

UNIVERSIDAD CARLOS III DE MADRID
Ingeniería Informática, 3º Curso
Examen de Informática Teórica II
Febrero de 2006

Duración: 3 horas

Pregunta 1 (3 puntos)

Diseña una máquina de Turing que acepte el siguiente lenguaje:

$$L = \{a^n b^{n-1} c^{n+3} \mid n \geq 1\}$$

- El contenido inicial de la cinta de entrada será una palabra con símbolos a, b y c. Por ejemplo:
%%%%%%%%aabbccccc%%%%%%%%
- La cinta de salida contendrá únicamente blancos.
- Nuestra máquina de Turing está definida de la siguiente forma: $MT = (\Gamma, \Sigma, b, Q, q_0, f, F)$ donde:

$$\begin{aligned} \Gamma &= \{a, b, c, A, B, C, ' '\} & \Sigma &= \{a, b, c\}, \\ b &= ' ' & Q &= \{q_1, q_2, q_3, q_4, q_5, q_6, q_7, qf, qe, qbe, qbf, \text{exito}, \text{fracaso}\} \\ q_0 &= q_1 & f &\rightarrow \text{función de transición.} & F &= \{\text{éxito}, \text{fracaso}\} \end{aligned}$$

Completa la función de transición con ayuda de las siguientes trazas de ejecución.

<pre><q1>aaabbccccc A<q2>aabbccccc Aa<q2>abbccccc Aaa<q2>bbccccc Aaabbcccc<q2>cc Aaabbccccc<q2>c Aaabbccccc<q2> Aaabbccccc<q3>c Aaabbccccc<q4>c Aaabbccccc<q5>c Aaabbcc<q6>c Aaabbcc<q7>cC Aaabb<q7>ccC Aaab<q7>bccC Aaa<q6>bBccC Aa<q6>abBccC A<q6>aabBccC <q6>AaabBccC A<q1>aabBccC AA<q2>abBccC AAabBcc<q2>C AAabBc<q6>cC AAabB<q7>cCC AAab<q7>BcCC AAa<q7>bBcCC AA<q6>aBBcCC A<q6>AaBBcCC AA<q1>aBBcCC AAA<q2>BBcCC AAABBc<q2>CC AAABB<q6>cCC AAAB<q7>BCCC AAA<q7>BBCCC AA<q7>ABBCCC AAA<qe>BBCCC AAABBCC<qe>C AAABBCCC<qe> AAABBCC<qbe>C AAABBC<qbe>C A<qbe>A <qbe>A <qbe> <exito></pre>	<pre><q1>abcccc A<q2>bcccc Ab<q2>cccc Abc<q2>ccc Abcc<q2>cc Abccc<q2>c Abcccc<q2> Abccc<q3>c Abcc<q4>c Abc<q5>c Ab<q6>c A<q7>bC <q6>ABC A<q1>BC AB<qf>C ABC<qf> AB<qbf>C A<qbf>B <qbf>A <qbf> <fracaso></pre>	<pre><q1>aaabbccccc A<q2>aabbccccc Aaabbcc<q7>cC Aaabb<q7>ccC Aaabb<q7>bccC Aaab<q6>bBccC Aaa<q6>bbBccC Aa<q6>abbBccC A<q6>aabbBccC <q6>AaabbBccC A<q1>aabbBccC AA<q2>abbBccC AAabbBc<q6>cC AAabbB<q7>cCC AAabb<q7>BcCC AAab<q7>bBcCC AAa<q6>bBBcCC AA<q6>abBBcCC AA<q1>abBBcCC AAA<q2>bBBcCC AAAb<q2>BBcCC AAAbBB<q6>cCC AAAbB<q7>BCCC AAAb<q7>BBCCC AAA<q7>bBBCCC AA<q6>ABBCCC AAA<q1>BBCCC AAAB<qf>BBCCC AAABBBCC<qf>C AAABBBCCC<qf> AAABBBCC<qbf>C <qbf>A <qbf> <fracaso></pre>
---	---	---

