Seat No.: _____ PR No._____

SANKALCHAND PATEL UNIVERSITY

B.Tech – SEMESTER (5) – EXAMINATION – WINTER 2018

Subject Code:1ET1030503 Date:25/10/ 2018

Subject Name: System Programming

Time: 3 Hrs. Total Marks: 70

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q. 1 Answer the following Short Questions. 20

1. Define: Language Processor.
2. Define: Handle, Grammar
3. How to avoid backtracking in top down parsing?
4. Define: Macro Assembler.
5. _____ Phase perform type checking task.
6. Define: Parse tree.
7. State True/False: Symbol table is used to store mnemonics and opcodes.
8. Define: Operator Precedence.
9. What is DFA?
10. Define: Semantic Gap.
11. What is the use of backpatching?
12. Describe the output of Lexical analysis.
13. Justify: True/ False: A language migratory provides portability of program.
14. What is JIT?
15. State True/False: “Single pass assemblers cannot handle forward references.”
16. Which phase of compiler will generate error if semicolon is missing in a c program?
17. Define: Ambiguous grammar with example.
18. Difference between Compiler and Interpreter..
19. Define: Triples.
20. Define: Assembler.

Q. 2 Answer the following questions. 14

A Explain Left recursion, Left factoring and backtracking in top down parsing?

B Differentiate one pass and two pass assembler. Explain how forward references are handle in two pass assembler.

OR

A Differentiate passes and phases of compiler. Explain in brief syntax analysis phases.

B Explain in brief the design of a macro preprocessor.

Q. 3 Answer the following questions. (Attempt any two) 12

A Explain macro expansion process with example.

B Construct operator precedence parser for following grammar:
E -> E + E | E * E | id

C List out various assembler directives. Explain any three with suitable example.

Q. 4 Answer the following questions. **12**

A Given a grammar

S -> Aa | bAc | Bc | bBa

A -> d

B -> d check whether the grammar is LL (1) or not?

B Explain different code optimization techniques.

OR

A Define forward references. How it can be solved using back-patching? Explain with example.

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

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

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

Q. 5 Answer the following questions. (Attempt any two) **12**

A Consider the statement a = b + c * 10.where a, b, c are of type float. Show the translation of the given statement by different phases of compiler to produce assembly language statements.

B Compare Variant I and Variant II of intermediate code generator for assembler?

C Explain Triples and Quadruples with an example.

D What are the advanced macro facilities? Explain With Example.

****

Seat No.: _____ **PR No._____**

SANKALCHAND PATEL UNIVERSITY

B. Tech – SEMESTER (5) – EXAMINATION – SUMMER 2019

Subject Code: 1ET1030503 **Date:26/04/ 2019**

Subject Name: System Programming

Time: 3 Hrs. **Total Marks: 70**

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q. 1 Answer the following questions.(Each of one mark) **20**

1 Define: Back patching

2 Define: Simple phrase grammar

3 Define: Left recursion

4 Define: Topdown parsing

5 _____ phase performs type checking task.
[a] Lexical Analysis [b] Syntax Analysis [c] Semantic Analysis

6 State True/False: "Code optimization phase is optional phase of compiler"

7 State True/False: "Top Down parsers do not work with left recursive grammar"

8 What is symbol table?

9 What is Semantic Gap?

10 State True/False: "Single pass assemblers cannot handle forward References"

11 State True/False: "Symbol table is used to store mnemonics and opcodes"

12 Define: Macro Assembler.

13 What is the use of POOLTAB in assembler?

14 What is JIT?

15 Describe the syntax of keyword parameter and the use of it.

16 Difference between Literal and Constant.

17 Justify (True/False): "The language migrator provides portability of program"

18 Which of system software resides in main memory? What input it takes?

19 A _____ is a language processor which bridges an execution gap but,It is not a language translator.

20 Define: Pattern

Q. 2 (A) Eliminate left recursion from following grammar **3**

$S \rightarrow A$

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

$B \rightarrow bBC \mid f$

$C \rightarrow g$

(B) Construct LL(1) parsing table for following grammar **4**

$S \rightarrow iCtSeS \mid iCtS \mid a$

$C \rightarrow b$

	(C)	Construct an optimized DFA : 0*1*(0/1)#	**7**

OR

	(C)	Prove that following regular expressions are equivalent by constructing the optimized DFA. (0/1)* (0*/1*)*	**7**
Q. 3	**(A)**	What is Peephole optimization? Explain any two optimization transformations in detail.	**3**
	(B)	State different storage allocation strategies. Explain static allocation and stack allocation in detail.	**3**
	(C)	Write operator precedence table for arithmetic operators "+", "*", "-", "/" "(", ")". Parse following expression using the table id * (id + id)/ (id *id)	**6**

