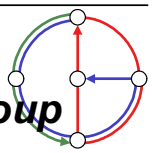
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing Group



AS 2007 Prof. Dr. R. Wattenhofer / Roland Flury / Stefan Schmid

Discrete Event Systems
Monster Exercise 5

1 Regular and Context-Free Languages

- Consider the following context-free grammar $G : S \rightarrow SS|1S2|0$. Describe the language $L(G)$ in words, and prove that $L(G)$ is not regular.
- The regular languages are a subset of the context-free languages. Give the context-free grammar for a language L that is regular.

2 Context-Free Grammars

Give context-free grammars for the following languages over the alphabet $\Sigma = \{0, 1\}$:

- $L = \{w|$ the length of w is odd$\}$
- $L = \{w|$ contains more 1s than 0s$\}$

3 Pushdown Automata

Consider the following context-free grammar G with non-terminals S and A, start symbol S, terminals '(', ')', and '0':

$$\begin{array}{lcl} S & \rightarrow & SA \,|\, \epsilon \\ A & \rightarrow & (S) \,|\, 0 \end{array}$$

a) What are the 4 shortest strings produced by G?

b) Context-free grammars can be ambiguous. Prove or disprove that G is unambiguous.

c) Design a push-down automaton M that accepts the language $L(G)$. If possible, make M deterministic.

4 Counter Automaton

A push-down automaton is basically a finite automaton augmented by a stack. Consider a finite automaton that (instead of a stack) has an additional *counter* C, i.e., a register than can hold a single integer of arbitrary size. Initially, $C = 0$. We call such an automaton a *Counter Automaton* M. M can only increment or decrement the counter, and test it for 0. Since theoretically, all possible data can be coded into one single integer, a counter automaton has unbounded memory. Further, let L_{count} be the set of languages recognized by counter automata.

- Let L_{reg} be the set of regular languages. Prove that $L_{reg} \in L_{count}$.
- Prove that the opposite is not true, that is, $L_{count} \notin L_{reg}$. Do so by giving a language which is in L_{count}, but not in L_{reg}. Characterize (with words) the kind of languages a counter automaton can recognize, but a finite automaton cannot.
- Which automaton is stronger? A counter automaton or a push-down automaton? Explain your decision.

5 Pumping Lemma revisited

a) Determine whether the language $L = \{1^{n^2} | n \geq 1\}$ is regular.

b) Consider a regular language L and a pumping number p such that every word $u \in L$ can be written as $u = xyz$ with $|xy| \leq p$ and $|y| \geq 1$, and that $xy^iz \in L \; \forall i \geq 0$.

What can you say about the minimum number of states needed for the corresponding DFA? What about the minimum number of states of the corresponding the NFA?

6 Push Down Automaton

For each of the following languages, draw a PDA (if possible deterministic) that accepts L.

a) $L = \{a^i b^j a^j b^i \mid i, j > 0\}$

b) $L = \{u \mid u \in \{0|1\}^*, \text{ and } u^{reverse} = u\}$

c) $L = \{u \mid u \in \{0|1\}^*, \text{ and } u^{reverse} \neq u\}$

7 Context Free Grammars

For each of the following languages give a CFG describing L.

a) $L = \{x\#y \mid x, y \in \{a, b\}^*, \text{ and } x \text{ is not a permutation of } y\}$

b) $L = \{x\#y \mid x, y \in \{a, b\}^*, \text{ and } x \neq y\}$

8 Tandem Pumping

For the following languages, determine whether they are context free or not.

a) $L = \{a^i b^j c^k \mid 0 < i < j < k\}$

b) $L = \{x \mid x \in \{0, 1\}^*, \text{ and } x \text{ contains an even number of '0' and an even number of '1'}\}$

c) $L = \{x\#y \mid x, y \in \{0, 1\}^*, \text{ and } x \text{ is a permutation of } y\}$

9 Transducer and Turing Machine

Alice is very happy because she was accepted for an internship at Tintel, one of the world's leading processor manufacturers. Unfortunately, she has only attended the famous DES lecture during her studies at ETH and knows nothing about electronic circuits. Therefore, she wants to solve her first assignment using a transducer - please assist her:

a) Alice first needs to compute the sum $a + b$ of two binary numbers a and b. The numbers are represented bitwise with LSB first. I.e. a is given as $a[0]a[1]a[2]a[3] \ldots a[n]$ and b as $b[0]b[1]b[2] \ldots b[n]$. Unfortunately, Alice receives a single string containing both, a and b mixed together in the form $a[0]b[0]a[1]b[1]a[2]b[2] \ldots a[n]b[n]$. Draw the transducer that outputs $a+b$ (in binary) with same bit ordering (LSB first) and only accepts upon a valid input (even number of binary digits).

b) Mister Intal (Alice's supervisor) is only familiar with assembler and the Turing machine. Therefore, Alice is now asked to translate her transducer to a Turing machine. The two numbers a and b are written on the machine tape in the form $a[n]a[n-1] \ldots a[1]a[0] + b[m]b[m-1] \ldots b[1]b[0]$, where $a[n]$ is the MSB of a. Note that the two numbers are separated by the '+' symbol, that they have an equal number of bits[1], i.e. $m = n$, and that $\Gamma = \{0|1| + |\square\}$. Initially, the head of the TM points to $a[n]$. At the end, it should point to the MSB of the result.

Hint You may extend the alphabet Γ to put temporary symbols on the tape.

[1] Your machine may crash or produce a wrong result if this condition does not hold.

c) In her last assignment, Alice is asked to implement a *binary to unary converter*. This converter takes a number a in binary (alphabet $\{0|1\}$) and converts it to a unary number u (alphabet $\{1\}$). Initially, the TM head points to the MSB of a. At the end, the head should point to the right-most digit of u.