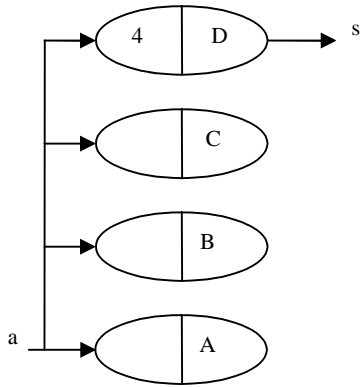
	a	b	c	A	B	C	
→q1					qf,'B', D		
q2		q2,'b',D				q6,'C',I	q3,' ',I
q3							
q4							
q5		qf,'b',I	q6,' ',I				
q6			q7,'C',I	q1,'A',D	qf,'B',I		
q7	qf,'a',I	q6,'B',I	q7,'c',I		q7,'B',I		
qe			qf,'c',I	qe,'A',D	qe,'B',D		qbe,' ',I
qf		qf,'b',D		qf,'A',D		qf,'C',D	
qbe				qbe,' ',I	qbe,' ',I	qbe,' ',I	exito,' ',D
qbf		qbf,' ',I			qbf,' ',I		
*Fracaso							
*Exito							

Solución:

	a	b	c	A	B	C	
→q1	q2,'A',D	qf,'b',D			qf,'B',D		
q2	q2,'a',D	q2,'b',D	q2,'c',D		q2,'B',D	q6,'C',I	q3,' ',I
q3			q4,' ',I				
q4		qf,'b',I	q5,' ',I				
q5		qf,'b',I	q6,' ',I				
q6	q6,'a',I	q6,'b',I	q7,'C',I	q1,'A',D	qf,'B',I		
q7	qf,'a',L	q6,'B',I	q7,'c',I	qe,'A',D	q7,'B',I		
qe			qf,'c',I	qe,'A',D	qe,'B',D	qe,'C',D	qbe,' ',I
qf	qf,'a',D	qf,'b',D	qf,'c',D	qf,'A',D	qf,'B',D	qf,'C',D	qbf,' ',I
qbe				qbe,' ',I	qbe,' ',I	qbe,' ',I	exito,' ',D
qbf	qbf,' ',I	qbf,' ',I	qbf,' ',I	qbf,' ',I	qbf,' ',I	qbf,' ',I	fracaso,' ',D
*Fracaso							
*Exito							

Pregunta 2 (1.5 punto)

Dado el siguiente circuito de Mc-Culloch Pitts:



Se pide:

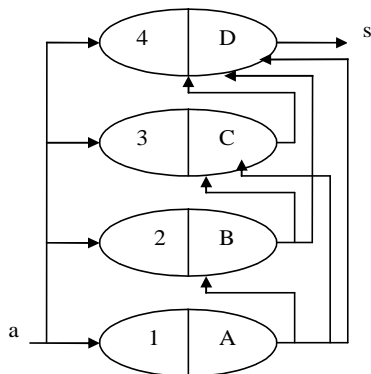
a) Determinar los umbrales y las conexiones entre células para que se cumpla:

a(t)	0	1	0	0	0	1	1	1	1
s(t+1)	0	0	0	0	0	0	0	0	1

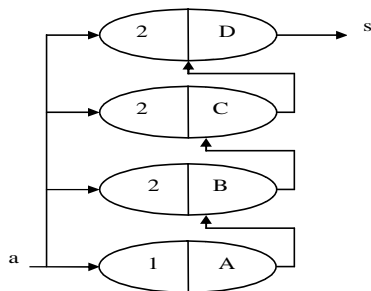
b) Construya un circuito diferente al anterior que realice la misma operación.

