

Roll No.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Total No. of Pages : 02

Total No. of Questions : 09

B.Tech.(CSE)/(IT) (Sem.-4)
SYSTEM PROGRAMMING
Subject Code : CS-210
Paper ID : [A0462]

Time : 3 Hrs.

Max. Marks : 60

INSTRUCTION TO CANDIDATES :

1. SECTION-A is COMPULSORY consisting of TEN questions carrying TWO marks each.
2. SECTION-B contains FIVE questions carrying FIVE marks each and students have to attempt any FOUR questions.
3. SECTION-C contains THREE questions carrying TEN marks each and students have to attempt any TWO questions.

SECTION A

1. Write briefly :
 - a. What are the elements of assembly language programming?
 - b. Define memory binding.
 - c. Define local optimization and global optimization of program.
 - d. Define overlays.
 - e. Define line editor.
 - f. Define absolute loader.
 - g. What is the difference between compiler and interpreter?
 - h. What is the use of symbol table?
 - i. Explain finite automata and its uses.
 - j. What is the difference between parse tree and syntax tree?

SECTION-B

2. What are the limitations of stack based memory allocation? Also discuss the advantages of array based allocation.
3. Comment on the following statement : “*There would be no need for linkers if all programs are coded as self relocating programs*”.
4. Explain the role of YACC.

5. Explain the steps involved in dynamic debugging of a program.
6. Explain the typical mechanism contained in the OS kernel.

SECTION-C

7. Explain the four step approach to develop a design specification for an assembler.
8. Write short notes on :
 - a) Booting techniques
 - b) Design of shell
9. Build a program flow graph for the following program :

```
z := 5;
w := z;
for i := 1 to 100 do
    x := a*b;
    y := c+d;
    if y < 0 then
        a := 25;
        f := c+d;
    else
        g := w;
        h := a*b+f;
        d := z+10;
```

end;

```
g := c+d;
```

```
print g, h, d, x, y;
```

Apply the following transformations to optimize the program :

- a) Common subexpression elimination
- b) Dead code elimination
- c) Constant propagation
- d) Frequency reduction