

## Evaluación Final Virtual Octubre 29 /10/ 2020

### Ejercicio 1

java complejidad computacional

Se tiene un arreglo con  $N$  datos representados por valores enteros (sin repeticiones), con  $3 \leq N \leq 100.000.000$ .

Los datos pueden estar corruptos, los cuales se pueden verificar utilizando la siguiente función que ya se encuentra disponible, la cual indica si el dato recibido como parámetro es corrupto o no:

**public boolean esCorrupto(int dato)**

Lo que se sabe, es que si el elemento de la posición  $i$  es corrupto, todos los elementos siguientes también lo son.

Se pide escribir un algoritmo que retorne la posición del primer dato corrupto del arreglo:

- a) Con complejidad  $O(n)$
- b) Con complejidad  $O(\log(n))$

Un caso de ejemplo podría ser:

F	F	F	F	F	F	F	F	T	T	T	T	T	T
0	1	2	3	4	5	6	7	8	9	10	11	12	13

En donde la respuesta debería ser: 8.

---

### Ejercicio 2

prolog

#### Parejas de baile

Un salón de baile desea organizar un concurso, para esto cuenta con un fichero con información de personas dispuestas a bailar. Las parejas se formarán según sus compatibilidades de acuerdo a las declaraciones siguientes:

hombre ( $n1, b1, b2$ ).

mujer ( $n2, b1, b2$ ).

donde  $n1$  es el nombre de un hombre o una mujer,  $b1$  primer estilo de baile en que le gustaría participar y  $b2$  el segundo estilo de baile de su agrado.

Considere que dos personas  $X$  e  $Y$  de sexos diferentes son adecuadas para formar pareja si  $X$  e  $Y$  tienen en común alguno de los dos estilos de baile que hayan declarado.

Se desea obtener el listado de posibles participantes, indicando el nombre de ambos y el estilo en el que participan. Codifique las reglas prolog, que permitan obtener dicho listado.

Ejemplo de declaración:

hombre(alberto, cumbia, salsa).

mujer(ana, salsa, merengue).

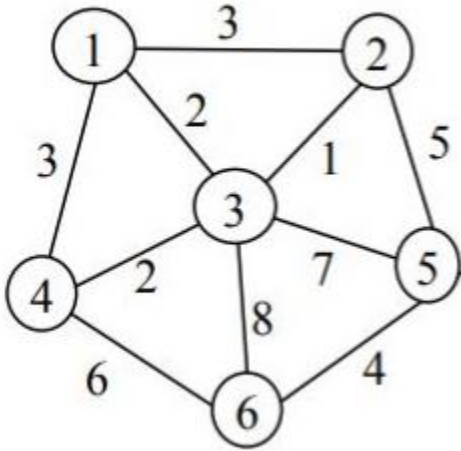
Ana y Alberto podrían participar del estilo salsa.

---

## Ejercicio 3

kruskal prim grafos

Aplicar el algoritmo de Kruskal sobre el siguiente grafo, mostrando su ejecución paso a paso, mostrando cómo se controla que no se cierren ciclos. Indicar la complejidad computacional.



Si aplicamos el algoritmo de Prim al mismo grafo, ¿podemos asegurar que se obtendría siempre la misma solución? ¿Podemos asegurar que el coste de la solución sería el mismo? ¿Por qué?