



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
 ESCUELA DE INGENIERÍA
 DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC1253 — Matemáticas Discretas — 1' 2016

Tarea 4 — Respuesta Pregunta 1

1. PD R es un *rectángulo* si, y solo si, para todo $a_1, b_1, a_2, b_2 \in A$ se cumple que:

$$\text{si } (a_1, b_1) \in R \text{ y } (a_2, b_2) \in R, \text{ entonces } (a_1, b_2) \in R.$$

Demostración para ambas direcciones:

(\Rightarrow) Si R es un rectángulo

$$\text{PD } \forall a_1, b_1, a_2, b_2 \in A. (a_1, b_1) \in R \wedge (a_2, b_2) \in R \rightarrow (a_1, b_2) \in R$$

Se sabe que R es un rectángulo, por lo que $R = B \times C$, con $B, C \subseteq A$. De la definición de producto cartesiano, se tiene que:

$$B \times C = \{(b, c) \mid b \in B \wedge c \in C\}$$

Por lo tanto, si $(a_1, b_1) \in R$, entonces se tiene que, como R es rectángulo, y un producto cartesiano:

$$(a_1, b_1) \in R \rightarrow a_1 \in B \wedge b_1 \in C$$

Análogamente, si $(a_2, b_2) \in R$, se tiene que:

$$(a_2, b_2) \in R \rightarrow a_2 \in B \wedge b_2 \in C$$

Ahora, como R es un producto cartesiano, se tiene que, por definición todos los (a, b) con $a \in B$ y con $b \in C$, estarán presentes en R . Ahora, se vio que $a_1 \in B$ y que $b_2 \in C$, por lo que, $(a_1, b_2) \in R$.

$$(\Leftarrow) \text{ Si } \forall a_1, b_1, a_2, b_2 \in A. (a_1, b_1) \in R \wedge (a_2, b_2) \in R \rightarrow (a_1, b_2) \in R$$

PD R es un rectángulo

Si se toman dos pares $(a_1, b_1) \in R$ y $(a_2, b_2) \in R$, tales que cumplan la hipótesis, se tiene que existirá un tercer par $(a_1, b_2) \in R$. Al ser un cuantificador universal, se tiene que todos los pares de pares de esa forma, que existan en R , cumplan dicha condición. Así se puede concluir que, si se reordena la hipótesis, el par (a_2, b_1) también estará en R , ya que:

$$\forall a_1, b_1, a_2, b_2 \in A. (a_2, b_2) \in R \wedge (a_1, b_1) \in R \rightarrow (a_2, b_1) \in R$$

Después, si se toman conjuntos B, C , tales que B sea el conjunto de todos los a que se encuentran en un par $(a, b) \in R$, y C el conjunto de todos los b que se encuentran en un par $(a, b) \in R$, se tiene que R será el producto cartesiano de esos dos conjuntos B y C .

Visto de otro modo, se tiene que de todo par de pares presente en R , por hipótesis, se puede llegar a formar los pares que relacionan ordenadamente a los elementos restantes, es decir, en la hipótesis, estaban relacionados a_1 con b_1 como el par (a_1, b_1) y a_2 con b_2 como el par (a_2, b_2) , y se llegaba a que tanto el par (a_1, b_2) y el par (a_2, b_1) estaban en R , es decir, los únicos dos restantes para el producto cartesiano entre $\{a_1, a_2\}$ y $\{b_1, b_2\}$, que es justamente la forma en la que se definieron B y C .

De modo que R es un producto cartesiano, en ese caso entre B y C , y ambos conjuntos están definidos con elementos de A , por lo que, son subconjuntos de A , y entonces, se cumple la condición para que R sea rectángulo. ■

2. Si R es una relación de equivalencia

PD Existe un $n \in \mathbb{N}$ tal que existen n rectángulos simples $R_i \subseteq A \times A$ tales que su unión generalizada es igual a R .

Para demostrar lo anterior, por existencia, mostrando que existe algún caso en el que se cumpla, se demuestra lo pedido.

Se sabe que a una relación de equivalencia $R \subseteq A \times A$ se le puede asociar un conjunto cociente, que es una partición de A . En particular si se toma A/R , se tiene que $|A/R| = n$, y ese será el n que se necesita encontrar. Ahora, si se toman rectángulos $R_i = B_i \times B_i$, donde B_i es alguna clase de equivalencia de A/R , es decir $[b]_R$, con $[b]_R \subseteq A$ tal que $b \in A$, entonces R_i será un rectángulo simple.

Ahora, la unión de esos rectángulos simples, es R , ya que por un lado $R \supseteq \bigcup_{i=1}^n R_i$, porque R tiene a todos los pares que están relacionados bajo R , y el producto cartesiano entre una clase de equivalencia y si misma, son pares presentes en R , es decir pares de elementos que se relacionan bajo R , esto por definición de clase de equivalencia, ya que todos sus elementos están relacionados entre sí. Al ser el producto cartesiano de todas las clases de equivalencia bajo R con ellas mismas, estos serán todos los pares.

Por otro lado, $R \subseteq \bigcup_{i=1}^n R_i$ es claro, ya que al ser el conjunto cociente una partición, no hay elementos que estén relacionados y que no se encuentren en alguna clase de equivalencia, de modo que la unión del producto cartesiano de ellas, contendrá a todos los pares de la relación, de hecho, de no hacerlo, se incumpliría alguna de las condiciones de relación de equivalencia, es decir, transitividad, reflexividad y simetría.

