CA215 Languages and Computability
Spring 2007

Attempt **three** questions. All questions carry equal marks.

Q 1.
(i) Define the constraints on α and β in a finite-state rewrite rule.

(ii) Define a Deterministic Finite Automaton in terms of the quintuple $M = (Q, \Sigma, \delta, q_0, F)$.

(iii) For the language $a^n b^m$, give the transition function.

(iv) Draw the finite-state network which accepts the same language.

(v) Given the configuration $(0, aaabb)$, trace the execution of the DFA in (iii) and (iv).

(vi) Provide the formal grammar in terms of the quadruple $\langle V_t, V_n, P, S \rangle$ that describes the language $a^n b^m$.

Q 2.
(i) Define the constraints on α and β in a context-free rewrite rule.

(ii) Describe how context-free grammars can handle types of recursion that finite-state grammars (FSG) cannot.

(iii) Consider the following ruleset:

- $S \rightarrow NP, VP$
- $NP \rightarrow D, N$
- $VP \rightarrow V$
- $VP \rightarrow V, NP$
- $D \rightarrow the$
- $N \rightarrow man$
- $V \rightarrow eats$

(a) Write down the strings in the language permitted by this grammar.

(b) Show which of these rules could be rules in a FSG. Explain why.

(c) Given these rules, state the members of the sets V_t and V_n.

(iv) Consider the following ruleset:

- $E \rightarrow E, +, E$
- $E \rightarrow E, -, E$
- $E \rightarrow a$

(a) Show that this grammar is ambiguous by giving two distinct parse trees for the sentence $a + a - a$.

(v) Construct a Pushdown Automaton for the Language $a^n b^n$.

Q 3.
(i) Define the constraints on α and β in a context-sensitive rewrite rule.

(ii) Assuming the following ruleset:

- $S \rightarrow$ *the program V*
- $S \rightarrow$ *the programs V*
- *the program V* $\rightarrow$ *the program runs*
- *the program V* $\rightarrow$ *the dog runs*
- *the programs V* $\rightarrow$ *the programs runs*
- *the programs V* $\rightarrow$ *the dogs runs*

(a) what's the complete language that this grammar defines?

(b) which of these rules could be part of a CFG or an FSG? Explain why.

(iii) Describe a Linear Bounded Automaton in terms of the quintuple $M = (Q, \Sigma, \Gamma, q_0, \delta)$.

(iv) Given the formulation of Type 2 and Type 1 rules, what are the implications for being able to draw trees demonstrating clearly the derivation of sentences in these language classes.

Q 4.
(i) Define the constraints on α and β in a Type 0 grammar rewrite rule.

(ii) What are the implications for the learning of such a language from input alone?

(iii) Give a formal definition of a Turing Machine in terms of the quintuple $M = (Q, \Sigma, \Gamma, q_0, \delta)$.

(iv) Construct a Turing Machine which erases the input string.

(v) For the initial configuration $(q_0, \underline{1}11\#)$, show how the Turing machine you provided in (iv) copes with that input tape.

Q 5.
The following are expressions in the *untyped* λ-calculus. Reduce the expressions as far as possible. Remember that abstraction is right-associative and that application is left-associative.

(i)	$(\lambda x.x)z$
(ii)	$(\lambda x.y)z$
(iii)	$(\lambda x.xx)z$
(iv)	$((\lambda x.(\lambda y.xy))\ z)\ u$
(v)	$(\lambda x.\lambda y.xy)\ z\ u$
(vi)	$(\lambda x.x(xy))(\lambda z.zx)$
(vii)	$(\lambda xy.x)(\lambda x.x)$
(viii)	$(\lambda x.xx)(\lambda x.xx)$
(ix)	$(\lambda x.xx)((\lambda x.x)(\lambda x.x))$
(x)	$(\lambda x.xy)(\lambda z.zx)(\lambda z.zx)$
(xi)	$(\lambda x.xy)((\lambda z.zx)(\lambda z.zx))$
(xii)	$(\lambda x.xyxx)((\lambda z.z)w)$