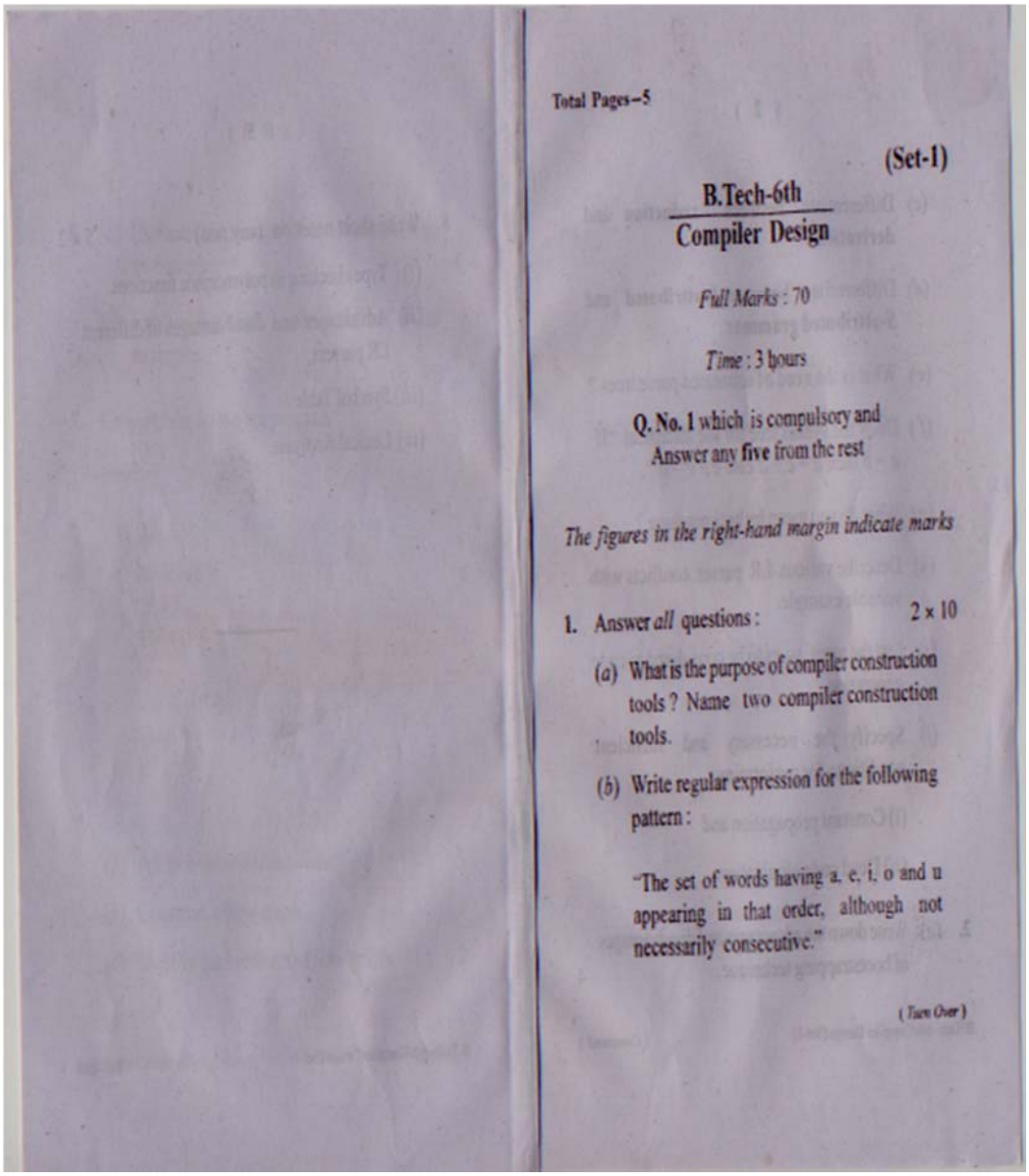
VEER SURENDRA SAI UNIVERSITY OF TECHNOLOGY, BURLA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING & INFORMATION TECHNOLOGY

SESSION 2014-2015 (EVEN SEMESTER)



Total Pages–5

(Set-1)

B.Tech-6th

Compiler Design

Full Marks : 70

Time : 3 hours

Q. No. 1 which is compulsory and Answer any **five** from the rest

The figures in the right-hand margin indicate marks

1. Answer *all* questions : 2 × 10

(*a*) What is the purpose of compiler construction tools ? Name two compiler construction tools.

(*b*) Write regular expression for the following pattern :

"The set of words having a, e, i, o and u appearing in that order, although not necessarily consecutive."

(*Turn Over*)

(c) Differentiate between **reduction** and **derivation.**

(d) Differentiate between ***L*-attributed** and ***S*-attributed grammar.**

(e) What is the need of annotated parse trees ?

(f) Draw the syntax tree for the statement "If $a = b$ then $a = c + d$ else $c = c - d$".

(g) What do you mean by backpatching ?

(h) Describe various LR parser conflicts with suitable example.

(i) List the main issues to be considered in code generation.

(j) Specify the necessary and sufficient conditions for performing :

(i) Constant propagation and

(ii) Dead code elimination.

2. (a) Write down the advantages and disadvantages of bootstrapping technique. 4

(b) Explain in detail the different phases of compilation with the following input string :

"$X = b * b - 4 * a * c$". 6

3. Find NFA, DFA and minimized DFA for the regular expression :

"$b * ab * ab$". 10

4. (a) Explain how to compute LEADING and TRAILING with examples. 4

(b) Construct the operator precedence parse table for the following grammar and show its shift-reduce actions for input string "abab".

$S \rightarrow aSbS|bSaS|\varepsilon$ 6

5. (a) What is a DAG ? How does it differ from abstract syntax tree (AST) ? 4

(b) Explain syntax directed definition for Boolean expressions with and without backpatching. 6

6. (*a*) Explain static storage allocation scheme with an example. 4

(*b*) Explain global optimization with examples. 6

7. Consider the following program :

```
main ( )
{
    int i ;
    int a[10];
    i = 1 ;
    while (i < = 10)
    {
        a[i] = 0;
        i = i + 1;
    }
}
```

(*a*) Produce three-address code. 5

(*b*) Construct a flow graph. 3

(*c*) Identify the loops in the flow graph. 2

8. Write short notes on (any *two*) : 5 × 2

(*i*) Type checking in polymorphic functions.

(*ii*) Advantages and disadvantages of different LR parsers.

(*iii*) Symbol Table

(*iv*) Lexical Analysis.

———