

Teoría Matemática de la Computación

Segundo Problemario

Prof. Miguel A. Pizaña
30 de Noviembre de 2006

I Tareas

1. Averiguar cuál es el máximo nivel de anidación que permite su compilador favorito.
2. Leer “Computer Machinery and Intelligence”
3. ¿Qué es la objeción del salón chino (Chinese Room)?
4. Leer “On Computable Numbers, with an Application to the Entscheidungsproblem”
5. ¿Qué dice el artículo 39 de la Constitución Política de los Estados Unidos Mexicanos? ¿y el 85?

II Lenguajes Regulares

1. Probar que los lenguajes regulares son cerrados bajo: complemento, unión, intersección, concatenación y cerradura de Kleene.
2. Probar que toda expresión regular puede ser convertida en un AFD.
3. Probar que todo AFD puede ser convertido a una expresión regular.
4. Probar que todo AFN puede ser convertido a AFD y viceversa.
5. Mencione un problema de decisión que se sepa que requiere un tiempo de ejecución no-elemental para su solución. ¿Qué es un tiempo de ejecución no-elemental?
6. Demostrar que cualquier AFD que reconozca al lenguaje denotado por $(0 + 1)^*0(0 + 1)^n$ requiere un número exponencial de estados ($\sim 2^n$).

7. Realizar las transformaciones indicadas usando los algoritmos vistos en clase.
 - (a) Transformar los AFN's de la Figura 1 en AFD's
 - (b) Transformar los AFD's de la Figura 2 en expresiones regulares.
 - (c) Transformar las siguientes expresiones regulares en AFN's
 1. $(0 + 1)^*00(0 + 1)^*$
 2. $(0 + \varepsilon)(1 + 10)^*$
 3. $(0 + 1)^*0(0 + 1)^7$
8. Los Lenguajes de programación de alto nivel (= con estructuras anidadas) no son regulares. ¿Cómo es que existen los compiladores entonces?
9. "Sumar uno" es una operación que no se puede hacer con memoria acotada. ¿Cómo es que usamos las computadoras para sumar uno?
10. ¿Cuántos números reales hay? ¿Cuántos de ellos pueden ser representados en una computadora real (de memoria finita)? Si las computadoras pudieran tener memoria ilimitada (=la memoria es infinita, pero sólo podemos usar una cantidad finita de casillas en cada momento) ¿Cuántos números reales podríamos representar en la computadora? ¿Cómo es que trabajamos con las computadoras con los números de punto flotante como si todo esto no representara un problema?

III Máquinas de Turing

1. ¿Qué es una máquina de Turing? ¿Algoritmo? ¿Lenguaje recursivamente enumerable? ¿Lenguaje recursivo? Dada una máquina de Turing M, ¿Cómo se define el lenguaje aceptado por M? Menciona alguna cosa que puede hacer una Máquina de Turing, que no pueda hacer ninguna computadora real.
2. Para cada uno de los siguientes lenguajes proporcione un algoritmo (=máquina de Turing que siempre para) que los reconozca:
 - (a) $\{n : n \text{ es par} \}$
 - (b) $\{0^n 1^n : n \in \mathbb{N}\}$.
 - (c) $\{ \text{paréntesis balanceados} \}$.

- (d) { expresiones aritméticas con paréntesis }
 - (e) { palíndromos de 0's y 1's }.
 - (f) $L_m = \{0^{n+m}1^n : n \geq 0\} \cup \{0^n1^{n+m} : n \geq 0\}$.
 - (g) $\{w \in \{0, 1\}^* : w \text{ tiene igual número de 0's que de 1's}\}$.
3. En cada inciso proporcione una Máquina de Turing que:
 - (a) Inserte un símbolo al inicio de la cinta.
 - (b) Dado un número binario no negativo n , calcule $n + 1$.
 - (c) Dado un número binario positivo n , calcule $n - 1$.
 - (d) Dado un par de número binarios separados por un “#”, calcule la suma.
 4. Demuestre que todo lo que un AFD puede hacer, también lo puede hacer alguna máquina de Turing.
 5. Demuestre que todo lenguaje regular es recursivo.
 6. Demuestre que todo lenguaje recursivo es recursivamente enumerable.
 7. ¿Qué es una máquina de Turing universal?
 8. Demuestre que existen lenguajes que son:
 - (a) No recursivamente enumerables.
 - (b) Recursivamente enumerables pero no recursivos.
 - (c) Recursivos pero no regulares.
 9. Enuncie la tesis de Church-Turing. La tesis de Church-Turing ¿puede ser demostrada matemáticamente? Explique.
 10. ¿Cómo le llama Turing a sus máquinas? ¿Qué diferencias hay entre su definición y la definición vista en clase? ¿Qué es para Turing un número computable? ¿son computables π , $\sqrt{2}$, $\frac{1}{3}$, $\tan(23)$? ¿Cuál es la razón por la que la máquina usa una cinta para hacer anotaciones? ¿Por qué usa un número finito de símbolos? ¿Qué es el Entscheidungsproblem? ¿Qué prueba Turing al respecto?
 11. ¿Qué es la prueba de Turing (Turing's Test)? ¿Usted cree que alguna máquina podrá algún día aprobar la prueba de Turing? Suponiendo que sí, ¿Eso indicaría que la máquina tiene inteligencia? ¿Puede el cerebro humano hacer algo que las Máquinas de

Turing no? Según Turing, ¿pueden las máquinas pensar? ¿Qué argumentos en favor y en contra da Turing? Según usted, ¿pueden las máquinas pensar? ¿por qué?

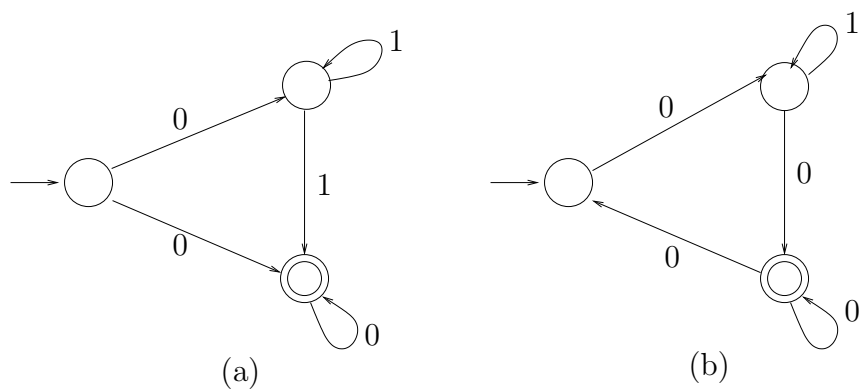


Figura 1: Convertir de AFN a AFD

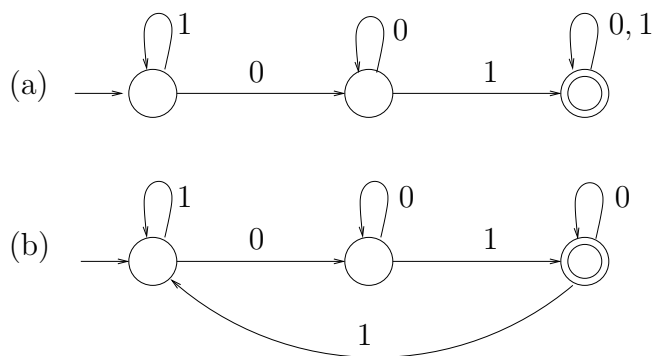


Figura 2: Convertir de AFD a expresión regular