 <Sentence> ::= <Noun Phrase><Verb Phrase> 5
 <Noun Phrase> ::= <Article><Noun> 5
 <Verb Phrase> ::= <Verb><Noun Phrase> 5
 <Article> ::= a | an | the
 <Noun> ::= boy | apple
 <Verb> ::= ate
 a) Write the sentinel form of the above grammar.
 b) Perform necessary reduction.
 c) Generate Parse tree.
3. a) Write an algorithm to construct DFA from NFA. 5
 b) Draw NFA and DFA for the following : $((a^*ba^*)|(b^*ab^*))(aa|bb)$
4. Explain in detail the various phases of a compiler. Describe the output for the following expression after each phase, considering the precedence and associativity of the operators: $a:=b*c + c*b$ 5

Q2 Answer The Following Questions.

1. Describe the language denoted by the following regular expressions. 5
 a) $0(0|1)^*0$
 b) $((\epsilon|0)1^*)^*$
 c) $(0|1)^*0(0|1)(0|1)$
 d) $0^*10^*10^*10^*$
 e) $(00|11)^*((01|10)(00|11)^*(01|10)(00|11)^*)^*$
 Handwritten notes: $\rightarrow SA \rightarrow$
 $\rightarrow SP$
 $\rightarrow LC \rightarrow t_1 = id_3 * id_2$
 $t_2 = id_2 * id_3$
 $t_3 = t_1 + t_2$
 $id_1 = t_4$
 Tree diagram: $id_1 \rightarrow + \rightarrow id_2 * id_3$
2. What are the main operations for a symbol table? Discuss the data structures associated with a symbol table maintained as a list of hashtables and how the operations of a symbol table are implemented in that case. Give an example of what your symbol table would look like for a sample program. 5

Q3 Answer The Following Questions.

- Construct the following grammar
 $E \rightarrow E+T \mid T$
 $T \rightarrow TF \mid F$
 $F \rightarrow F* \mid a \mid b$

- Construct the SLR parsing table for this grammar.
- Construct the LALR parsing table.

- Show that the following grammar
 $S \rightarrow Aa \mid bAc \mid dc \mid bda$
 $A \rightarrow d$
 is LALR(1) but not SLR(1).

- For the following grammar

$S \rightarrow \{SX\} \mid a$
 $X \rightarrow \epsilon \mid +SY \mid Yb$
 $Y \rightarrow \epsilon \mid -SXc$

- Compute First and Follow sets for the non-terminals (terminals are $a, b, c, +, -, [,]$)
- Construct its LL(1) parsing table
 Is this grammar LL(1)?

- Consider the following grammar and parse the input string "id-id*id" using Shift
 Reduce parser LR(0).

$E \rightarrow E - E \mid E * E \mid id$

- Construct error-correcting operator-precedence and LR parsers for the following grammar:

$Stmt \rightarrow$ if e then stmt
 if e then stmt else stmt
 while e do stmt
 begin list end
 s

$List \rightarrow$ list ; stmt
 stmt

Q4 Answer The Following Questions.(Any Two)

The Pascal standard defines the statement

for $v :=$ initial to final do stmt

To have the same meaning as the following code sequence

```

Begin
  t1 := initial; t2 := final;
  if t1 <= t2 then begin
    v := t1;
    stmt
  while v != t2 do begin
    v := succ(v);
    stmt
  end
end
end
end
    
```

Construct a syntax-directed definition that generates correct three-address code for pascal for statements.

Dated: 2nd May 2018

Instructions:

1. Write your B Tech Admission No/Roll No and other details clearly on the answer books while write your BTech Admission No on the question paper, too.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q-1 Explain various phases of a compiler. Describe the output for the following statement, after each phase of a compiler. [04]
 $a = a + b * c * 2.$

Q-2 Answer The Followings: [08]

1. Describe the languages denoted by the following regular expressions: [02]

1) $(aa+ab+ba+bb)^*$

2) $b^* + a^* + (ba)^*$

3) $a^* bba^*$

4) $((ca) b^*)^*$

2. How do the parser and scanner communicate? Explain with block diagram. Also discuss how lexical errors are recovered? [02]

3. What is handle and handle pruning? Explain with example. [02]

4. Eliminate Left recursion for following: [02]

$A \rightarrow ABd \mid Aa \mid a$

$B \rightarrow Be \mid b$

Q-3 Answer The Followings: (Any Four) [12]

1. Is the following grammar suitable for LL(1) parsing? If not, make it suitable for it. Compute first and follow. Generate LL(1) parsing table. [03]

$S \rightarrow AB$

$A \rightarrow Ca \mid \epsilon$

$B \rightarrow BaAC \mid c$

$C \rightarrow b \mid \epsilon$

[03]

Check whether the grammar is LR(0), SLR(1)?

