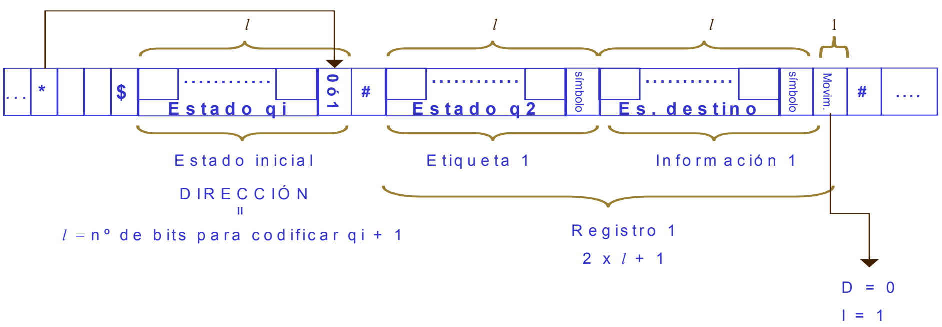
UNIVERSIDAD CARLOS III DE MADRID
Ingeniería Informática, 3º Curso
Examen de Informática Teórica II
Febrero de 2005

Duración: 3.5 horas

Pregunta 1 (3 puntos)

a) **(1 punto)** Definición y descripción de la cinta de una máquina de Turing universal (MTU).
La cinta de la MTU contiene en la parte izquierda del carácter $ la cinta de una MT binaria y con cinta limitada (MTBCL). La posición de la cabeza lectora de dicha cinta se simboliza en la cinta de la MTU mediante * y el bit en esa posición pasa a ser el último bit de la dirección.
La parte derecha del carácter $ contiene la dirección y número finito de registros. En la dirección aparece codificado en binario el estado inicial de la MTBCL y el bit que señala la cabeza lectora de la MTBCL. Cada transición de la MTBCL se codifica en un registro, salvo la transición de parada. Cada registro está formado por la etiqueta y la información. La etiqueta contiene codificado la parte izquierda de cada transición, es decir el estado actual y el bit a leer. La información contiene la parte derecha de la transición, es decir, el estado al que transita, el bit a escrbir y el movimiento (D es o e I es 1).
Se debe verificar que L=Longitud(dirección)=Longitud(etiqueta)=Longitud(información) +1.
El número de bits L depende del número de estados de la MTBCL, L= $E(\log_2 |Q|) + 1$
En la gráfica se presenta el esquema de la cinta:



b) **(1 punto)** Dada la siguiente máquina de Turing con alfabeto de cinta {a,b,c, ϕ}, alfabeto de entrada {a,b} y tabla de transición:

	a	b	c	ϕ
p	pcD	paD		qϕI
q	qaI		qbI	rϕP
r				

Se pretende construir la máquina de Turing binaria equivalente, donde a = 00, b = 01, c= 10, ϕ = 11. Complete la tabla de transición que se muestra a continuación, para que la máquina resultante sea equivalente a la Máquina de Turing anterior.

	0	1
→p		qI,1,I
p0	pa,0,I	
pa	pa0,1,D	vacío
pa0	p,0,D	vacío
pb		vacío
pb0	vacío	

q		q1,1,D
q0		vacío
qa		vacío
qa0		vacío
q1	qc,0,I	r,1,I
qc	vacío	qc1,0,D
qc1	qx,1,I	vacío
ql	q,0,I	q,1,I
qx		qy,1,I
qy		

r	vacío	r,1,P

	0	**1**
→p	p0, 0, D	qI,1,I
p0	pa,0,I	pb, 1, I
pa	pa0,1,D	vacio
pa0	p,0,D	vacio
pb	pb0, 0, D	vacio
pb0	vacio	p, 0, D
q	q0, 0, D	q1,1,D
q0	qa, 0, I	vacio
qa	qa0, 0, D	vacio
qa0	qx, 0, I	vacio
q1	qc,0,I	r,1,I
qc	vacio	qc1,0,D
qc1	qx,1,I	vacio
qI	q,0,I	q,1,I
qx	qy, 0, I	qy,1,I
qy	q, 0, I	q, 1, I
r	vacio	r,1,P

c) **(1 punto)** A partir de la máquina de Turing binaria anterior construir la cinta de entrada de la máquina de Turing universal para los cinco primeros estados, p, p0, pa, pa0, pb, y CMTB2 = ...101P101.

c) L(etiqueta) = E ($\log_2$ |Q|) + 1 — 17 estados -> $\log_2 17 = \log 17 / \log 2 = 4{,}1$

L = 5 + 1 = 6

m(registro) = 2 x L + 1 — m = 2 x 6 +1 = 13

m = 13 bits la longitud de cada registro.

