COMP 181
Models of Languages and Computation
Spring 2001
Final Exam
Friday, May 4, 2001
Closed Book - Closed Notes
This exam has four pages.
Don't forget to write your name or ID and pledge on the exam sheet.

1. (Each part is worth 2 points) Fill in the blanks with one of the following:

a) finite
b) regular but not finite
c) deterministic context free but not regular
d) context free but not deterministic context free
e) recursive (that is, decidable) but not context free
f) recursively enumerable (that is, partially decidable) but not recursive
g) not recursively enumerable

Also recall that if $i$ is an integer, then $i$ can be interpreted as a binary number, that is, as a bit string, and this bit string can be read as a character string in ASCII or some other system, and then if this character string describes a Turing machine, then $T_i$ is the Turing machine described by this character string, else $T_i$ is some fixed Turing machine. The point is that $T_i$ is some Turing machine that can be computed from $i$, and every Turing machine can be represented as $T_i$ for some $i$.

1.1) The language $\{x \in \{0,1\}^* : x \text{ has length greater than } 5\}$ is b

1.2) The language $\{x \in \{0,1\}^* : x \text{ contains more ones than zeroes}\}$ is c

1.3) The language $\{0^n c 1^n c 0^n : n \geq 0\}$ is e

1.4) The language $\{x \in \{0,1\}^* : x \text{ contains the substring } 101\}$ is b

1.5) The language $\{10, 100, 1000, 10000\}$ is a

1.6) The language $\{w \in \{a,b,c\}^* : w \text{ does not have the same number of a's, b's, and c's}\}$ is d

1.7) The language $\{0^n c 1^n : n \geq 0\}$ is c

1.8) The language $\{(i,j) : \text{ Turing machine } T_i \text{ halts on input } j\}$ is f

1.9) The language $\{(i,j) : \text{ Turing machine } T_i \text{ does not halt on input } j\}$ is g

(0U1)*101(0U1)*

2. (Each part is worth 2 points) Fill in the blanks with one of the following:

a) finite
b) regular but not necessarily finite
c) deterministic context free but not necessarily regular
d) context free but not necessarily deterministic context free
e) recursive (that is, decidable) but not necessarily context free
f) recursively enumerable (that is, partially decidable) but not necessarily recursive
g) not recursively enumerable

2.1) If $L$ is a language represented by a regular expression then $L$ is ___b___
2.2) If $L$ is the language accepted by a nondeterministic finite automaton then $L$ is ___b___
2.3) If $L$ is the language accepted by a deterministic push-down automaton then $L$ is ___c___
2.4) If $L$ is the language accepted by an arbitrary push-down automaton then $L$ is ___d___
2.5) If $L$ is the intersection of a regular language and a context-free language then $L$ is ___d___
2.6) If $L$ is the language generated by a context-free grammar then $L$ is ___d___
2.7) If $L$ is the intersection of a finite language and a regular language then $L$ is ___a___
2.8) If $L$ is $L_1 \cup L_2$ where $L_1$ is regular and $L_2$ is context-free then $L$ is ___d___
2.9) If there is a Turing machine $T$ with two halting states $y$ and $n$ and for strings (words) in $L$, $T$ halts in state $y$ and for strings (words) not in $L$, $T$ halts in state $n$ then $L$ is ___e___
2.10) If there is a Turing machine $T$ which halts for strings in $L$ and does not halt for strings not in $L$ then $L$ is ___f___

3. (4 points) Consider the following extensions of a Turing machine. Which of them permit the Turing machine to decide problems that could not be decided before? Circle all correct answers.

a) Nondeterminism.
b) Random access.
c) Two-way infinite tape.
d) More read-write heads.

none

4. (4 points) For each of the following problems, say whether it is decidable (recursive) or partially decidable (recursively enumerable). Thus there are two answers to give for each part:

a) Given a Turing machine $T$, to determine whether $T$ will halt when started on a blank tape. N, Y
b) Given a Turing machine $T$ and a string $w$, to determine whether $T$ will halt when started on the input $w$. N Y
c) To determine whether China will give back the US spy plane. Y Y
d) Given a context-free grammar $G$ in Chomsky normal form and a string $w$, to determine whether $w \in L(G)$. Y Y

For the following question, choose the best answer and circle or underline it.

