IIC1253 - Sección 1 - Segundo Semestre 2014
Profesor: Marcelo Arenas
Ayudantes: Martín Muñoz (*mmunos@uc.cl*) - Matías San Martín (*masanmartin@uc.cl*)

Ayudantía 10.

Cantor

Problema 1. Sea $\mathcal{C}$ el conjunto de todos los conjuntos numerables. ¿Es $\mathcal{C}$ numerable?

Problema 2. Utilice diagonalización para demostrar que el conjunto $\{0,1\}^\omega$ de los strings binarios infinitos no es numerable.

Problema 3. Sea $\mathcal{F} = \{f : \mathbb{N} \to \mathbb{N} \mid f \text{ es función total}\}$. Demuestre que $\mathcal{F}$ no es numerable. ¿Qué pasa si pedimos que f sea inyectiva? ¿O mejor aún, que f sea no decreciente? ¿Es $\mathcal{F}$ equinumeroso a $2^{\mathbb{N}}$? (sin usar la hipótesis del continuo, no haga trampa).

Problema 4. El teorema fundamental de la aritmética puede ser formulado de la siguiente manera: Dado $n \in \mathbb{N}$, existe una única secuencia no decreciente y finita de números primos $p_1, p_2, ..., p_k$, tales que $n = p_1 \times p_2 \times ... \times p_k$. Ahora, sea $\mathbb{M}$ tal que cada elemento $m \in \mathbb{M}$ lo define una secuencia *infinita* de números primos. Demuestre que $\mathbb{M}$ no es numerable.

Problema 5. El set ternario de Cantor es un conjunto con propiedades interesantes. Está definido de la siguiente manera: Sea $C_0 = [0,1]$ y C_k la union de los intervalos que quedan al quitarle el tercio (abierto) de al medio a cada intervalo de C_{k-1}. Es decir:

- $C_0 = [0,1]$.
- $C_1 = [0, \frac{1}{3}] \cup [\frac{2}{3}, 1]$.
- $C_2 = [0, \frac{1}{9}] \cup [\frac{2}{9}, \frac{1}{3}] \cup [\frac{2}{3}, \frac{7}{9}] \cup [\frac{8}{9}, 1]$.

El set ternario de Cantor es $C = \bigcap_{k=0}^{\infty} C_k$. Demuestre que C no es numerable.

(**Hint:** piense en la representación infinita **en base 3** de los números en el intervalo [0,1]).

Problema 6. En clases definieron que una propiedad $P \subseteq N$ es decidible si existe un programa en Java (o en cualquier otro lenguaje en realidad) que recibe como entrada $n \in \mathbb{N}$ y si $n \in P$ retorna *si*, en caso contrario retorna *no*. Luego de eso se concluyó que habían propiedades no decidibles. ¿Cómo se demuestra esto usando numerabilidad?. Una demostración alternativa es entregando un lenguaje que efectivamente no sea decidible. Esto haremos ahora.

Sea $H \subseteq \mathbb{N}$ el lenguaje de todos los números que representan la codificación binaria de un programa en Java que compute lo siguiente:

1. Recibe un string p que representa un programa y un string x que representa un input.
2. Si el programa p con entrada x se detiene, retorna *true*, en caso contrario, retorna *false*

Demuestre que H no es decidible. [1]

[1] Esto es conocido como el *Halting problem* y sirve como base en varias demostraciones en el área de CCs. que estudia lenguajes recursivos.

Problema 7. Diagonalización no es la única forma de demostrar que $\mathbb{R}$ no es numerable. Usando $\mathbb{R} \sim 2^{\mathbb{N}}$ no nos sirve ya que $\mathbb{N} \prec 2^{\mathbb{N}}$ se demuestra con diagonalización. Lo que ahora se pide demostrar *sin diagonalización* que $\mathbb{R}$ no es numerable. Para eso puede ser útil un lema previo:

Lemma 1 *Si* $[a_1, b_1] \supseteq [a_2, b_2] \supseteq [a_3, b_3] \supseteq \ldots$ *es una secuencia de intervalos anidados cerrados, entonces* $\bigcap_{n=1}^{\infty} [a_n, b_n] \neq \emptyset$. [2]

La demostración de este lema queda como ejercicio.

[2] Es interesante notar que este lema no funciona si tomamos intervalos en $\mathbb{Q}$.