Pontificia Universidad Católica de Chile
Facultad de Ingeniería
Departamento de Ciencias de la computación
Curso: Matemáticas discretas
Ayudantes: Francisca Caprile, Catalina Ortega, Matías Fernández e Ignacio Vergara

Ayudantía 12

17 de noviembre de 2023

2º semestre 2023 - Profesores G. Diéguez - S. Bugedo - N. Alvarado- B. Barías

---

Ejercicio 1 | Relaciones

Dados dos números $q_1, q_2 \in \mathbb{Q}$, se define la relación Q como:

$$q_1 \; Q \; q_2 \text{ si y solo si } q_1 - q_2 \in \mathbb{Z}.$$

Demuestre que es una relación de equivalencia.

Solución

Para demostrar que es una relación de equivalencia, se debe demostrar que es reflexiva, simétrica y transitiva.

Reflexiva: Por demostrar que $q \; Q \; q$ para todo $q \in \mathbb{Q}$. Debemos demostrar que $q - q = 0 \in \mathbb{Z}$, lo cual es cierto para cualquier q. Por lo tanto, es reflexiva.

Simétrica: Por demostrar que si $q_1 \; Q \; q_2$, entonces $q_2 \; Q \; q_1$. Si $q_1 = q_2$, la propiedad es trivial. En caso contrario, si $q_1 \; Q \; q_2$, entonces $q_1 - q_2 = k \in \mathbb{Z}$. Se debe demostrar que $q_2 - q_1 = m \in \mathbb{Z}$, lo cual es cierto ya que $q_2 - q_1 = -k \in \mathbb{Z}$. Si $k \in \mathbb{Z}$ entonces $-k \in \mathbb{Z}$ Por lo tanto, es simétrica.

Transitiva: Por demostrar que si $q_1 \; Q \; q_2$ y $q_2 \; Q \; q_3$, entonces $q_1 \; Q \; q_3$. Si $q_1 \; Q \; q_2$ y $q_2 \; Q \; q_3$, entonces $q_1 - q_2 = k$ y $q_2 - q_3 = l$, con $k, l \in \mathbb{Z}$. Se debe demostrar que $q_1 - q_3 = m \in \mathbb{Z}$,

$$\begin{aligned} q_1 - q_2 &= k \\ q_1 &= k + q_2 \\ q_2 - q_3 &= l \\ -q_3 &= l - q_2 \\ q_1 - q_3 &= k + q_2 + l - q_2 \\ q_3 &= k + l \end{aligned}$$

Luego $k + l \in \mathbb{Z}$, ya que Z es cerrado bajo la suma.

Por lo tanto, la relación es una relación de equivalencia.

Ejercicio 2 | Funciones y cardinalidad

Sea el conjunto

$$\mathcal{I} := \{A \subseteq \mathbb{N} | A \text{ es infinito y } \mathbb{N} \backslash A \text{ es infinito}\}$$

Demuestre que $\mathcal{I}$ es equinumeroso con $\mathcal{P}(\mathbb{N})$

Solución

Para demostrar que $\mathcal{I}$ es equinumeroso con $\mathcal{P}(\mathbb{N})$ utilizaremos el teorema de Schröder-Bernstein. Entonces debemos formar dos funciones $f : \mathcal{I} \to \mathcal{P}(\mathbb{N})$, $g : \mathcal{P}(\mathbb{N}) \to \mathcal{I}$ ambas inyectivas.

En primer lugar se puede observar que todo elemento en $\mathcal{I}$ se encuentra en $\mathcal{P}(\mathbb{N})$, por lo cual es directo que una función f inyectiva a la cual recurrir es la identidad, $f(X) = X$, la cual es inyectiva trivialmente.
Para formar $g : \mathcal{P}(\mathbb{N}) \to \mathcal{I}$ es importante notar que existen tres casos posibles para $X \subseteq \mathbb{N}$,

1. X es finito.
2. X es infinito pero tiene complemento finito.
3. X es infinito y tiene complemento infinito.

Considerando esto, la idea es formar g de tal manera que a partir de $x \in X$ lo asociaremos con particiones infinitas de $\mathbb{N}$.
Considerando entonces, $\mathbb{P}$ y $\mathbb{I}$ como los conjuntos de números pares e impares respectivamente, sabemos que existe una biyección entre $\mathbb{I}$ y $\mathbb{N}$, llamaremos a ésta función $h : \mathbb{N} \to \mathbb{I}$. Definimos entonces $g : \mathcal{P}(\mathbb{N}) \to \mathcal{I}$ como

$$g(X) = \begin{cases} \{h(x)|x \in X\} \cup \mathbb{P} & \text{si } X \text{ es finito} \\ \{h(x)|x \in X\} & \text{si } X \text{ es infinito} \end{cases}$$

Ahora, para mostrar que ésta función efectivamente es la que necesitamos falta mostrar que todas las imágenes de g están en $\mathcal{I}$, esto es, son subconjuntos de $\mathbb{N}$ infinitos o con complementos infinito. Posterior a esto, falta demostrar que la función es inyectiva.
Para mostrar que todas las imágenes de g están en $\mathcal{I}$:

