Pontificia Universidad Católica de Chile
Departamento de Ciencia de la Computación
IIC1253 - Matemáticas Discretas

# Interrogación 2

10 de noviembre de 2021
Profesores: Marco Bucchi - Gabriel Diéguez - Fernando Suárez

## Instrucciones

- La duración de la interrogación es de 2 horas.
- Durante la evaluación **no puede** hacer uso de sus apuntes o slides del curso.
- Rellene sus datos en cada hoja de respuesta que utilice.
- Cada pregunta debe responderse en hojas separadas.
- Entregue al menos una hoja por pregunta.
    - Si entrega la pregunta **completamente en blanco**, tiene nota mínima 1.5 en vez de 1.0 en la pregunta entregada.
- Escriba sus respuestas con lápiz pasta. Por el uso de lápiz mina usted pierde el derecho a recorreción.

## Pregunta 1 - Funciones (vista en clases)

Sean $A$ y $B$ conjuntos. Demuestre que si $f : A \to B$ es una función biyectiva, entonces la relación inversa $f^{-1}$ es una función biyectiva de $B$ en $A$.

### Solución

- Función: supongamos que $yf^{-1}x_1$ e $yf^{-1}x_2$, con $y \in B$ y $x_1, x_2 \in A$. Por definición de relación inversa, esto significa que $x_1fy$ y $x_2fy$. Como $f$ es inyectiva, $x_1 = x_2$, y por lo tanto $f^{-1}$ es función.
- Total: como $f$ es sobre, para todo $y \in B$ existe $x \in A$ tal que $y = f(x)$. Luego, para todo $y \in B$ existe $x \in A$ tal que $x = f^{-1}(y)$, y por lo tanto $f^{-1}$ es total.
- Inyectiva: supongamos que $f^{-1}(y_1) = f^{-1}(y_2) = x$, con $y_1, y_2 \in B$ y $x \in A$. Por definición de relación inversa, esto significa que $f(x) = y_1$ y $f(x) = y_2$. Como $f$ es función, $y_1 = y_2$, y por lo tanto $f^{-1}$ es inyectiva.

- Sobre: como $f$ es total, para todo $x \in A$ existe $y \in B$ tal que $y = f(x)$. Luego, para todo $x \in A$ existe $y \in B$ tal que $x = f^{-1}(y)$, y por lo tanto $f^{-1}$ es sobre.

**Pauta (6 pts.)**

1,5 puntos por cada propiedad.
Puntajes parciales y soluciones alternativas a criterio del corrector.

## Pregunta 2 - Funciones y Cardinalidad

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

$$\mathcal{F} = \{f \mid f : A \to B \text{ es una función}\}$$
$$\mathcal{G} = \{g \mid g : C \to D \text{ es una función}\}$$

Demuestre que $\mathcal{F} \approx \mathcal{G}$.

**Solución**

Como $A \approx C$ y $B \approx D$, existen funciones biyectivas $\alpha : A \to C$ y $\beta : B \to D$. Considere 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

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

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

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

- $g_1 \subseteq g_2$: sea $(c, d) \in g_1$. Como $(f, g_1) \in h_1$, existen $a \in A$ y $b \in B$ tales que $f(a) = b$, $\alpha(a) = c$ y $\beta(b) = d$. Entonces, como $(f, g_2) \in h_1$, también se cumple que $(c, d) \in g_2$.
- $g_2 \subseteq g_1$: análoga a la anterior.

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

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

Ahora debemos demostrar que $g$ efectivamente es una función:

- Función: sean $(c, d_1), (c, d_2) \in g$. 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 más importante, $f(a) = b_1$ 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 $\alpha$ 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 $\beta$ 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 $h_1(f_1) = h_1(f_2)$ (1). Por 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 $\beta(b_2) = d$, y como $\alpha$ y $\beta$ son biyectivas, $a_1 = a_2$ y $b_1 = b_2$. Por lo tanto, se tiene que $f_2(a_1) = b_1$, y entonces $(a_1, b_1) \in f_2$.

- $f_2 \subseteq f_1$: análoga a la anterior.

**Pauta (6 pts.)**

Para cada función:

- 1 punto por dar la función.
- 0,5 puntos por demostrar que es función.
- 0,5 puntos por demostrar que es total.
- 1 punto por demostrar que es inyectiva.

Se aceptará dar la demostración de una de las funciones y argumentar que la otra es análoga. Puntajes parciales y soluciones alternativas a criterio del corrector.

