

# Teoría Matemática de la Computación

## Primer Problemario

Prof. Miguel A. Pizaña  
7 de junio de 2016

1. Dudar de todo, al menos una vez en la vida.
2. ¿Qué es la duda sistemática de Descartes? ¿Qué hay que hacer, según Descartes, para alcanzar el verdadero conocimiento?
3. Explique qué es un modelo y en qué es diferente de la realidad. En este contexto, explique el proceso de abstracción, deducción e interpretación.
4. ¿Cómo hace uno para medir qué tan bien aproxima un modelo matemático a la realidad? ¿Un modelo que no predice **exactamente** a la realidad debe ser descartado?
5. ¿Qué tan cierto es el teorema: ' $1+1=2$ '? Explique en detalle. Señale dos situaciones de la vida real en las que uno esperaría que el teorema fuera aplicable, pero de hecho no lo es. ¿Esto afecta la veracidad del teorema? ¿Cómo sabe uno cuándo puede uno aplicar el teorema y cuándo no?
6. ¿Qué tan cierto es el teorema de Pitágoras? Explique con detalle. Señale dos situaciones de la vida real en las que uno esperaría que el teorema fuera aplicable, pero de hecho no lo es. ¿Esto afecta la veracidad del teorema? ¿Cómo sabe uno cuándo puede uno aplicar el teorema y cuándo no?
7. ¿La tierra es plana? ¿verdad que no? ¿Entonces por qué usamos el "tiro parabólico" si ese modelo está basado en una tierra plana? ¿Es en realidad parabólico el movimiento de una pelota lanzada? ¿Qué efectos alejan a la pelota del movimiento parabólico? Mencione al menos 5.
8. ¿Qué es el efecto Magnus?

9. ¿Qué es el movimiento Browniano?
10. ¿Cuáles son las ideas centrales de la Teoría de la Relatividad General?
11. ¿Qué es la longitud de Planck? ¿Cuánto mide en metros?
12. ¿Qué dice el principio holográfico? ¿Qué tiene que ver con la tridimensionalidad del universo?
13. Dentro de un millón de años, una unidad de procesamiento quiere accesar una localidad de memoria en menos de un nanosegundo. ¿Cuál es la máxima distancia del procesador a la que puede estar físicamente ubicada tal localidad de memoria? (Suponiendo que las leyes de la física conocidas no hayan cambiado demasiado).
14. Demuestre que el universo no existe.
15. Suponga que lanza usted una moneda un millón de veces y en todas esas ocasiones usted ve a la moneda caer hasta el suelo en un lindo arco parabólico. ¿Eso es garantía de que la siguiente vez que usted lance la moneda, ésta no se va a convertir en un monstruo de espagueti volador? Explique.
16. Suponga que usted juega al *melate* un millón de veces y en todas esas ocasiones usted NO se saca el premio. ¿Eso es garantía de que la siguiente vez que usted juegue *melate* tampoco se va a sacar el premio? Explique.
17. ¿Cómo explica al diferencia en sus respuestas a los dos problemas anteriores, siendo que la evidencia experimental es idéntica?
18. Pruebe que el problema del paro es indecidible. ¿Cuáles son los supuestos usados durante el proceso de abstracción para llegar a este resultado?
19. Pruebe que si se agrega la restricción de que los programas corran con cierta memoria finita fija, entonces el problema del paro-con-memoria-finita, es decidible. ¿Este resultado teórico es útil en la práctica? Explique.

20. Según los dos modelos usados en los problemas anteriores es problema del paro es tanto decidible como indecidible. ¿Cuál es el consenso científico al respecto? ¿Cómo es el problema del paro *en la realidad*?
21. ¿Qué es un algoritmo de compresión? ¿Qué es un algoritmo de compresión con pérdida? ¿Qué es un algoritmo de compresión sin pérdida? ¿Cuándo se usan unos y otros?
22. Supongamos que  $A$  denota al conjunto de todos los archivos. Defina un algoritmo de compresión *chido* como una función que transforma archivos en archivos de tal manera que:
  - a)  $f : A \rightarrow A$  (es función)
  - b)  $f$  es inyectiva (sin pérdida)
  - c)  $|f(a)| \leq |a|$  para todo archivo  $a \in A$  y
  - d) Existe  $a_0 \in A$  tal que  $|f(a_0)| < |a_0|$ .