- Todas las imágenes están compuestas por números en $\mathbb{I}$ o en $\mathbb{P}$, por lo cual se cumple que son subconjuntos de $\mathbb{N}$.
- En el caso de que $X \subseteq \mathbb{N}$ es finito, por definición de g tenemos que $\mathbb{P} \subseteq g(X)$, por lo cual como $\mathbb{P}$ es infinito, necesariamente $g(X)$ también lo es. Además su complemento, $\mathbb{N} \backslash g(X)$ contiene a todos los impares, salvo una cantidad finita de ellos los cuales fueron mapeados desde X usando h. Por lo tanto su complemento también es infinito obteniendo así que $g(X) \in \mathcal{I}$.
- En el caso de que X sea infinito, $g(X)$ también lo es, pues todo elemento de X tendrá como imagen un numero impar diferente debido a que h es una biyección. Además, su complemento $\mathbb{N} \backslash g(X)$ contiene a todos los numeros pares, puesto que todos los elementos en $g(X)$ son impares pues son imagenes de h. Por lo tanto su complemento también es infinito obteniendo así que $g(X) \in \mathcal{I}$.

Finalmente falta mostrar que g es inyectiva,

$$g(X) = g(Y) \to X = Y$$

Para ello, se tienen los siguientes casos

- Si X e y son infinitos. Supongamos que $g(X) = g(Y)$ tenemos entonces que,

$$\{h(x)|x \in X\} = \{h(y)|y \in Y\}$$

además, sabemos que h es invertibles pues es inyectiva, luego

$$\{h^{-1}(h(x))|x \in X\} = \{h^{-1}(h(y))|y \in Y\}$$

$$\{x|x \in X\} = \{y|y \in Y\}$$

$$X = Y$$

Por lo tanto en éste caso g si es inyectiva.

- Si X e y son finitos.Supongamos que $g(X) = g(Y)$ tenemos entonces que,

$$\{h(x)|x \in X\} \cup \mathbb{P} = \{h(y)|y \in Y\} \cup \mathbb{P}$$

pero sabemos que $\{h(x)|x \in X\}$ y $\{h(y)|y \in Y\}$ no tienen números pares, por lo cual necesariamente se cumple que

$$\{h(x)|x \in X\} = \{h(y)|y \in Y\}$$

llegando a la misma situación que el caso anterior, por lo tanto en éste caso g también es inyectiva.

- Si X e Y no son finitos o infinitos a la vez. Para este caso no es posible asumir que $g(X) = g(Y)$ pues alguna de las imagenes va a contener numeros impares y la otra no.

Finalmente podemos concluir que existe $g : \mathcal{P}(\mathbb{N}) \to \mathcal{I}$ inyectiva, lo que sumado a $f : \mathcal{I} \to \mathcal{P}(\mathbb{N})$ por teorema de Schröder-Bernstein nos permite concluir que $\mathcal{I}$ es equinumeroso con $\mathcal{P}(\mathbb{N})$.

Ejercicio 3 | Funciones y cardinalidad (I2 2021-2)

Sean A, B, C y D conjuntos infinitos tales que $A \approx C$ y $B \approx D$. Considere los siguientes conjuntos:

$$\mathcal{F} = \{f|f : A \to B \text{ es una función}\}$$

$$\mathcal{G} = \{g|g : C \to D \text{ es una función}\}$$

Demuestre que $F \approx G$.

Solución

Dado que $A \approx C$ y $B \approx D$, existen funciones biyectivas $\alpha : A \to C$ y $\beta : B \to D$. Consideremos las siguientes relaciones:

- $h_1 \subseteq \mathcal{F} \times \mathcal{G}$ tal que $(f, g) \in h_1$ si y solo si

$$\forall a \in A, \forall b \in B \text{ se cumple que } (a, b) \in f \text{ si y solo si } (\alpha(a), \beta(b)) \in g.$$

- $h_2 \subseteq \mathcal{G} \times \mathcal{F}$ tal que $(g, f) \in h_2$ si y solo si

$$\forall c \in C, \forall d \in D \text{ se cumple que } (c, d) \in g \text{ si y solo si } (\alpha^{-1}(c), \beta^{-1}(d)) \in f.$$

Por el teorema de Schröeder-Bernstein, demostrar que ambas relaciones son funciones inyectivas implica que $\mathcal{F} \approx \mathcal{G}$. Demostraremos esto para h_1, siendo la demostración para h_2 completamente análoga.

Demostración para h_1

Función: Sean $f \in \mathcal{F}$ y $g_1, g_2 \in \mathcal{G}$ tales que $(f, g_1) \in h_1$ y $(f, g_2) \in h_1$. Debemos demostrar que $g_1 = g_2$.

- $g_1 \subseteq g_2$: Sea $(c, d) \in g_1$. Como $(f, g_1) \in h1$, existen $a \in A$ y $b \in B$ tales que $f(a) = b$, $\alpha(a) = c$, y $\beta(b) = d$. Entonces, como $(f, g_2) \in h1$, también se cumple que $(c, d) \in g_2$.
- $g_2 \subseteq g_1$: Análogo al caso anterior.