Luego, habiendo mostrado que $R \supseteq \bigcup_{i=1}^n R_i$ y $R \subseteq \bigcup_{i=1}^n R_i$, se tiene que $R = \bigcup_{i=1}^n R_i$, en particular para $n = |A/R|$, por lo que se demuestra lo pedido. ■

La otra dirección, esto es: si existe un $n \in \mathbb{N}$ tal que existen n rectángulos R_i simples, subconjuntos de $A \times A$, tal que, $R = \bigcup_{i=1}^n R_i$, entonces R es una relación de equivalencia, no es cierta, ya que si se toma un n arbitrario, de modo que existan n rectángulos simples $R_i = B_i \times B_i$, con $B_i \in A$, tales que $R = \bigcup_{i=1}^n R_i$, como por ejemplo:

$$n = 2$$

$$A = \{a, b, c\}$$

$$B_1 = \{a, b\}$$

$$B_2 = \{b, c\}$$

$$R_1 = B_1 \times B_1 = \{(a, b), (b, a), (a, a), (b, b)\}$$

$$R_2 = B_2 \times B_2 = \{(b, c), (c, b), (b, b), (c, c)\}$$

$$R = R_1 \cup R_2 = \{(a, b), (b, a), (a, a), (b, b), (b, c), (c, b), (b, b), (c, c)\}$$

Se ve claramente que R cumple con el lado derecho de la implicancia (que sería el lado izquierdo de la dirección contraria), pero no cumple el lado izquierdo (que sería el derecho en la dirección inversa), porque no es una relación de equivalencia, ya que, no cumple las 3 propiedades necesarias (simetría, reflexividad y transitividad) juntas, de hecho no es transitiva, ya que, existiendo $(a, b) \in R$ y $(b, c) \in R$, pasa que $(a, c) \notin R$. Por lo tanto, se muestra con un contra-ejemplo que no es cierta la dirección contraria.



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
 ESCUELA DE INGENIERÍA
 DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC1253 — Matemáticas Discretas — 1' 2016

Tarea 4 — Respuesta Pregunta 2

1. Para demostrar que $\mathcal{C}(f^\downarrow) \leq \log_2(n+1) + 1$, se considera la definición de $\mathcal{C}(f)$, la que señala que es el máximo número de bits enviados para el mejor protocolo de f .

La información que se tiene que enviar de la función f^\downarrow , es el número de la posición donde se encuentra el primer uno en la cadena de bits en binario o el largo de la cadena $+1$ en caso de que no existan 1s, en binario.

Como la definición de complejidad señala que es el máximo número de bits para el mejor protocolo, es decir el peor caso, hay que centrarse en el caso en el que el número es mayor, ya que en el algoritmo para transformar un número a binario, la cantidad de bits de la cadena binaria es creciente a medida que crece el número. Así, interesa saber el número de bits que tiene la codificación binaria de un número, ya que esa es la información que se enviará.

El algoritmo para codificar un número a binario siempre entrega una cadena de bits de largo igual a $\log_2(n) + 1$, es decir, si h_{bits} es la función que codifica un número n a binario, $|h_{bits}(n)| = \log_2(n) + 1$. Por ejemplo la codificación del número 4 en binario es 100, y claramente $|100| = 3 = \log_2(4) + 1$. Nota: Para los casos donde n no es potencia de 2, se toma el piso de $\log_2(n)$.

Ahora, sabiendo eso, se tiene que el peor caso para el operador $^\downarrow$ es cuando no existen 1s en la cadena, de modo que el resultado es $n + 1$. De ahí, utilizando la cantidad de bits que se obtiene luego de codificar un número a binario, se tiene que para $n + 1$, la cantidad es $\log_2(n + 1) + 1$.

Así, si consideramos un protocolo para f^\downarrow (no tiene que ser el mejor porque se está demostrando que existe una cota superior), que tiene a g_L como la función que realiza w^\downarrow sobre una cadena w , para después poder comparar en g_B , el máximo número de bits que se envía por parte de g_L a g_B , es $\log_2(n + 1) + 1$. No necesariamente es el mejor protocolo, pero como se dijo, esto no es necesario, ya que lo pedido es acotar por dicha cota.

De ahí que, usando lo anterior, se demuestra que $\mathcal{C}(f^\downarrow) \leq \log_2(n + 1) + 1$. ■

2. Si se considera la función $f^=$, se puede ver que se necesita toda la información de una cadena u para compararla según la igualdad con otra cadena v , ya que la igualdad de palabras es el resultado de la comparación de cada uno de los bits de una palabra con otra, en este caso, de bits.

Entonces, es evidente ver que se requiere mandar toda la palabra de bits en g_L , para que luego en g_B se compare con la otra. Por lo tanto, para cualquier protocolo, se requerirá que g_L envíe una cantidad de bits igual a el largo de la palabra (n), de modo que $\mathcal{C}(f^=) \geq n$.

Visto de otro modo que $\mathcal{C}(f^=) \leq n$, implicaría que g_L para algún protocolo enviase menos de n bits para realizar la comparación. Esto implicaría que la palabra se está "achicando" de algún modo, con algún algoritmo. Eso se podría realizar si no importaran, o los valores o las posiciones de todos los bits, como en el operador anterior $^\downarrow$. Pero en este caso, importan ambos, por lo que no hay otra opción que enviar la palabra entera, para cualquier protocolo. ■