OR

	(A)	How does compiler implement scope rules?	**3**
	(B)	An assembly program contains the statement X EQU Y + 25 Indicate how does EQU statement can process if (1) Y is a back reference (2) Y is a forward reference	**3**
	(C)	Explain in detail how does following input get processed in toy compiler. int a; real b,c; c= a + b * 0.6;	**6**
Q. 4	**(A)**	Define: L-Attributed definition in detail.	**3**
	(B)	Write and explain the algorithm for macro expansion.	**3**
	(C)	Write algorithm for practical approach of top down parsing	**6**

OR

	(A)	Explain pure and impure interpreter.	**3**
	(B)	Explain the front end of toy compiler with suitable example.	**3**
	(C)	Draw the expression tree for the string f+(x+y)*((a+b)/(c-d)) by their evaluation order and mention register required label in each node.	**6**
Q. 5	**(A)**	Explain absolute loader in detail.	**3**
	(B)	Explain Ambiguous grammar with example .	**3**
	(C)	Explain Naïve Bottom up parsing algorithm with example and also mention which problem occurs during parsing.	**6**

OR

	(A)	Compare and Contrast macro preprocessor and macro assembler.	**3**
	(B)	What is overlay? Explain the execution of an overlay structure program.	**3**
	(C)	What is the use of static pointer and dynamic pointer in compiler? Explain working of Display with suitable example.	**6**

****

Seat No.: _____ PR No._____

SANKALCHAND PATEL UNIVERSITY

B. Tech – SEMESTER (5) – EXAMINATION – WINTER 2019

Subject Code: System Programming Date:14/11/ 2019

Subject Name: 1ET1030503

Time: 3 Hrs. Total Marks: 70

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q. 1 Answer Following Questions (Each of 01 Mark) 20

1. What is the use of backpatching?
2. Justify (True/False): A language migrator provides portability of program.
3. Define: symbol Table.
4 .Define: Semantic Gap.
5. Define: Back patching.
6. Define: Macro Assembler.
7.Define:Scanning
8.Define:Parsing
9.Define:RegularGrammer
10.Define:Regular Expression
11.Define:DFA
12. Justify (True/False): The Macro can be compiled or Interpreted.
13. Define: Interpreter.
14. Justify (True/False): A Program containing forward references cannot be assembled in a single pass.
15. .How does the language specification gap can be bridged?
16. Why Preprocessing is required?
17. Model statements of the macro can be stored in MNT. True/False?
18. What is the use of pool table in Assembler?
19. Why CRT is required in Assembler?
20. What is the use of cross compiler?

Q. 2 Answer Following Questions .

A	List language processing activities. Explain Synthesis Phase	3
B	Define Macro Expansion? Discuss two different ways of Macro Expansion	4
C	Consider following assembly program. Show (i) Contents of Symbol Table (ii) intermediate codes using Variant I representation	7

```
START 100
READ A
READ B
READ C
MOVER AREG, A
ADD AREG, B
ADD AREG, C
MULT AREG, C
MOVEM AREG, RESULT
```

```
PRINT RESULT
STOP
A DS 1
B DS 1
C DS 1
RESULT DS 1
END
```

Instruction opcodes: READ – 09, MOVER – 04, MOVEM – 05, ADD – 01, MULT – 03, PRINT – 10, STOP – 00 Assembler-directive codes: START – 01, END – 02 Register code: AREG – 01

OR

	C	Define forward references. How it can be solved using back-patching? Explain with example	7
Q. 3	A	Given a grammar, E → TA A →+TA | ε T →VB B →*VB | ε V→id | (E) Develop an LL (1) parser table and parse the string: id * (id + id).	5
	B	Construct an optimized DFA : 0*1*(0/1)#	7

OR

	A	What is Peephole optimization? Explain any two optimization Transformations in detail.	5
	B	Explain design of different types of loaders.	7
Q. 4	A	What are the advanced macro facilities? Explain with suitable example.	5
	B	Explain design of an Editor and list types of debugger.	7

OR

	A	Describe the use of REPT and IRP statement.	5
	B	Explain in brief design of the linker.	7
Q. 5	**Attempt any Three (Each of 4 Mark)**		
	A	Write Macro: Print square of 1 to 20 numbers	4
	B	Write Macro: Evaluate amount= rent * NoOfDays + guidecharges	4
	C	Write a macro to duplicate the set of statements 5 times.	4
	D	Write an Assembly program for the addition of 3 numbers and write corresponding Mnemonic code for it.	4