$S \rightarrow dA / aB$

$A \rightarrow bA / c$

$B \rightarrow bB / c$

Write down functions for non-terminals of the given grammar to implement Recursive decent parser. [03]

$S \rightarrow ABC$

$A \rightarrow 0A1 \mid \wedge$

$B \rightarrow 1B \mid \wedge$

$C \rightarrow 1C0 \mid \wedge$

Consider the following grammar and parse the input string "id+id*id" using operator precedence parser. [03]

$E \rightarrow E + E \mid E * E \mid id$

6. Define: handle and handle pruning.
7. (a) Giving a valid reason justify that the following grammar is not suitable for LL (1) parsing.
- (b) After making it suitable for LL(1) parsing, generate parsing table.

$S \rightarrow AB$
 $A \rightarrow Ca | \epsilon$
 $B \rightarrow BaAC | \epsilon$
 $C \rightarrow b | \epsilon$

Q2. Answer The Following Questions. (Any Two)

1. Define Cross Assembler, Meta Assembler and Micro Assembler.
2. IRP Statement With Suitable Example.
3. What is the significance of object module? List out all the components of object module.

Q3. Answer The Following Questions. (Any Two)

1. Differentiate Non Relocatable programs, Relocatable programs and Self-relocating program. Write an algorithm of program linking.
2. What is an overlay? Explain overlay structured program and its execution.
3. Explain following advanced macro facilities with example:
 1. Alteration of flow of control during expansion
 2. Attributes of formal parameter

Q4. Answer The Following Questions.

1. Write a macro which takes A, B as Positional Parameter, C, D as Keyword parameters and calculates $A = A/B + C \cdot D$ in AREG. Also, Generate Data Structures of Macro which you have created.
2. What are the advantages of assembly language over machine level language? Generate Machine Level Language of the given assembly language program.

```

START      200
MOVER      AREG, =5'
MOVEM      AREG, X
MOVER      BREG, =2'
ORIGIN     L1+3
LTORG

L1
NEXT
  ADD      -5
  SUB      -2
  RC       AREG, =1'
  LTORG   BREG, =2'
          IT, BACK

BACK
  EQU      -1
  ORIGIN   -2
  MULT     L1
  STOP     NEXT+5
  DS       CREG, =4'
  END      1

```

ADD - 01, SUB-02, MULT - 03, MOVER - 04, MOVEM - 05, COMP-06, BC-07, DIV-08, READ - 09, PRINT - 10, STOP - 00.
 START-01, END-02, ORIGIN-03, EQU-04, LTORG-05.
 LT-1.

Computer Engineering Department
End Semester Examination: M.Sc -III-Mathematics (6th Semester)
Subject: Systems Software, Code: CO304

Marks: 50

Time: 12:00 to 15:00

Date: 7th may, 2018.

Instructions:

1. Write your Admission No/Roll No and other details clearly on the answer books while write your Admission No on the question paper, too.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q. Answer the following [30]

1. Consider the following grammar : [08]

$E \rightarrow E + T$
 $E \rightarrow T$
 $T \rightarrow T * F$
 $T \rightarrow F$
 $F \rightarrow (E) \mid id$

(a) Construct the collection of sets of LR(0) items for this grammar. [03]

(b) Construct the parsing table using the SLR algorithm. [03]

(c) Show all the moves allowed by the parsing table on input $id * id + id$. [02]

2. Construct the canonical parsing table for the following grammar: [04]

$S \rightarrow S$
 $S \rightarrow CC$
 $S \rightarrow cC \mid d$

3. Show that the following grammar is not LALR(1) grammar. [04]

$S \rightarrow S$
 $S \rightarrow aAd \mid bBd \mid aBe \mid aAe$
 $A \rightarrow c$
 $B \rightarrow c$

(a) Define operator grammar. Comment whether the given grammar is operator grammar or not with valid reason. [02] [05]

(b) For the given grammar generate operator-precedence relations and parse the input string : $id + id + id$ [03]

Grammar : $E \rightarrow E + E \mid E * E \mid id$

4. Compute First and Follow for the following grammar: [02]

$S \rightarrow AcB \mid CbB \mid Ba$
 $A \rightarrow da \mid BC$
 $B \rightarrow g \mid \epsilon$
 $C \rightarrow h \mid \epsilon$

6. Define: handle and handle pruning.
7. (a) Giving a valid reason justify that the following grammar is not suitable for LL (1) parsing.
- (b) After making it suitable for LL(1) parsing, generate parsing table.

$S \rightarrow AB$
 $A \rightarrow Ca \mid \epsilon$
 $B \rightarrow BaAC \mid \epsilon$
 $C \rightarrow b \mid \epsilon$

Q2. Answer The Following Questions. (Any Two)

1. Define Cross Assembler, Meta Assembler and Micro Assembler.
2. IRP Statement With Suitable Example.
3. What is the significance of object module? List out all the components of object module.

