CS4123 THEORY OF COMPUTATION. B97 **EXAM 2**
Prof. Carolina Ruiz **Dec. 18, 1997**
Department of Computer Science
Worcester Polytechnic Institute

Name: C. Ruiz

Instructions

- Answer all 5 problems. Each problem is worth 20 points.
- You can use (without having to prove them again in this exam) any results proven in class, quizzes, homeworks, handouts, or in the book
- **Show your work and justify your answers**
- Use the space provided to write your answers
- Write your name on each page
- ASK if you have any questions

Problem 1. Turing Machines (20 points)

Let $\Sigma = \{0, 1\}$. Construct the state–transition diagram of a deterministic Turing machine that decides the following language:

$L = \{w | w \in \Sigma^\star$ and w contains at least two 0s and contains no two or more consecutive 0s$\}$

Make ALL transitions explicit in the diagram.

Solution:

Name: C. Ruiz

Problem 2. Recursive Functions (20 points)

Consider the function:

$$\text{EVEN}(n) := \begin{cases} 1, & \text{If } n \text{ is an even number, i.e. } n = 0, 2, 4, 6, 8, \ldots \\ 0, & \text{Otherwise} \end{cases}$$

Show that the function EVEN is primitive recursive.
Any functions used in your solution and that were not shown to be primitive recursive in class or one of the handouts must be shown to be primitive recursive as part of your solution.

Hints: Define EVEN using the following template:

$\text{EVEN}(0) =$

$\text{EVEN}(k+1) =$

Solution:

$\text{EVEN}(0) = 1$

$\text{EVEN}(k+1) = \text{MINUS}(1, \text{EVEN}(k)) \; = \text{“}1 - \text{EVEN}(k)\text{”}.$

Name: C. Ruiz

Problem 3. Decidability (20 points)

Let $\Sigma = \{a, b, c\}$. Show that the following language is decidable:

$$L = \{a^n b^n c^n | n \geq 0\}$$

where a^n denotes a string containing n consecutive occurrences of the letter "a".

Solution: Let's construct a TM M that decides L:

M = "on input string w:

1. If w is the empty word, then accept w.
2. Otherwise, scan the input to make sure that w consists of a sequence of one or more "a"s followed by a sequence of one or more "b"s followed by a sequence of one or more "c"s.
3. REPEAT

   Locate the first "a" on the tape and cross it out.
   Locate the first "b" on the tape and cross it out. If no "b" is found, reject w.
   Locate the first "c" on the tape and cross it out. If no "c" is found, reject w.

   UNTIL no more "a"s are found on the tape.
4. Scan the tape to check if any "b"s or "c"s remain on the tape.
   If so, reject w.
   Otherwise, accept w."

This machine M stops on all inputs; it accepts an input w iff w is of the form $a^n b^n c^n$ for some $n \geq 0$; and rejects input w otherwise.

Name: C. Ruiz

Problem 4. Undecidability (20 points)

Let Σ be an alphabet. Show that

$$\text{FINITE}_{TM} = \{\langle M\rangle | M \text{ is a Turing machine and } M \text{ accepts only finitely MANY words}\}$$

is undecidable.

Hint: We already showed that $A_{TM} \leq_m \text{INFINITE}_{TM}$, where

- $\text{INFINITE}_{TM} = \{\langle M\rangle | M \text{ is a Turing machine and } M \text{ accepts infinitely many words}\}$
- $A_{TM} = \{< M, w > | M \text{ is a TM, } w \text{ is a word, and } M \text{ accepts } w\}$.

Solution:

Since A_{TM} is undecidable and we proved already that $A_{TM} \leq_m \text{INFINITE}_{TM}$, then we know that $\text{INFINITE}_{TM}$is undecidable. Note that $\text{FINITE}_{TM}$is the complement of INFINITE_{TM}. Hence, FINITE_{TM}has to be undecidable, because otherwise INFINITE_{TM}would be decidable.

(**Note:** Formally speaking INFINITE_{TM} is not equal to $\Sigma^\star - \text{FINITE}_{TM}$. Nevertheless you can express INFINITE_{TM}in terms of FINITE_{TM}as follows:

$$\text{INFINITE}_{TM} = \{\langle M\rangle | M \text{ is a TM}\} \cap [\Sigma^\star - \text{FINITE}_{TM}]$$

Since $\{\langle M\rangle | M \text{ is a TM}\}$ is decidable and decidable languages are closed under intersection and complementation, if FINITE_{TM}were decidable so would INFINITE_{TM}.)

Name: C. Ruiz

Problem 5. N, NP, NP–completeness (20 points + 10 extra points)

Let $G = (V, E)$ be an undirected graph. We say that G contains a k–independent set if there is a subset I of V containing k nodes such that whenever a and b belong to I then there is NO edge between a and b in E.
Consider the language:

INDEPENDENT = $\{\langle G, k\rangle | G$ is an undirected graph and G contains a k–independent set$\}$.

5.1. (20 points) Show that INDEPENDENT is in NP.

Solution: The following nondeterministic TM constructs a k-independent set for G if one exists:

M_{ind} = "on input $\langle G, k\rangle$:

1. Nondetermistically construct a set I containing k nodes from V.
2. For every pair of nodes a and b in I:
   (a) Check that there is no edge (a, b) in E.
   (b) If there is such edge in E, reject $\langle G, k\rangle$.
3. If the checking on step 2 succeeded, then accept $\langle G, k\rangle$."

input: $\langle G, k\rangle$, where $G = (V, E)$
size of the input: $n = |V|$
number of steps for input of size n:

Instruction	number of steps the instruction takes	
1	$O(n)$	
2	$O(n^2)$	$k = O(n)$ and so there are at most n^2 pairs of nodes from I to check
3	$O(1)$	

So, M_{ind} runs in $O(n^2)$ and so polynomial time on the size of the input.

5.2. (10 extra points) Show that INDEPENDENT is NP–complete.

Hint: Show that CLIQUE $\leq_P$ INDEPENDENT, where

CLIQUE = $\{\langle G, k\rangle | G$ is an undirected graph and G contains a k–clique$\}$.

Solution: Consider the following reduction from CLIQUE to INDEPENDENT:

CLIQUE	$\leq_P$	INDEPENDENT
$\langle G, k\rangle$ belongs to CLIQUE	iff	$f(\langle G, k\rangle) = \langle \bar{G}, k\rangle$ belongs to INDEPENDENT

where $\bar{G} = (V, \bar{E})$ is constructed from the graph $G = (V, E)$ as follows: $\bar{G}$ contains the same nodes as G does; and there is an edge in $\bar{G}$ between nodes a and b if and only if there in no such edge in G. In other words, $\bar{E}$ is the complement of E. (OVER)

Note that $\bar{G}$ can be constructed from G in at most $O(n^2)$, where $n = |V|$, since there are at most n^2 edges to include in $\bar{G}$. Hence the function f above can be computed by a polynomial time TM.

Let's prove that G has a k-clique V' iff V' is a k-independent set in $\bar{G}$:

G has a k-clique V'
iff for each pair of nodes a and b in V', the edge (a, b) belongs to E
iff for each pair of nodes a and b in V', the edge (a, b) does not belong to $\bar{E}$
iff V' is a k-independent set for $\bar{G}$.