SYSTEM SOFTWARE
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – VI

Subject Code	15IS652	IA Marks	20
Number of Lecture Hours/Week	3	Exam Marks	80
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course objectives: This course will enable students to

- Define System Software such as Assemblers, Loaders, Linkers and Macroprocessors
- Familiarize with source file, object file and executable file structures and libraries
- Describe the front-end and back-end phases of compiler and their importance to students

Module – 1	Teaching Hours
Introduction to System Software, Machine Architecture of SIC and SIC/XE. **Assemblers:** Basic assembler functions, machine dependent assembler features, machine independent assembler features, assembler design options. **Macroprocessors:** Basic macro processor functions, machine independent macro processor features, Macro processor design options, implementation examples **Text book 1: Chapter 1: (1.1-1.3.2), Chapter2: 2.1- 2.4 ,Chapter4**	08 Hours
Module – 2	
Loaders and Linkers: Basic Loader Functions, Design of an absolute loader, a simple Bootstrap loader, Machine-dependent loader features-relocation, program linking, algorithm and data structures for a linking loader, Machine –independent loader features-automatic library search, Loader options, loader design options-linkage editor, dynamic linkage, bootstrap loaders, implementation examples-MS DOS linker. **Text book 1 : Chapter 3**	08 Hours
Module – 3	
System File and Library Structure: Introduction, Library And File Organization, Design Of A Record Source Program File Structure, Object Code, Object File, Object File Structure, Executable File, Executable File Structure, Libraries, Image File Structure. **Object Code translators:** introduction, binary code translators, object code translators, translation process, hybrid method, applications **Reference 1: chapter 5 and chapter 15**	08 Hours
Module – 4	
Lexical Analysis: Introduction, Alphabets And Tokens In Computer Languages, Representation, Token Recognition And Finite Automata, Implementation, Error Recovery. **Text book 2: Chapter 1(1.1-1.5), Chapter 3(3.1-3.5)**	08 Hours
Module – 5	
Syntax Analysis: Introduction, Role Of Parsers, Context Free Grammars, Top Down Parsers, Bottom-Up Parsers, Operator-Precedence Parsing **Text book 2: Chapter 4 (4.1 – 4.6)**	08 Hours
Course outcomes: The students should be able to:	

- Explain system software such as assemblers, loaders, linkers and macroprocessors
- Design and develop lexical analyzers, parsers and code generators
- Utilize lex and yacc tools for implementing different concepts of system software

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012
2. Compilers-Principles, Techniques and Tools by Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman. Pearson, 2nd edition, 2007

Reference Books:

1. Systems programming – Srimanta Pal , Oxford university press, 2016
2. System software and operating system by D. M. Dhamdhere TMG
3. Compiler Design, K Muneeswaran, Oxford University Press 2013.
4. System programming and Compiler Design, K C Louden, Cengage Learning

H.O.D.
Dept. Of Information Science & Engineering
Alva's Institute of Engg. & Technology
Mijar, MOODBIDRI - 574 225