Q3. Answer The Following Questions. (Any Two)

1. Differentiate Non Relocatable programs, Relocatable programs and Self-relocating program. Write an algorithm of program linking.
2. What is an overlay? Explain overlay structured program and its execution.
3. Explain following advanced macro facilities with example:
 1. Alteration of flow of control during expansion
 2. Attributes of formal parameter

Q4. Answer The Following Questions.

1. Write a macro which takes A, B as Positional Parameter, C, D as Keyword parameters and calculates $A=A/B+C \cdot D$ in AREG. Also, Generate Data Structures of Macro which you have created.
2. What are the advantages of assembly language over machine level language? Generate Machine Level Language of the given assembly language program.

```

START      200
MOVER      AREG, =5'
MOVEM      AREG, X
MOVER      BREG, =2'
ORIGIN     L1+3
LTORG

L1
NEXT      -5
          -2
          AREG, =1'
          BREG, =2'
          LT, BACK

BACK      -1
          -2
          L1
          NEXT+5
          CREG, =4'

X
STOP
DS
END        1
  
```

ADD - 01, SUB-02, MULT - 03, MOVER - 04, MOVEM - 05, COMP-06, BC-07, DIV-08, READ - 09, PRINT - 10, STOP - 00.
 START-01, END-02, ORIGIN-03, EQU-04, LTORG-05, LT-1.

Supplementary Exam – Jan-2018
B.Tech. IIIrd year - VIth Semester
Systems Software (CO304)

Seat No - _____
Total Marks : 50

Date: 30/01/2018

Time : 10:00 to 1:00

Instructions:

1. Write your B.Tech Admission No and other details clearly on the answer book and on the question paper.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q-1 ANSWER THE FOLLOWING **25**

1. Explain Left recursion and Left factoring top down parsing. **5**
i. Eliminate left recursion from following grammar.

$S \rightarrow A$

$A \rightarrow Ad \mid Ae \mid aB \mid aC$

$B \rightarrow bBC \mid f$

$C \rightarrow g$

ii. Solve below example by left factoring.

$S \rightarrow \text{if } E \text{ then } S \mid \text{if } E \text{ then } S \text{ else } S \mid a$

$E \rightarrow b$

2. Develop an LL(1) parser table for the following grammar and Parse the string using the parsing table : $(id * id) + (id * id)$ **5**

$E \rightarrow TA$

$A \rightarrow +TA \mid \epsilon$

$T \rightarrow VB$

$B \rightarrow *VB \mid \epsilon$

$V \rightarrow id \mid (E)$

3. Construct NFA and DFA for following regular expression: $(0 \mid 1)^* 001\#$ **5**

OR

Construct a DFA without constructing NFA for the following regular expression.

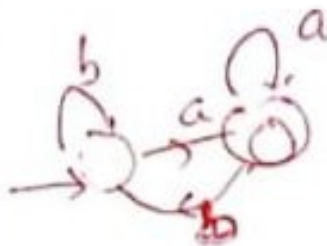
$(a \mid b)^* a$

4. What is peephole optimization? Enlist and explain techniques applied in peephole optimization. **5**

5. Construct LR(0) parsing table for the below grammar **5**

$S \rightarrow AA$

$A \rightarrow aA \mid b$



Q-2 ANSWER THE FOLLOWING

25

1. Consider following assembly language program:
Show (i) Contents of Symbol Table (ii) Intermediate codes using Variant 1 representation.

5

AGAIN	START	101
	READ	N
	MOVER	BREG, ONE
	MOVEM	BREG, TERM
	MULT	BREG, TERM
	MOVER	CREG, TERM
	ADD	CREG, ONE
	MOVEM	CREG, TERM
	COMP	CREG, N
	BC	LE, AGAIN
	MOVEM	BREG, AGAIN
	PRINT	RESULT
	STOP	
	DS	
N	DS	1
RESULT	DC	1
ONE	DS	'1'
TERM	END	1

Instruction opcode: STOP - 00, ADD - 01, MULT - 03, MOVER - 04, MOVEM - 05, COMP - 06, BC - 07, READ - 09, PRINT - 10, LE - 02
Declaration statements: DC - 01, DS - 02
Register code: BREG - 02, CREG - 03

2. Define two macros of your choice to illustrate nested calls to these macros. Also show their corresponding expansion. 5
3. Define forward references. How it can be solved using back-patching? Explain with example. 5
4. Explain use and field of following tables of macro. 5
5. KPDTAB, MDT, EVTAB, SSTAB. 5
- Define self-relocating program and write an algorithm of program relocation. 5
- OR
- What is an overlay? Explain overlay structured program and its execution.