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

IIC1001 — Algoritmos y Sistemas Computacionales — 2023-1

Interrogación 2

Lunes 5-Junio-2023

Duración: 120 minutos
Responder cada pregunta en una hoja separada.

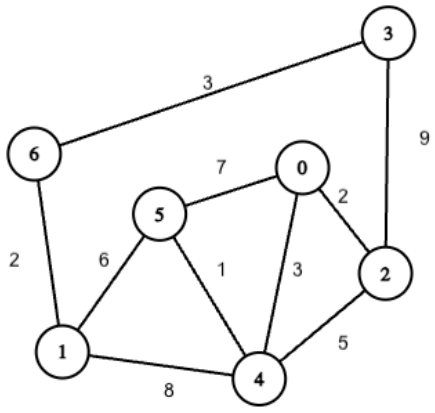
1. **[12p]** A partir del siguiente conjunto de *strings*:
   $\{\texttt{Jorge}, \texttt{Valeria}, \texttt{ignacio}, \texttt{1203}, \texttt{Catalina}, \texttt{cristian}, \texttt{Francisca}, \texttt{kamilo}\}$

   1.1) **[4p]** Construya una tabla de hash donde la función de hash sea la longitud de cada string, y resolviendo colisiones mediante una lista ligada.

   1.2) **[4p]** Suponga que tiene un conjunto de N *strings*, donde N puede ser muy grande. Ningún *string* es de largo mayor a K. ¿Cuál es la complejidad, en el peor caso, de buscar un elemento en esta tabla?

   1.3) **[4p]** Proponga una función de hash para que NO se provoquen colisiones con el caso particular del conjunto de *strings* propuesto (el conjunto de ejemplo).

2. **[12p]** A partir del siguiente grafo no dirigido, $G = (V, E)$:



   2.1) **[4p]** Obtener un recorrido BFS a partir del nodo 0

   2.2) **[4p]** Utilice el algoritmo de Dijkstra para obtener las rutas más cortas desde 0 hasta los demás. Debe mostar el camino desde 0 hacia cada uno de los otros nodos, y el costo de la ruta.

   2.3) **[4p]** Ejecute el siguiente algoritmo sobre el grafo G para construir un nuevo grafo $G' = (V', E')$: (1) tomar el mismo conjunto de nodos $V' = V$; (2) Agregar a E' la arista de E con **menor** peso, y que **no forme un ciclo en** G'. El algoritmo termina cuando no hay más aristas que agregar. Dibuje el resultado e indique su complejidad.

3. **[13p]** Representaciones y operaciones númericas:

   3.1) **[9p]** Complete la siguiente tabla usando conversiones numéricas (no detalle el procedimiento):

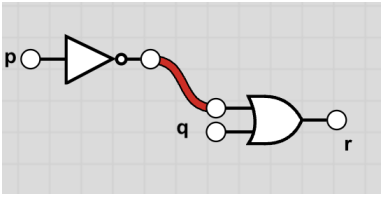
decimal	binario	hexadecimal
1200		
2048		
3072		
	0010 0001	
	0101 1010	
	0001 1001 1100	
		AC
		E7
		288

   3.2) **[4p]** Efectúe las siguientes operaciones de números hexadecimales explicitando su procedimiento. Puede hacerlo en notación hexadecimal o binaria, pero **no es válido** convertirlas a decimal (salvo si quiere comprobar un resultado):

   a) `0x23` + `0x0E`
   b) `0x79` − `0x64`
   c) `0x513D` + `0x8`
   d) `0x50FD` − `0x503C`

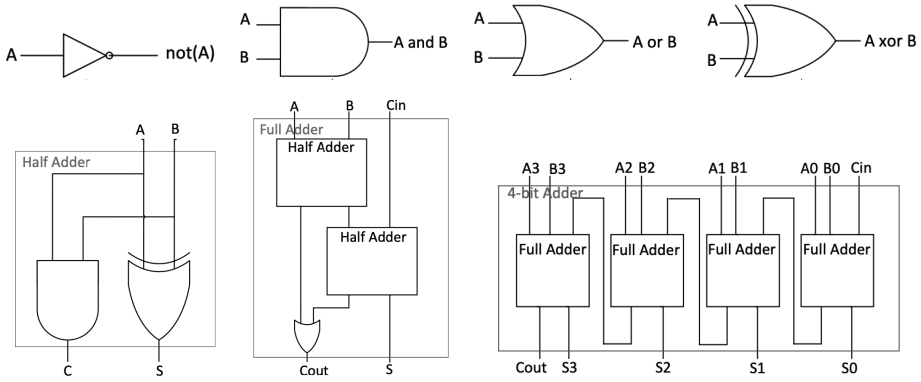
4. **[12p]** Para las siguientes construcciones con celdas lógicas:

   4.1) **[4p]** Construir una tabla con todas las salidas posibles.



   Bonus (1pto): ¿qué valor lógico representa esto?

   4.2) **[8p]** Ejecute la suma de los valores hexadecimales: $0xC$ y $0x6$ usando un full-adder de 4 bit. Muestra la entrada y salida del full-adder. Como referencia se agregan los diagramas vistos en clases.



5. **[11p]** Una manera de codificar secuencias de dígitos mediante bit es la siguiente: la secuencia empieza y termina con un bit 0; cada dígito d se representa mediante d bit 1 seguidos; y para separar dos dígitos consecutivos se utiliza un bit 0. Por ejemplo, la secuencia 242 se representa como 011011110110. Se empieza con un 0, a

continuación 2 bit en 1, luego un bit 0 como separación, a continuación 4 bit en 1, un bit 0 en como separación, 2 bit en 1, y finalmente termina con 0. El dígito 0 se representa con 10 bit 1 seguidos.

Para que esta secuencia se pueda escribir en Byte, hay que completar con 0's a la izquierda hasta que la cantidad de bit sea un múltiplo de 8. En el ejemplo anterior, que tiene 12 bit, se le agregan 4 bit 0 a la izquierda. El resultado queda 0000011011110110. Esto se puede escribir con 2 Byte.

Construya un archivo con el siguiente formato:

- **[3p]** 8 Byte, donde cada byte contiene el **caracter** ASCII correspondiente a la fecha de hoy en formato `YYYYMMDD`. Atención que son dos caracteres para el mes y dos para el día.
- **[1p]** 1 Byte con el valor numérico 201 si usted está en la sala AE201, o el valor 202 si usted está en la sala AE202.
- **[2p]** 1 Byte con el valor numérico que representa la cantidad de Byte (el valor X) usados en la siguiente parte.
- **[4p]** X Byte con la codificación del número 16251830 de acuerdo a lo explicado.
- **[1p]** El archivo debe quedar con 16 bytes en cada línea. Si es necesario debe agregar el byte correspondiente al valor numérico 255 al final, y repetirlo hasta completar una línea de 16 byte.