## Pregunta 3 - Análisis de algoritmos

Considere el siguiente algoritmo iterativo:

**Precondiciones:** un arreglo $A = [a_0, \ldots, a_{n-1}]$, su largo $n$ y un número $x$.
**Postcondiciones:** el algoritmo retorna la cantidad de veces que $x$ aparece en $A$.

$\textsc{Contar}(A, n, x)$

$k = 0$
**for** $i = 0$ **to** $n - 1$ **do**
    **if** $A[i] == x$ **then**
        $k$+=1
    **end if**
**end for**
**return** $k$

Considere ahora el siguiente algoritmo recursivo:

**Precondiciones:** un arreglo $A = [a_0, \ldots, a_{n-1}]$ no vacío y su largo $n$
**Postcondiciones:** el algoritmo retorna un número que se repite más de $\lfloor \frac{n}{2} \rfloor$ veces en $A$, o $-1$ si no existe.

$\textsc{Mayoría}(A, n)$

**if** $n == 1$ **then**
    **return** $A[0]$
**end if**
$p = \left\lfloor \frac{n}{2} \right\rfloor$
$m_1 = \textsc{Mayoría}(A[:p], p)$
$m_2 = \textsc{Mayoría}(A[p:], n - p)$
**if** $m_1 \neq -1$ **and** $\textsc{Contar}(A, n, m_1) > p$ **then**
    **return** $m_1$
**else if** $m_2 \neq -1$ **and** $\textsc{Contar}(A, n, m_2) > p$ **then**
    **return** $m_2$
**end if**
**return** $-1$

a) Encuentre una expresión $T(n)$ para el número de comparaciones que realiza el algoritmo $\textsc{Mayoría}$ al ser llamado con un arreglo de largo $n$.

b) Calcule la complejidad en el peor caso de $T(n)$ en notación $O(\cdot)$ sin utilizar el teorema maestro.

c) Corrobore su resultado en b) utilizando el teorema maestro. ¿Le parece un buen algoritmo?

**Solución**

a) Buscamos determinar la cantidad de comparaciones $T(n)$, donde $n$ es el largo de la lista. Consideremos la siguente ecuación de recurrencia[1]:

$$T(n) = \begin{cases} 1 & n = 1 \\ T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) + 2n + 3 & n > 1 \end{cases}$$

b) Utilizaremos el reemplazo $n = 2^k$ con $k \in \mathbb{N}$.

Luego, $T(2^k) = 2T(2^{k-1}) + 2^{k+1} + 3$, siguiendo esta ecuación de recurrencia resulta:

$$\begin{aligned} T(n) &= 2T(2^{k-1}) + 1 \cdot 2^{k+1} + 3 \\ T(n) &= 4T(2^{k-2}) + 2 \cdot 2^{k+1} + 9 \\ T(n) &= 8T(2^{k-2}) + 3 \cdot 2^{k+1} + 21 \\ &\vdots \\ T(n) &= 2^i T(2^{k-i}) + 2i \cdot 2^k + 3 \cdot 2^i - 3 \end{aligned}$$

si consideramos la iteración $i = k$:

$$T(n) = 2^k \cdot T(1) + 2k \cdot 2^k + 3 \cdot 2^k - 3$$

donde $k = \log_2(n)$, luego podemos reescribir la recurrencia como:

$$T(n) = n + 2n \log_2(n) + 3 \log_2(n) - 3$$

con lo que

$$T(n) \in O(n \log_2(n) \mid \text{POTENCIA}_2)$$

Además, sabemos que $n \log_2(n)$ es asintóticamente no decreciente y 2-armónica.

Por lo tanto, si logramos demostrar que $T(n)$ es asintóticamente no decreciente podremos concluir que $T(n) \in O(n \log_2(n))$.

**PD**: $(\exists n_0 \in \mathbb{N})(\forall n \geq n_0)(T(n) \leq T(n+1))$

Tomamos $n_0 = 2$. **PD**: $(\forall n \geq 2)(T(n) \leq T(n+1))$

Para facilitar la demostración vamos a demostrar un resultado más fuerte:

**PD**: $(\forall n \geq 2)(T(m) \leq T(n))$, con $2 \leq m \leq n$

[1]Para el caso recursivo consideramos las comparaciones de las líneas 1, 7 y 9. Sin embargo, dado que no afectan en el resultado pueden omitirse y se considerará correcto.

De este resultado se deduce que $T(n)$ es asintóticamente no decreciente (en lugar de demostrarlo para el antecesor, lo hacemos para todos los anteriores).