Función total: Dado $f \in \mathcal{F}$, debemos demostrar que existe $g \in \mathcal{G}$ tal que $(f, g) \in h_1$. Definimos $g \subseteq C \times D$ como:

$$g = \{(c, d) \mid \exists a \in A, \exists b \in B \text{ tales que } f(a) = b \wedge \alpha(a) = c \wedge \beta(b) = d\}$$

Debemos demostrar que g es efectivamente una función:

- Función: Sean (c, d_1) y $(c, d_2) \in g$. Por definición de g, existen $a \in A$ y $b_1, b_2 \in B$ tales que $\alpha(a) = c$, $\beta(b_1) = d_1$, $\beta(b_2) = d_2$, y $f(a) = b_1$ y $f(a) = b_2$. Como f es función, tenemos que $b_1 = b_2$, y por lo tanto g es función.
- Función total: Dado $c \in C$, como α es una función biyectiva, existe $a \in A$ tal que $\alpha(a) = c$. Como f es total, existe $b \in B$ tal que $f(a) = b$, y como β también es total, existe $d \in D$ tal que $\beta(b) = d$. Por lo tanto, por definición de g, se cumple que $g(c) = d$.

Inyectiva

Supongamos que $h1(f_1) = h1(f_2)$ (1). Debemos demostrar que $f_1 = f_2$.

- $f_1 \subseteq f_2$: Sea $(a_1, b_1) \in f_1$, es decir, $f_1(a_1) = b_1$. Sean también $c \in C$, $d \in D$ tales que $\alpha(a_1) = c$ y $\beta(b_1) = d$. Por definición de h_1, $(c, d) \in h_1(f_1)$, y por (1) se tiene que $(c, d) \in h_1(f_2)$. Luego, existen $a_2 \in A$ y $b_2 \in B$ tales que $f_2(a_2) = b_2$, $\alpha(a_2) = c$, y $\beta(b_2) = d$. Como α y β son biyectivas, $a_1 = a_2$ y $b_1 = b_2$. Por lo tanto, $f_2(a_1) = b_1$, y entonces $(a_1, b_1) \in f_2$.
- $f_2 \subseteq f_1$: Análogo al caso anterior

Ejercicio 4 | Algoritmos y complejidad

1. Considere el siguiente algoritmo:

Algorithm 1: theavengersaredead(n)

Data: n
Result: i

```
k ← 1;
for i ← 1 to n do
    k ← k × n;
i ← 1;
while i ≤ k do
    i ← i × 2;
return i;
```

Encuentre una función f y demuestre (usando la definición formal de la notación Θ) que el tiempo de theavengersaredead en términos de n es $\Theta(f(n))$.

2. Demuestre que si $\mathrm{f}_1 \in \mathcal{O}(g_1)$ y $f_2 \in \mathcal{O}(g_2)$, entonces $f_1 + f_2 \in \mathcal{O}(\max\{g_1, g_2\})$

Solución

1. El "for"del algoritmo toma n pasos, y al terminar queda $k = n^n$. Luego, el while se ejecuta hasta que $i \leq k$. Es claro que i se ve de la forma $i = 2^l$ donde l es el número de iteraciones que lleva el algoritmo. Por lo tanto el "while"se ejecuta hasta que $2^l > n^n$. Despejamos l :

$$\begin{aligned} &2^l > n^n \quad / \log_2(\cdot) \\ &l > n \log_2 n \end{aligned}$$

Entonces el "while" hará $n \log n$ iteraciones. Por lo tanto, la función del tiempo del algoritmo es:

$$T(n) = n + n \log n + O(1)$$

Finalmente, usamos $f(n) = n \log n$ y el algoritmo es claramente $\Theta(f(n))$.

2. Dado que $f_1(n) \in \mathcal{O}(g_1(n))$ y $f_2(n) \in \mathcal{O}(g_2(n))$, sabemos que:
   - Existe $c_1 > 0, n_0^1 \in \mathbb{N}$ tal que $f_1(n) \leq c_1 \cdot g_1(n)$ para todo $n \geq n_0^1$.
   - Existe $c_2 > 0, n_0^2 \in \mathbb{N}$ tal que $f_2(n) \leq c_2 \cdot g_2(n)$ para todo $n \geq n_0^2$.

   Si $n_0 = \text{máx}\left\{n_0^1, n_0^2\right\}$, entonces para todo $n \geq n_0$:

$$\begin{aligned} f_1(n) + f_2(n) &\leq c_1 \cdot g_1(n) + c_2 \cdot g_2(n) \\ &\leq c_1 \cdot \text{máx}\{g_1(n), g_2(n)\} + c_2 \cdot \text{máx}\{g_1(n), g_2(n)\} \\ &\leq C \cdot \text{máx}\{g_1(n), g_2(n)\} \end{aligned}$$

   Queda demostrado lo pedido ya que existe un $C = c_1 + c_2$ y existe un $n_0 \in \mathbb{N}$ tal que:

$$f_1(n) + f_2(n) \in \mathcal{O}(\text{máx}\{g_1(n), g_2(n)\})$$