Reg. No. : .....

Name : .....

Sixth Semester B.Tech. Degree Examination, May 2016
(2008 Scheme)
08.601 : COMPILER DESIGN (R/F)

Time : 3 Hours Max. Marks : 100

PART – A

Answer **all** questions.

1. Explain the structure of a compiler. 4
2. Explain the relation between compilers and computer architecture. 4
3. Describe the languages denoted by the following regular expressions : 4
   i) a(a|b)*a
   ii) a*ba*ba*ba*
4. Formally define first and follow associated with a grammar G. 4
5. With an example, explain transition diagrams for predictive parsers. 4
6. With a diagram explain the model of a table driven parser. 4
7. With an example, explain the concept of handles and handle pruning. 4
8. Explain the applications of Syntax directed translation. 4
9. Construct the DAG for the expression
   ((x + y) – ((x + y) * (x – y))) + ((x + y) * (x – y)). 4
10. Explain the issues in the design of a code generator. 4

(10×4=40 Marks)

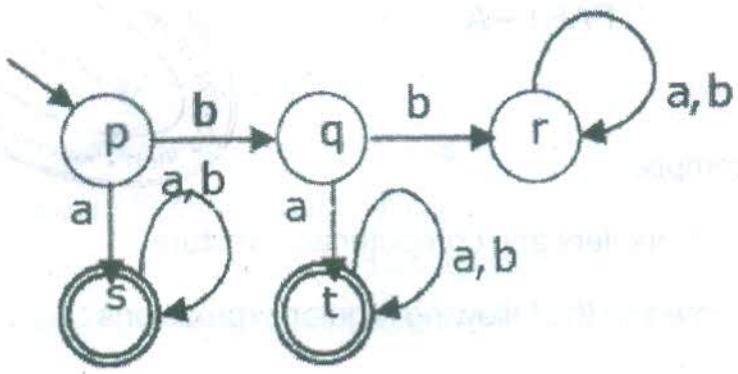
P.T.O.

PART – B

Answer **any one full** question from **each** Module.

Module – 1

11. a) A Deterministic Finite Automation (DFA) D with alphabet $\Sigma = \{a, b\}$ is given below. Derive a valid minimal DFA by applying the algorithm in step by step, which accepts the same language as D. **10**



b) Divide the following C++ program into appropriate lexemes. Which lexemes should get associated lexical values ? What should those values be ? **9**

```
f l o a t ls(x) f l o a tx {
/* returns x-squares, but never more than 100*/
return (x<= –10.01 || x >=10.00)?100:x*x;
}
```

c) The lexical analysis for a modern computer language such as Java needs the power of one of the theoretical machine models in automata theory. Name that machine. **1**

OR

12. a) Write regular definitions for the following language : **6**

- All strings of a's and b's with an even number of a's and an odd number of b's.

b) Convert the following regular expressions to deterministic finite automata, using direct conversion technique. 9

$(a|b)^*abb(a|b)^+$

c) Draw a DFA to accept all the strings in Σ, where $\Sigma = \{a, b\}$. 5

Module – 2

13. a) Consider the context-free grammar

$S \rightarrow SS+|SS^*|a$

i) Show how the string aa + a* can be generated by this grammar. 2

ii) Construct a parse tree for this string. 2

iii) What language does this grammar generate ? Justify your answer. 2

b) Describe all the viable prefixes for the following grammar. 13

$S \rightarrow S(S)|e$.

c) Name two theoretical machine models used for parsing with different expressive power. 1

OR

14. a) Construct the SLR sets of items for the (augmented) grammar given below. Compute the GOTO function for these sets of items. Show the parsing table for this grammar. Is the grammar SLR ? 10

$S \rightarrow SS+|SS^*|a$

b) Construct recursive-descent parser, for the following grammar : 9

$S \rightarrow +SS|-SS|a$

c) Which of the following suffices to convert an arbitrary CFG to an LL(1) grammar ? 1

i) Removing left recursion alone

ii) Factoring the grammar alone

iii) Removing left recursion and factoring the grammar

iv) None of the above.

Module – 3

15. a) Construct a syntax-directed translation scheme that translates post-fix arithmetic expressions into equivalent infix aithmetic expressions. 8

b) Write a program segment for matrix multiplication in C and translate into three address statements. Assume that matrix entries are numbers that require 8 bytes and the matrices are stored in row major order. 9

c) Why is the code optimization carried out on the intermediate code ? 3

OR

16. a) Write notes on the following. 16

i) Optimization of basic blocks

ii) Flow of control optimization

iii) Register allocation and assignment

iv) Machine independent optimization.

b) Assume that only two registers are available in the instruction set architecture of a processor. The code motion moves the statements from one place to another while preserving correctness. The only allowed complier optimization is code motion. In that case, in the complied code, what is the minimum number of spills to memory ? Justify your answer. 4