Demostraremos esto por inducción fuerte sobre $n$:

**<u>BI:</u>** Para $n = 2$, el único $m$ que debemos mostrar es el mismo 2, y en este caso la propiedad se cumple trivialmente.

**<u>HI:</u>** Supongamos que $\forall k \in \{2, \ldots, n-1\}$ se cumple que

$$T(m) \leq T(k) \text{ , con } 2 \leq m \leq k$$

**<u>TI:</u>** PD: $(\forall m \leq n)(T(m) \leq T(n))$, con $n, m \geq 2$.

Como $2 \leq m \leq n$ se cumple que $1 \leq \left\lfloor \frac{m}{2} \right\rfloor \leq \left\lfloor \frac{n}{2} \right\rfloor$ y que $1 \leq \left\lceil \frac{m}{2} \right\rceil \leq \left\lceil \frac{n}{2} \right\rceil$. Además, tenemos que $\left\lfloor \frac{n}{2} \right\rfloor < n$ y $\left\lceil \frac{n}{2} \right\rceil < n$, y entonces podemos aplicar la HI:

$$T\left(\left\lfloor \frac{m}{2} \right\rfloor\right) \leq T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) \tag{1}$$
$$T\left(\left\lceil \frac{m}{2} \right\rceil\right) \leq T\left(\left\lceil \frac{n}{2} \right\rceil\right) \tag{2}$$

Sumando (1) y (2)

$$\begin{gathered}
T\left(\left\lfloor \frac{m}{2} \right\rfloor\right) + T\left(\left\lceil \frac{m}{2} \right\rceil\right) \leq T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) \\
T\left(\left\lfloor \frac{m}{2} \right\rfloor\right) + T\left(\left\lceil \frac{m}{2} \right\rceil\right) + 2n \leq T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + 2n \\
T\left(\left\lfloor \frac{m}{2} \right\rfloor\right) + T\left(\left\lceil \frac{m}{2} \right\rceil\right) + 2n + 3 \leq T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + 2n + 3 \\
T(m) \leq T(n)_\square
\end{gathered}$$

Finalmente como $T(n) \in O(n \log_2(n) \mid \text{POTENCIA}_2)$, $T(n)$ y $n \log_2(n)$ son asintóticamente no-decrecientes, y $n \log_2(n)$ es 2-armónica podemos conluir que $T(n) \in O(n \log_2(n))$.

c) Si aplicamos el teorema maestro a nuestra ecuación de recurrencia tendremos los parámetros:

$a_1 = 1$, $a_2 = 1$, $b = 2$, $c = 2$, $d = 1$, $c_0 = 1$, $a_1 + a_2 = 2$, $b^d = 2$

Dado que $a_1 + a_2 = b^d$ entramos en el segundo caso y obtenemos $T(n) \in O(n \log_2(n))$.

<u>Comentario</u>: El algoritmo es más lento que recorrer linealmente la lista y contar las repiticiones de cada número. Sin embargo, utiliza memoria constante ya que no es necesario almacenar todos los números de la lista.

**Pauta (6 pts.)**

- 2 ptos por a) (puede no considerar el +3 en el paso recursivo).
- 1 pto por demostrar $T(n) \in O(n \log_2(n) \mid \text{POTENCIA}_2)$.
- 1 pto por demostrar que $T(n)$ es asintóticamente no-decreciente.
- 2 ptos por c).

Puntajes parciales y soluciones alternativas a criterio del corrector.

## Pregunta 4 - Grafos y Cardinalidad

Sea $A$ un conjunto infinito, y $\Gamma = \{G \mid G \text{ es un grafo tal que } V \subseteq A\}$. Es decir, $\Gamma$ es el conjunto de todos los grafos cuyos vértices son elementos de $A$.

a) Demuestre que si $A$ es enumerable, entonces $\Gamma$ es enumerable.

b) Demuestre que si $A$ no es enumerable, entonces $\Gamma$ no es enumerable.

c) Demuestre que $\Gamma/\cong$ (el conjunto cuociente de $\Gamma$ bajo la relación de isomorfismo de grafos) es enumerable.

**Solución**

a) En primer lugar, los grafos estudiados en el curso son finitos. Esto quiere decir que $V$ será un conjunto finito, digamos de tamaño $n$. Luego, la cantidad de grafos que podemos construir con él (vale decir, la cantidad de relaciones binarias sobre $V$) es finita (se deja como ejercicio determinarla). Por lo tanto, si mostramos que la cantidad de subconjuntos finitos de $A$ (es decir, la cantidad de posibles conjuntos de vértices $V$ para los grafos en $\Gamma$) es enumerable, estaremos mostrando también que $\Gamma$ lo es (pues podríamos enumerar los grafos posibles para cada $V$).

