KATHMANDU UNIVERSITY
End Semester Examination [C]
July, 2017

JUL 1 4 2017

Level	: B. E.	Course	: COMP 409
Year	: IV	Semester	: I
Time	: 2 hrs. 30 mins.	F. M.	: 40

SECTION "B"

[6Q. × 4 = 24 marks]

Attempt *ANY SIX* questions.

1. What is the use of deterministic finite automata in lexical analysis? Also explain input buffering scheme.

2. Explain with specific example how tokens are specified. Also explain with subroutine how recognition of token is carried out during lexical analysis phase.

3. Consider the following piece of code:

```
c = 10;
a = c;
i = j = 0;
while(i != (a*3))
{
        if (j) print("Hello");
        i++;
        d[i] = 10*i;
        b = 5 * I;
}
```

Identify different kinds of optimizations possible. Rewrite the code after making optimized. [4]

4. Assign the attribute and write semantic action to the following grammar. [2+2]

S → E$
E→E + E
E → E*E
E → I
I →I digit
I → digit

Also construct translation tree for the string **54 *5 + 22** where digit → [0 - 9]

5. Given a statement **x = (a+b) * - c/d** of a three address code scheme, represent this into Quadruple, Triple and Indirect Triple Representation method.

6. Write a regular expression over an alphabet {a,b} such that it generate a string in which the first symbol is either **a** or **b**, second symbol is **b**, third symbol is **a** and it might end with **aa** or **bb** or both. Augment the regular expression and finally convert it to equivalent DFA using syntax tree. [4]

7. Write Short notes (*ANY TWO*) with suitable example:
   a. Recovery Mode b. Attribute Grammar c. ε-NFA d. Symbol Table

SECTION "C"

[2 Q × 8 = 16 marks]

Attempt *ANY TWO* questions.

8. Consider the given grammar G:

G: { E → T
E →E + T
T →int
T → (E) }

Construct LR(1) parsing table. Test whether the string w = int + (int + int) can be parsed or not.

9. Prove the statement "Every SLR (grammar) is unambiguous but every unambiguous grammar cannot be solved by SLR parsing" using the following grammar. [8]

1. S → L = R
2. S→ R
3. L → *R
4. L → id
5. R→L

10. Consider the following grammar:

S →SAa | SBb | a | b
A → b
B → a

Construct the predictive top down parsing table. Test whether the string w = baab is valid or not. [8]