5. (4 points) Suppose $L$ is a language and the relation $\approx_L$ has $n$ equivalence classes. Let $M$ be an arbitrary deterministic finite automaton such that $L(M) = L$. Then

a) The number of states of $M$ is at least $n^2$.
b) The number of states of $M$ is at least $2^n$.
c) The number of states of $M$ is at least $n$.
d) The number of states of $M$ is at most $2^n$.

6. (10 points) Consider the following two proofs, one of which we call the Donald Duck proof and one of which we call the Mickey Mouse proof:

Donald Duck proof: Consider the language $L = \{ca^m ca^m ca^m : m \geq 0\}$. Suppose the opponent chooses $n$. I choose the string $ca^n ca^n ca^n$. The opponent chooses to express this string as the concatenation $uvxyz$ of the strings $u$, $v$, $x$, $y$, and $z$ with not both $v$ and $y$ equal to $e$, the empty string. I choose $i = 2$. Now, if $v$ and $y$ contain only $a$'s, then the string $uv^2xy^2z$ will have unequal numbers of $a$'s in its three groups of $a$'s. If $v$ or $y$ contain a $c$, then $uv^2xy^2z$ will have too many $c$'s. In all cases, $uv^2xy^2z$ is not in $L$.

Therefore the language $L$ is not context free.

Mickey Mouse proof: Consider the language $L = \{ca^m ca^m ca^m : m \geq 0\}$. Suppose the opponent chooses $n = 5$. I choose the string $ca^5ca^5ca^5$. The opponent chooses to express this string as the concatenation $uvxyz$ of the strings $u$, $v$, $x$, $y$, and $z$ with $u = caa$, $v = aaaca$, $x = aaaaca$, $y = aa$, and $z = aa$. I choose $i = 0$. The string $uv^0xy^0z$ is $caaaaaacaaa$ which is not in $L$ because it has only two $c$'s and because the number of $a$'s is not right. Therefore the language $L$ is not context free.

If one of these proofs is better than the other, say which is better and justify your answer. If both proofs are good or both proofs are bad, say why.

DD is better

because it gives a general strategy to win.

7. (8 points) What does the deterministic Turing machine $M = (K, \Sigma, \delta, s, H)$ do, where $K = \{s, t, h\}$, $\Sigma$ includes $\{a, b, c, \sqcup\}$ and possibly other symbols, $H = \{h\}$, and $\delta$ includes the following rules, along with possibly other rules:

$\delta(s, \sqcup) = (t, a)$
$\delta(s, a) = (h, a)$
$\delta(s, b) = (h, b)$
$\delta(t, \sqcup) = (h, \sqcup)$
$\delta(t, a) = (t, \rightarrow)$
$\delta(t, b) = (t, b)$

Here $\sqcup$ represents a blank. State in words what happens when $M$ is started with the read write head scanning a blank and a sequence of $a$'s and $b$'s to the right of the scanned blank. What will $M$ write on the tape? Under what conditions will $M$ halt?

8. (10 points) Consider the context-free grammar $(\{S, A, a, b, c\}, \{a, b, c\}, R, S)$ where the rules $R$ are as follows:

$$S \to A$$
$$A \to AA \qquad A \to a$$
$$A \to b \qquad A \to c$$

Is this grammar ambiguous? Justify your answer.

yes

2 parse trees

for aaa for example

9. (12 points) Let $L$ be $\{i : T_i$ does not halt on input $i\}$ where $T_i$ is as defined in question 1. Show that there is no Turing machine $T$ such that $T$ halts on input $i$ if $i$ is in $L$, and $T$ does not halt on input $i$ otherwise. (Hint: Since $T$ is a Turing machine, $T$ is $T_j$ for some integer $j$.)

Give T the string j as input and derive a contradiction.

(More details are needed for your answer.)