   Como $A$ es enumerable, tenemos una biyección $f : A \to \mathbb{N}$ que nos permite poner los elementos de $A$ en una lista $(a_1, \ldots, a_n, \ldots)$ tal que todos los elementos de $A$ aparecen en ella exactamente una vez cada uno (por simplicidad empezamos en el 1). Enumeramos ahora los subconjuntos finitos de $A$ de la siguiente manera:

   - Empezamos con $\varnothing$.
   - Listamos todos los subconjuntos de $A$ tales que los índices de sus elementos suman 1. Aquí sólo tenemos a $\{a_1\}$.
   - Los que suman 2: $\{a_2\}$.
   - Los que suman 3: $\{a_1, a_2\}, \{a_3\}$.
   - Los que suman 4: $\{a_1, a_3\}, \{a_4\}$.

- Los que suman 5: $\{a_1, a_4\}, \{a_2, a_3\}, \{a_5\}$.
- Los que suman 6: $\{a_1, a_2, a_3\}, \{a_1, a_5\}, \{a_2, a_4\}, \{a_6\}$.
- Y así sucesivamente.

Es claro que para cada suma tenemos una cantidad finita de subconjuntos de $A$ que serán listados. Además, cada subconjunto finito de $A$ tiene una suma finita de sus subíndices, y por lo tanto aparecerá en la lista. Por lo tanto, si asignamos los números naturales a los subconjuntos según esta forma de listarlos, tenemos una biyección entre los subconjuntos finitos de $A$ y $\mathbb{N}$, mostrando entonces que el conjunto de subconjuntos finitos de $A$ es enumerable. En conclusión, según lo argumentado al principio, tenemos que si $A$ es enumerable, entonces $\Gamma$ es enumerable.

b) Tomemos el conjunto $\Gamma_1 = \{G \mid G \text{ es un grafo tal que } V = \{a\}, \text{ con } a \in A\}$ de todos los grafos de tamaño 1 con vértices en $A$. Es claro que $\Gamma_1 \subseteq \Gamma$. Notemos que dado un conjunto de vértices $V$ de tamaño 1, sólo hay un posible grafo para él (el vértice solo, sin aristas), y por lo tanto

$$|\Gamma_1| = |\{\{a\} \mid a \in A\}| = |A|$$

Luego, $\Gamma_1$ no es enumerable, y como $\Gamma_1 \subseteq \Gamma$, concluimos que $\Gamma$ no es enumerable.

c) Debemos determinar la cardinalidad del conjunto $\Gamma/\cong\, = \{[G]_{\cong} \mid G \in \Gamma\}$. En tal caso, dado un grafo $G$ tal que $V \subseteq A$ y $|V| = n$, todos los grafos isomorfos a $G$ pertenecen a la misma clase de equivalencia; vale decir, todos los grafos $G'$ con cualquier conjunto de vértices $V' \subseteq A$ tal que $|V'| = n$ y existe una biyección entre $V$ y $V'$ que mantiene las aristas. Por lo tanto, para cada tamaño de $n$ tenemos una cantidad finita de clases de equivalencia posibles (todos los grafos posibles de tamaño $n$), y por lo tanto podemos enumerar las clases de equivalencia para cada tamaño, con lo que obtenemos una enumeración para $\Gamma/\cong$. Formalmente, definimos una familia de conjuntos

$$G_n = \{[G]_{\cong} \mid G \in \Gamma \text{ tal que } |V| = n\}$$

Por ejemplo, $G_1$ contiene sólo una clase de equivalencia, pues todos los grafos de tamaño 1 son isomorfos. $G_2$ contiene 2 clases de equivalencia (dos vértices solos, y dos vértices conectados por una arista), y en general para cada $G_i$, tendrá una cantidad finita de clases de equivalencia. Entonces, listamos las clases de equivalencia de cada $G_i$ empezando en 1. Es claro que cualquier clase de equivalencia $[G]_{\cong} \in \Gamma$ aparecerá en la enumeración, pues los grafos son finitos, y luego $[G]_{\cong}$ deberá pertenecer a algún $G_i$. Entonces, construimos una biyección entre $\Gamma/\cong$ y $\mathbb{N}$ asignando los números naturales en orden a las clases de equivalencia listadas, con lo que demostramos que $\Gamma/\cong$ es enumerable.

**Pauta (6 pts.)**

2 puntos por demostrar cada una de las propiedades.
Puntajes parciales y soluciones alternativas a criterio del corrector.