UNIVERSITY OF SWAZILAND
SUPPLEMENTARY EXAMINATION, JULY 2006

Title of Paper : THEORY OF COMPUTATION

Course number : CS 211

Time allowed : Three (3) hours.

Instructions : (1) Read all the questions in Section-A and Section-B before you start answering any question.
(2) Answer all questions in Section-A. Answer **any two** questions of Section-B. Maximum mark is 100.
(3) Use correct notation and show all your work on the answer script.

This paper should not be opened until the invigilator has granted permission.

SECTION-A (maximum marks 50)

QUESTION 1(a) (marks 6 + 6+ 6). The following languages are given on symbol set {a, b}. For each language, write four examples of words of different lengths such that two of them belonging to the language and two of them not belonging to the language. Design appropriate deterministic finite acceptors (**dfa's**) that accept these languages. Assume that – u, w ∈ $\{a, b\}^+$ and $\lambda \notin$ L1, L2 or L3.

(i). L1 = {a w b}

(ii). L2 = {u w , |u| = 2}

(iii). L3 = {w, |w| is always an odd integer}

QUESTION 1(b) (marks 6). Write regular expressions representing L1, L2 and L3 languages of Q1(a).

QUESTION 2(a) (marks 4 + 8 + 8) The following non deterministic finite acceptor (**nfa**) is given :

M = ({q0, q1, q2}, {a, b}, q0, δ, {q1}), where δ is given as :

δ(q0, a) = {q0, q1} ; δ(q0, b) = q0 ;

δ(q1, a) = q2 ; δ(q1, b) = q1 ;

δ(q2, a) = q2 ; δ(q2, b) = q2

(i). Draw the transition digraph of the **nfa.**

(ii). Write the corresponding right linear grammar generating L(M).

(ii). Convert the above **nfa** into an equivalent **dfa**.

Q2(b) (marks 6). Write a transition digraph of the **dfa** and corresponding right linear grammar for the following language

$$L = \{a^3 b^{2n} \; ; n \geq 0 \}$$

SECTION-B (maximum marks 50)

Note: Answer **any two** questions in this section.

QUESTION 3(a) (marks 5 + 5). A Context Free Grammar (CFG) for simple arithmetic expressions is given as –

G = ({E, T, I}, {a, b, c, +, *, (,)},E, P)

where the set of productions P is

```
{        E  →  T | E + T ,

         T  →  I | ( E ) | T * I | T * ( E ) ,

         I  →  a | b | c          }
```

Write left most and right most derivations for the expression, a + (b * c) writing the rule number used at each derivation step. Draw the corresponding derivation tree. What is the complexity of G ?

QUESTION 3(b) (marks 15). Find Context Free grammars (**CFG**) G1 and G2 that generate the following languages. Assume $n \geq 0$ and $m \geq 0$ -

(i). L(G1) = $\{a^{2n} b^n c^m\}$
(ii).L(G2) = $\{a^n b^m c^m d^n\}$

Test your CFG 's by writing left most derivations for -

w = aaaabb (using G1) and w = aabbbcccdd (using G2).

Include production number at each step of derivation.

QUESTION 4(a) (marks 15). Design a deterministic pushdown automaton (**dpda**) to recognize the language –

L = { w, such that w always starts with one or more a's, followed by only b's. Count of a's and b's are equal }

Describe the functional steps of your **dpda**. Write instantaneous descriptions for w = aaabbb.

Q4(b) (marks 10). Design a non deterministic pushdown automaton (**npda**) to recognize the language generated by the following grammar in Greibach Normal Form–

```
G = ({S, A, B}, {a,b},S, P)
```

where the set of productions P is

```
{    S    →   aABB | Aaa ,

     A    →   aBB | a ,

     B    →   bBB | aBB | a     }
```

QUESTION 5 (marks 25). The following Turing Machine is given -

```
TM = ({q0, q1, q2, q3}, {1}, {1, □}, δ, q0, □, {q3})

     δ(q0, 1) = (q0, x, R)
     δ(q0, □) = (q1, □, L)
     δ(q1, x) = (q2, 1, R)
     δ(q2, 1) = (q2, 1, R)
     δ(q2, □) = (q1, 1, L)
     δ(q1, 1) = (q1, 1, L)
     δ(q1, □) = (q3, □, R)
```

Analyze each transition and describe the functionality of TM. Write the instantaneous descriptions when this TM starts in q0 at the left most symbol of input string - `1111`, `i.e.`

q_0 `1111` $\vdash^*$?

(End of Examination Paper)