Demuestre que los algoritmos de compresión chidos no existen. ¿Cómo reconcilia usted este resultado con el hecho de que los algoritmos de compresión de hecho se usan todos los días en las computadoras?

23. ¿Un espejo invierte izquierda-derecha, arriba-abajo o adelante-atrás? Explique. ¿Por qué la mayoría de la gente tiene la idea equivocada al respecto?
24. ¿Qué es la libertad de cátedra? ¿la libertad académica? ¿neoliberalismo? ¿keynesianismo?
25. ¿Cuáles son los 3 principales modelos matemáticos que se usan para aproximar las nociones de “algoritmo” y “computadora”?
26. ¿Qué es una Máquina de Acceso Aleatorio Uniforme (uRAM)? ¿Qué es una Máquina de Acceso Aleatorio logarítmico (IRAM)? ¿Cómo se definen? ¿para que sirven? ¿Cuánto tiempo tardan en realizar un operación elemental? ¿Qué almacenan en cada celda de memoria? ¿cuántas celdas tienen? ¿Cuáles son las principales variaciones del modelo RAM básico? ¿Cuáles aspectos de la realidad aproximan bien estos modelos? ¿Cuáles aproxima mal? ¿Cuántos y cuáles son los supuestos falsos en los que se basan estos modelos?

27. Muestre que el algoritmo usual para multiplicar matrices tarda un tiempo del orden de  $n^3$  en el modelo uRAM.
28. Muestre que el algoritmo usual para inicializar matrices tarda un tiempo del orden de  $n^2$  en el modelo uRAM.
29. Muestre que el algoritmo usual para calcular el promedio de un arreglo tarda un tiempo del orden de  $n$  en el modelo uRAM.
30. Muestre que el algoritmo MergeSort tarda un tiempo del orden de  $n \log(n)$  en el modelo uRAM.
31. Muestre que el algoritmo Burbuja tarda un tiempo del orden de  $n^2$  en el modelo uRAM.
32. Si desea ordenar un arreglo de mil millones de enteros, cuanto tiempo esperaría tardarse usando: (1) Burbuja y (2) MergeSort. Exprese sus tiempos en la unidad adecuada (milisegundos, segundos, meses, años, etc.).
33. ¿Qué consideraciones tenemos que tener a la hora de medir tiempos de ejecución de algoritmos? ¿Cuáles son las posibles fuentes de error o imprecisión al hacer mediciones? ¿qué métodos o funciones podemos usar para medir el tiempo de ejecución de un programa en nanosegundos?
34. El modelo uRAM predice un tiempo lineal (proporcional al tamaño del arreglo) para el algoritmo que calcula promedios de arreglos. ¿Qué tan bien se lleva esta predicción con el experimento? Explique.
35. El modelo uRAM predice que el tiempo para inicializar un arreglo por renglones es idéntico al tiempo necesarios para inicializarlo por columnas. ¿Qué se observa en el experimento? ¿Cómo explica la discrepancia?
36. ¿Qué dice el principio de localidad? ¿Qué son la jerarquía de memoria, la memoria caché, una página, un marco, el principio de localidad y la hiperpaginación? ¿De qué manera afectan estas nociones a las predicciones del modelo uRAM y a los correspondientes experimentos de medición de tiempos de ejecución?
37. Mencione al menos 3 razones por las que no se utilizan los mismos microchips para las sondas de la NASA que para las computadoras personales.

38. ¿Cuántos bits usa su lenguaje favorito para almacenar los tipos de datos: char, int, long int y long long int?
39. ¿Qué tan distintas pueden ser las predicciones del modelo uRAM y el modelo lRAM? Muestre un ejemplo en donde las predicciones de un modelo y otro sean muy diferentes.
40. ¿En qué condiciones es mejor usar el modelo uRAM y en qué condiciones es mejor usar el modelo lRAM? ¿Cuándo ambos modelos dan predicciones asintóticamente idénticas?
41. ¿Un 'int' es un entero? El código: `x=1;while(x++);` para o se encicla? Explique en detalle.