Solución:

a)



b)



Pregunta 3 (3.5 puntos)

Las moléculas de ADN consisten en una secuencia construida a partir de cuatro elementos básicos, llamados nucleótidos o bases (Adenina, Guanina, Timina y Citosina). Una serie de complejos mecanismos bioquímicos realizan una traducción a partir del ADN para obtener como resultado las proteínas que intervienen en el metabolismo celular. Una cuestión importante a la hora de la creación de proteínas es el *tipo de transición* que contiene la secuencia de ADN. Estas transiciones pueden ser del tipo denominado Intrón/Exón o Exón/Intrón. Podría ocurrir también que la secuencia de ADN no contuviera ninguna transición. Las enzimas encargadas de realizar este proceso detectan el tipo de transición de forma automática, pero se desconoce el criterio en base al cual se puede determinar en qué secuencias de nucleótidos existen transiciones del tipo Intrón/Exón o Exón/Intrón.

Se pretende construir un perceptron multicapa para intentar identificar de forma fiable el tipo de transición que contienen las secuencias de ADN. Para ello se dispone de un conjunto de 3100 fragmentos de ADN, para los cuales es conocido si contienen transiciones Intrón/Exón, Exón/Intrón o ninguna de ellas. Cada uno de los fragmentos contiene una secuencia de 60 bases. Un ejemplo extraído de dicho conjunto es:

ACTCATCGAACTCTGCTGATAGCCAATGAGGTAATTTCTTTATGATTCCCTACAGTCTGT, Exon/Intron
TGACCTGATCTTTGCTCTCCCCCTGGCCAGTTGAGGAGGAGAACCCGGACTTCTGGAACC, Intron/Exon
AGAACCACCTACTTCTTGGGGAGGTAGGTCTGCTTCCCTTCAACTCAGGATACAACCTGCT, Ninguna
GTGGGCTGAAGCCTGGCTCTGTCCCTGCAGGGTGCCTGGTATGTGTGGAACCGCACTGAG, Intron/Exon
AAGCACACCACGGATCTAGATGCCAGTAAAGTGAGTTCAAATATCCCCTTCTGATTTGC, Exon/Intron
AGGAAAAAGAGGAAGGGAGGGAGGGAGGAAGGAAGGAAAGGAAGGAAGGGAGAGAGA, Ninguna
siendo A=Adenina, G=Guanina, T=Timina y C=Citosina.

Se pide:

- Formular el problema desde el punto de vista de redes de neuronas.
- Indicar el número de neuronas de entrada y salida para el perceptron multicapa.
- Explicar el mecanismo para la obtención de los patrones de entrenamiento y validación
- Explicar detalladamente los pasos para realizar el aprendizaje del perceptron multicapa
- Criterio para determinar a partir de la salida de la red, de qué tipo de transición se trata.
- Suponiendo que se dispone de cierta información que dice que el tipo de transición, que en el caso de existir, está localizada entre las bases de las posiciones 30 y 31, ¿realizaría algún cambio en su planteamiento anterior?. Razone su respuesta.

Solución

a) La información de entrada a la red sería la secuencia de ADN y la información de salida el tipo de transición. Es necesario codificar la información de entrada y salida con valores numéricos. Una posible codificación, aunque no es la única (habría que validarla experimentalmente) sería

A=0.05, G=0.4, T=0.7, C=0.99.

Para la salida: transición tipo Exon/Intron: salida deseada 0; transición tipo Intron/Exon: salida deseada 1; ninguna transición: salida deseada 0.5

b) Según el planteamiento anterior, la red tendría 60 neuronas de entrada y 1 neurona de salida

c) Del conjunto total disponible se extraen aleatoriamente el 20% (por ejemplo) que será utilizado como conjunto de test. Habría que comprobar que todas las clases Exón/Intrón, Intrón/Exón y Ninguna tienen representación tanto en el conjunto de entrenamiento como en el de validación.

d) Una vez definida la arquitectura de la red (neuronas de entrada, ocultas y salida), los pasos son:

- Inicialización aleatoria de los pesos y umbrales
- Se toma un patrón del conjunto de entrenamiento, entrada y su salida deseada y se calcula la salida de la red para dicha entrada.
- Se evalúa el error cometido por la red como la diferencia al cuadrado entre la salida de la red y la salida deseada
- Se modifican todos los parámetros de la red utilizando el algoritmo de retropropagación
- Se repiten los pasos 2, 3, y 4 para todos los patrones de entrenamiento, completando así un ciclo de aprendizaje.
- Se realizan n (por ejemplo 10) ciclos de aprendizaje y se evalúa el error que comete la red para los patrones de validación o test. Para ello se presenta a la red todos los patrones de validación y se mide el error que comete en la salida
- El proceso de aprendizaje continúa hasta que se cumple algunos de los siguientes criterios:
 - Se estabiliza el error de entrenamiento y validación
 - El error de entrenamiento sigue bajando o está estable pero el error de validación empieza a aumentar.

e) Si se considera una única salida, es necesario definir intervalos APRA cada una de las clases. Así por ejemplo, si Y es la salida de la red

Si $0 \leq Y < 0.3$ el patrón contiene una transición del tipo Exon /Intrón

Si $0.3 < Y < 0.7$ el patrón no contiene transición

Si $0.7 \leq Y < 1$ el patrón contiene una transición del tipo Intrón/Exón

f) En principio se podría pensar en presentar la red sólo dos variables de entrada, las correspondientes a las posiciones 30 y 31. utilizando los ejemplos anteriores, se observa que si se extraen las bases en las posiciones 30 y 31, se obtienen los siguientes patrones GG, Exon/Intron

GT, Intron/Exon
 CT, Ninguna
 GG, Intron/Exon
 AG, Exon/Intron
 AG, Ninguna

Se observa entonces que para patrones iguales en la entrada, la salida es diferente. Por tanto este planteamiento no sería correcto.

Sí se podría pensar en reducir el número de entradas a la red, presentando una ventana de bases entorno a las posiciones 30 y 31. Por ejemplo se podría considerar una ventana de anchura x , que contenga las posiciones alrededor de las posiciones 30 y 31. habría que hacer diferentes simulaciones variando la anchura de la ventana y analizar la capacidad de aprendizaje de al red.

Para anchura $x=3$, se presentan como entrada a la red las posiciones: 28 29 30 31 32 33

Para anchura $x=5$, se presentan como entrada a la red las posiciones: 26 27 28 29 30 31 32 33 34 35

Para anchura $x=7$, se presentan como entrada a la red las posiciones: 24 25 26 27 28 29 30 31 32 33 34 35 36 37

Pregunta 4 (2 puntos)

Sea un Automata Celular unidimensional con dos estados, $Q=\{0,1\}$; índice de localidad 1, es decir $r=1$; y la vecindad viene dada por la célula colocada a la derecha, es decir $V=\{+1\}$. Se pide:

1. ¿Cuántas reglas o funciones de transición diferentes se pueden definir bajo estas condiciones?
2. Explique un mecanismo para enumerar dichas reglas.
3. Dado el autómata celular unidimensional con estados $\{0,1\}$, $r=1$ y $V=\{+1\}$ y función de transición dada por:

$$x_i(t) = x_i(t-1)$$

Explique razonadamente su comportamiento a lo largo del tiempo para cualquier configuración inicial

Solución

1. La función de transición depende de dos argumentos, por tanto el número de reglas es $2^4 = 16$
2. Cada regla se especifica por su secuencia binaria y por tanto por su número decimal equivalente, siguiendo el siguiente esquema

Cada función de transición es una asignación de 0 o 1 a los posibles argumentos que son: 11, 10, 01, 00. La asignación binaria a dichos argumentos se convierte a decimal. Así, podríamos hablar de:

Argumentos	11	10	01	00
Regla 1	0	0	0	1
Regla 2	0	0	1	0
Regla 3	0	0	1	1
Regla 4	0	1	0	0
.....				

3. La función de transición $x_i(t) = x_i(t-1)$ se corresponde con la regla 12

Argumentos	11	10	01	00
Regla 12	1	1	0	0

Dicha regla mantiene el estado de la célula de una iteración a otra, independientemente de su valor de vecindad. Por tanto, la configuración inicial de partida se mantiene invariable a lo largo del tiempo.