****

Seat No.: _____ PR No._____

SANKALCHAND PATEL UNIVERSITY

B. Tech. – SEMESTER (5) – EXAMINATION – WINTER 2021

Subject Code: 1ET1030503 **Date:20/11 / 2021**

Subject Name: System Programming

Time: 3 Hrs. **Total Marks: 70**

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q. 1 Answer the following Short Questions 20

(1) Define Regular Grammar
(2) Define Regular Expression
(3) Define Language processor
(4) Define Language translator
(5) Define DS (Declare Storage) and DC (Declare Constant).
(6) Define Preprocessor
(7) Difference between Macro and Subroutine
(8) Define Semantic gap
(9) Define Specification gap
(10) Define Execution Gap

Q. 2 Answer the following questions. 14

(A) What is system programming? Differentiate system software and application software. Give name of different kind of system software.

(B) List various phases of a language processor. Explain roles of first two phases of it. Also explain symbol table.

OR

(A) What is Assembler? Which are the basic features of assembler?

(B) Explain analysis and synthesis phases of an assembler by clearly stating their tasks.

Q. 3 Answer the following questions. (Any Two) 12

(A) Explain Left Recursion, Left Factoring and Backtracking in Top-down parsing with suitable example.

(B) Define a macro. Explain the concept of macro definition and macro call.

(C) What is Macro Expansion? Discuss two different ways of Macro Expansion.

Q. 4 Answer the following questions. (Any Two) 12

(A) What are the advanced macro facilities? Explain With Example.

(B) Explain Phases of Compiler.

(C) Explain Triplets and Quadruples with an example.

Q. 5 Answer the following questions. (Any Three) 12

(A) Explain types of grammar.

(B) Explain Absolute loader with example.

(C) Explain self-relocating programs and overlay structured programs.

(D) Define the Debugging Procedures?

****

Seat No.: _____ PRN_____

SANKALCHAND PATEL UNIVERSITY

B. Tech – SEMESTER (5) – EXAMINATION – SUMMER 2022

Subject Code: 1ET1030503 **Date:26/05/2022**

Subject Name: System Programming

Time: 3 Hrs. **Total Marks: 70**

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q. 1 Answer the Following Questions. 20

1. Differentiate System Software and Application Software. 4
2. Define (1)Assembler (2)Parsing (3) Interpreter 4)Macro 4
3. Differentiate between passes and phases of complier. 4
4. Explain use and field of following tables of a macro KPDTAB, MDT, EVTAB, SSTAB 4
5. Explain Following terms with suitable example. (1) Expansion Time Variable (2) Positional Parameter. 4

Q. 2 Answer the following questions. 14

(A) What is interpreters explain pure and impure interpreters. 7

(B) Given the source program: 7

```
          START   100
A         DS      4
L1        MOVER   AREG, =‘5’
          ADD     AREG, B
          MOVEM   AREG, A+2
D         EQU     A+1
L2        PRINT   D
          ORIGIN  L2+1
          MOVER   CREG,C
          ADD     CREG,=’1’
          MOVEM   CREG,A+3
          LTORG
          MOVER   DREG ,=’1’
          ADD     DREG,C
          MOVEM   DREG,A
          PRINT   A+2
          STOP
D         DS      1
C         DS      1

          END
```

(i) Show the contents of the symbol table, literal table & pool table.
(ii) Show the intermediate code generated for the program.

OR

(A) Given the Grammar, evaluate the string a1-(a2+a3) using shift reduce parser. 7

$S \rightarrow S+S$

$S \rightarrow S-S$

$S \rightarrow (S)$

$S \rightarrow a$

(B) List various phases of language processor. Explain any one phase in Detail. 7

Q. 3 Answer the following questions. (Any Two) 12

(A) What is program relocation? How relocation is performed by linker? Explain with example. 6

(B) By taking the example of factorial program explain how activation record will look like for every recursive in case of factorial (3). 6

(C) What is memory binding? Explain dynamic memory allocation using Extended stack model. 6

(D) Explain design of editor. 6

Q. 4 Answer the following questions. (Any Two) 12

(A) Explain recursive descendant parsing algorithm. 6

(B) Explain design of linker in brief. 6

(C) Explain types of grammar with example. 6

(D) Explain triple and quadruple representation with example. 6

Q. 5 Answer the following questions. (Any Two) 12

(A) Explain with examples - expansion time variables, expansion time Statements, AIF and AGO for macro programming. Show their usage for expansion time loop by giving example. 6

(B) List and explain advance macro facility with suitable example. 6

(C) Explain lexical analysis of language processor also Explain any three Code Optimization Techniques. 6

(D) Discuss Operator preceding parser with illustration of example. 6

****