D=0
I=1

p = 00000 p0 = 00001 pa = 00010 pa0 = 00011 pb = 00100 pb0 = 00101
q = 00110 q0 = 00111 qa = 01000 qa0 = 01001 q1 = 01010 qc = 01011
qc1 = 01100 qI = 01101 qx = 01110 qy = 01111 r = 10000

Las transiciones quedarían de la forma:

f(p,0) = (p0,0,D) -> 0000000000100
f(p,1) = (qI,1,I) -> 0000010110111
f(p0,0) = (pa,0,I) -> 0000100001001

f(p0,1) = (pb,1,I) -> 0000110010011
f(pa,0) = (pa0,1,D) -> 0001000001110
f(pa0,0) = (p,0,D) -> 0001100000000
f(pa,0) = (pa0,1,D) -> 0001000001110
f(pb,0) = (pb0,0,D) -> 0010000010100

La cinta de la máquina de Turing Universal sería:

...101*01$001#000000000100#0000010110111#0000100001001#0000110010011#00010000 01110#0001100000000#0001000001110#0010000010100..........

Pregunta 2 (1 punto)

Los circuitos de células de McCulloch-Pitts se caracterizan porque cada célula recibe un conjunto de entradas y emiten una salida que depende de dichas entradas y de la función de activación. Esta característica es compartida con las redes de neuronas artificiales. Explique la diferencia fundamental entre ambos modelos.

La diferencia fundamental es que para las redes de neuronas existe un procedimiento automático (ley de aprendizaje) para la determinación de todos los parámetros (pesos y umbrales). Dicho procedimiento no existe para los circuitos de células. Por otra parte, las redes de neuronas poseen un mayor número de parámetros libres que son los pesos, lo cual permite que las redes de neuronas puedan aproximar funciones más complejas.

Pregunta 3 (2,5 puntos)

Sea el siguiente AFD=({0,1},{q_0,q_1,q_2},q_0,q_2,f)

f	0	1
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_2	q_2

Supóngase que cuando el autómata recibe el símbolo 0, no se recibe de manera segura, sino que existe cierta probabilidad P de que se esté recibiendo dicho símbolo.

Se pide:

a) **(0,75 puntos)** Diseñar el autómata probabilístico que contemple esta situación.

$$\text{AFD: } M(0)=\begin{pmatrix}0&1&0\\0&1&0\\0&0&1\end{pmatrix}\ M(1)=\begin{pmatrix}1&0&0\\0&0&1\\0&0&1\end{pmatrix}$$

$$\text{AP: } M(0)=\begin{pmatrix}1-P&P&0\\0&P&1-P\\0&0&1\end{pmatrix}\ M(1)=\begin{pmatrix}1&0&0\\0&0&1\\0&0&1\end{pmatrix}$$

b) **(0,5 puntos)**¿Existe algún valor de P para que la palabra x=00 fuera una palabra aceptada por el AP con probabilidad igual a 1?. Razone su respuesta.

(1,0,0)xM(0)xM(0)=(?,?,P(1-P))
P(1-P)=1, Esta ecuación no tiene raices reales, por lo tanto x=00 nunca será aceptada

c) **(0,5 puntos)**¿Con qué umbral sería aceptada la palabra x=001 por el AP, si la probabilidad de recibir un cero es 0.6?. Razone su respuesta.

(1,0,0)xM(0)xM(0)xM(1)=(?,?,0.84)
Umbral<=0.84

d) **(0,75 puntos)** Suponiendo que la probabilidad de recibir un cero es 0.9, y que el estado inicial del autómata podría ser q_0 o q_1 con cierta probabilidad, pero nunca q_2, ¿cuál tendría que ser el estado inicial más probable para que la palabra x=00, fuera una palabra reconocida por el AP de manera segura?.
vector de estados iniciales (s,1-s,0)
(s,1-s,0)xM(0)xM(0)=(?,?, 0.1*0.9+(1-s)*0.1)
Para que se cumpla, 0.1*0.9+(1-s)*0.1=1, s tendría que ser un valor negativo, luego, nunca podrá ser aceptada de manera segura. Se observa que 0.09+(1-s)*0.1=0.19-0.1s, toma valores más altos cuanto más pequeño es s, luego para que fuera aceptada con cierta probabilidad, el estado inicial más probable tendría que ser q1.

