CBCS SCHEME

USN □□□□□□□□□□ **15CS63**

## Sixth Semester B.E. Degree Examination, Dec.2018/Jan.2019
## System Software and Compiler Design

Time: 3 hrs. Max. Marks: 80

***Note: Answer any FIVE full questions, choosing ONE full question from each module.***

Important Note : 1. On completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages.
2. Any revealing of identification, appeal to evaluator and /or equations written eg. 42+8 = 50, will be treated as malpractice.

### Module-1

1 a. Explain in detail SIC/XE machine architecture. **(08 Marks)**

b. Write an SIC/XE program to calculate DELTA = ALPHA + BETA * GAMMA – 10 **(08 Marks)**

**OR**

2 a. Write an algorithm for Pass – 1 of an assembler. **(08 Marks)**

b. Generate the object code for the following SIC/XE source program.

```
SUM     START   0
FIRST   CLEAR   X
        LDA     #0
        +LDB    #TOTAL
        BASE    TOTAL
LOOP    ADD     TABLE, X
        TIX     COUNT
        JLT     LOOP
        STA     TOTAL
COUNT   RESW    1
TABLE   RESW    2000
TOTAL   RESW    1
        END     FIRST
```

| Mnemonic | ADD | JLT | LDA | LDB | LDX | RSUB | STA | TIX | JSUB | J | LDT | CLEAR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| opcode | 18 | 38 | 00 | 68 | 04 | 4C | 0C | 2C | 08 | 3C | 74 | B4 |

**(08 Marks)**

### Module-2

3 a. Write PASS-1 and PASS-2 algorithm for a linking loader. **(08 Marks)**

b. Explain dynamic linking, automatic library search, loader design options with suitable examples. **(08 Marks)**

**OR**

4 a. Write the SIC/XE program for a bootstrap loader with suitable comments. Explain in brief the algorithm of a bootstrap loader. **(08 Marks)**

b. Explain in brief (i) MS-DOS linker and (ii) CRAY MPP linker. **(08 Marks)**

### Module-3

5 a. List and explain the various phases of a compiler and show the output of each phase for the expression a := b + c * 25 **(08 Marks)**

b. Construct transition diagram for recognizing relational operators. Sketch the program segment to implement it, showing the first state and one in final state. **(08 Marks)**

**OR**

6 a. Explain input buffering strategy used in lexical analysis phase. **(06 Marks)**

b. Write the regular definition for unsigned number, also write the transition diagram. **(06 Marks)**

c. Construct the transition diagrams for a set of keywords like begin, end, if then and else and identifiers and constants along with a minimum set of relational operators. **(04 Marks)**

## Module-4

7 a. What is shift reduce parser? Explain the conflicts that may occur during shift reduce parsing. **(04 Marks)**

b. Construct LALR parsing tables for the grammar shown below using LR(1) items.

$S' \to S$
$S \to Cc$
$C \to cC \mid d$ **(08 Marks)**

c. How left recursion can be eliminated from grammars? Write down the simple arithmetic expression grammar and rewrite the grammar after removing left recursion. **(04 Marks)**

**OR**

8 a. What is left factoring? Rewrite the following grammar after "left factored"

$S \to iEts \mid iEtSeS \mid a$
$E \to b$ **(04 Marks)**

b. Write a note on the parser generator – yacc. **(04 Marks)**

c. Construct canonical LR(1) items for the augmented grammar

$S' \to S :$
$S \to Cc$
$C \to cC \mid d$ **(08 Marks)**

## Module-5

9 a. Define synthesized and inherited attributes with examples. **(04 Marks)**

b. Briefly explain the main issues in code generation. **(08 Marks)**

c. Explain in brief dead code elimination. **(04 Marks)**

**OR**

10 a. Construct DAG for the expression

$a + b * (a + b) + c + d$ **(04 Marks)**

b. Give SDD of a simple calculator. **(04 Marks)**

c. Write a note on common sub expression. **(04 Marks)**

d. What are the steps involved in optimization of basic blocks. Explain any 2 steps in brief. **(04 Marks)**

* * * * *