<u>Pregunta 4 (3,5 puntos)</u>

a) **(1 punto)** Dadas las siguientes leyes de aprendizaje para calcular los parámetros del perceptron simple, explique las diferencias y similitudes entre ellas.

Ley 1: $w_i\,(t+1) = w_i\,(t) + d(x) *\cdot x_i$ $\quad$ $u(t+1) = u(t) + d(x)$ i=1,2,...n

Ley 2: $w_i\,(t+1) = w_i\,(t) + \alpha *\cdot d(x) *\cdot x_i$ $\quad$ $u(t+1) = u(t) + \alpha *\cdot d(x)$ i=1,2,...n

Ley 3: $w_i(t+1) = w_i(t) + (d(x) - y(x)) * x_i$ $\quad$ $u(t+1) = u(t) + (d(x) - y(x))$ i=1,2,...n

Siendo: $(w_1, w_2,..., w_n)$ los pesos del perceptron simple; u el umbral de la neurona de salida; α la razón de aprendizje; d(x) la salida deseada para el patrón de entrada $x=(x_1, x_2,..., x_n)$; $y(x)= f(x_1\cdot w_1 + x_2\cdot w_2 + ... + x_n\cdot w_n+u)$ y f la función umbral.

La ley 2 introduce la razón de aprendizaje, parámetro que permite que patrones bien clasificados no pasen a estar mal clasificados. Controla el cambio que sufren los parámetros, influyendo en la velicidad de convergencia.

La ley 3 aplicada al perceptron simple tiene el mismo comportameinto que la ley 1, pues aunque considera la diferencia entre la salida deseada y la salida de la red, en el caso del perceptron simple, esta diferencia es siempre constante (función umbral en la salda). Es la ley que debe utilizarse cuando la salida deseada se define como 0 o 1

b) **(2 puntos)**Construir un perceptron simple, utilizando algunas de las leyes de aprendizaje del apartado anterior, para agrupar en dos clases los datos que se muestran en la siguiente tabla, datos que reflejan si una persona tiene o no el peso ideal en función de su altura y su peso.

	Altura (metros)	**Peso (kg.)**	**Peso Ideal**
Dato 1	1.5	50	SI
Dato 2	1.6	52	SI
Dato 3	1.7	58	SI
Dato 4	1.65	55	SI
Dato 5	1.6	62	NO
Dato 6	1.7	65	NO
Dato 7	1.55	60	NO

Nota importante: Para la construcción de dicho perceptron simple debe realizar como mínimo dos ciclos de aprendizaje.

Con el objetivo de encontrar la solución en un número pequeño de iteraciones, se parte de un hiperplano inicial que clasifique correctamente todos los patrones, salvo uno, para realizar así al menos un ciclo. Posteriormente se utilizará la ley 2 con razón de aprendizaje adecuada para que clasifique bien dicho patrón, sin perjudicar a las clasificaciones correctas.

Con este propósito los parámetros iniciales se fijan a W1=0, w2=1, U=-56, razón de aprendizaje=0.011
En las tablas se muestran las iteraciones;

	Parámetros Iniciales		Parámetros (iteración 1)
w1	0		-0,00187
w2	1		0,942
u	-56		-56,001

	Ciclo 1		Salida deseada	Ciclo2
Dato1	1,6	52	-1	
Salida red	-4		-1	-7,019992
			-1	
Dato2	1,7	58	-1	
Salida red	2		1	-1,368179
Nueva salida	-1,368179		-1	
Dato3	1,5	50	-1	
Salida red	-8,903805		-1	
Dato4	1,65	55	-1	
Salida red	-4,1940855		-1	-4,1940855
Dato5	1,6	62	1	
Salida red	2,400008		1	2,400008
Dato6	1,7	65	1	
Salida red	5,225821		y=1	5,225821
Dato7	1,55	60	1	
Salida red	0,5161015		y=1	0,5161015

El ciclo 2 es necesario hacerlo pues en el primer ciclo se han modificado los parámetros de la red, una vez que el dato1 ya estaba bien clasificado.
c) **(0,5 puntos)** ¿Se podría utilizar un perceptron simple si al conjunto de datos anterior se le añade el patrón (1.6, 48, NO)?. Si la respuesta es afirmativa, ¿habría que hacer algún cambio en el perceptron simple construido en el apartado anterior?. Si la respuesta es negativa, indique si sería posible abordar el problema con redes de neuronas artificiales.
No, pues se trata entonces de un problema de clasificación no lineal. Se puede resolver con el perceptron multicapa.