

+20

### Organización de Datos (75.06) Primer Cuatrimestre de 2014. Examen parcial, primera oportunidad. [2014\_1c\_Parcial\_1]

Importante: Antes de empezar complete nombre y padrón en el recuadro. Lea bien todo el enunciado antes de empezar. Para aprobar se requiere un mínimo de 60 puntos (60 puntos = 4) con al menos 15 puntos entre los ejercicios 1 y 2. Este enunciado debe ser entregado junto con el parcial si quiere una copia del mismo puede bajarla del grupo de la materia. En el ejercicio 3 elija 2 de los 4 ejercicios y resuelva única y exclusivamente 2 ejercicios. Si tiene dudas o consultas levante la mano, está prohibido hablar desde el lugar, fumar o cualquier actividad que pueda molestar a los demás. El criterio de corrección de este examen estará disponible en forma pública en el grupo de la materia.

"The question of whether a computer can think is no more interesting than the question of whether a submarine can swim." (E. Dijkstra)

#	1	2	3.1	3.2	4	5	6	7	Entrega Hojas:
Corr	B	B-	A	D	B	-	B		Total:
Puntos	15/15	10/15	10/10	10/10	15/15	-	10/10	10/10	80 /100

Nombre: DARIO L. MINONES  
Padrón: 86644  
Corregido por: MELALOS HERNAN

+20 = 100

1) Se tiene un archivo con información sobre visitas a páginas web de la forma: (URL, visitas, fecha). Existe solo un registro por día para cada URL. Se quiere generar un archivo que, por cada URL, indique cuál fue la fecha en la que tuvo más visitas y la cantidad de visitas.

Programar lo pedido en Map Reduce usando agregación para minimizar la cantidad de datos que deben transferirse en la red.

Atención: La resolución es muy simple, trivial, así que es fundamental resolver la agregación para el puntaje completo. (\*\*\*) (15 pts)

2) Se tiene un archivo que representa una red social con billones de links entre usuarios de la forma (id1,id2). Se quieren encontrar todos los triángulos de tipo (id1,id2,id3) tales que existen (id1,id2),(id2,id3),(id1,id3). Explicar, sin programar, qué algoritmo usaría para listar todos los triángulos que existen en la red social usando Map Reduce. En concreto se pide para el siguiente fragmento del archivo:

(A,B) (B,C) (A,C) (A,D) (C,D) (D,E) (D,F) (E,F) (A,E)

- Qué recibe el método Map, qué hace y qué emite.
- Qué recibe el método Reduce, qué hace y qué emite.

Si lo hace en dos o más iteraciones entonces explique los puntos anteriores por cada iteración. (\*\*\*\*\*) (15pts)

(opcional x puntos extras) ¿De qué forma podría usarse el contar la cantidad de triángulos para saber si un determinado grafo es o no una red social? Justifíquelo. (\*\*\*\*\*) (5 pts)

3) Resolver 2 (dos) y solo 2 de los siguientes ejercicios a elección (si resuelve más de 2 el ejercicio vale 0 puntos, sin excepciones). En cada caso indicar V o F justificando adecuadamente sus respuestas. Si no justifica vale 0 puntos sin excepciones.

a) En una colección de 1300 documentos el b óptimo para un término que aparece en 900 de ellos es 1. (\*\*) (10 pts)

b) Si para una cierta consulta q d1 es más relevante que d2 entonces luego de aplicar LSI d1 seguirá siendo más relevante que d2. (\*\*\* ) (10 pts)

c) Si la probabilidad de las distancias 8 a 15 es similar a 1/256 entonces es buena idea representarlas usando código Delta. (\*\*) (10 pts)

d) En los casos en los que hay más términos que documentos el código unario es óptimo. (\*) (10 pts)

4) Comprimir con PPMC de orden máximo 2 (dos) el siguiente archivo: ABABAA

Indicar en cada paso el estado de los modelos y el archivo final comprimido en binario. Importante: Si deja expresado el archivo final como las probabilidades vale 0 puntos (sin excepciones) (\*\*\*\*\*) (15 pts)

5) Se quiere aplicar LSH a un conjunto de documentos para encontrar los pares de documentos más similares. Queremos que si  $J(D1,D2) >= 0.7$  entonces la probabilidad de que D1 y D2 sean candidatos sea  $>= 0.9$  y queremos que si  $J(D1,D2) <= 0.5$  entonces la probabilidad de que sean candidatos sea  $<= 0.3$ . Indique cuántas funciones minhash usaría y qué combinación de AND y OR usaría para lograr lo pedido. (\*\*\*\*\*) (15 pts)

6) Si tenemos más de un dispositivo físico disponible (discos) para hacer el sort externo de un archivo de varios gigabytes de longitud. ¿Qué algoritmo usaría para realizar el sort? Justifique. (\*) (10 pts)

7) Encontrar el m mínimo y los parámetros a y b de forma tal que la función de hashing  $ax + b \bmod m$  sea perfecta para las siguientes claves: 1,3,5,12 (\*\*\*\*\*) (10 pts)

Minibar 86644

2 cont.2ds Stes con

R

Mop es la cantidad ~~que se producen~~<sup>de</sup> para

Reducir tanto los roles que tienen por cada  
diseño y si es 3 tiene un triángulo.

Esta expresión podría basarse en combinatoria.

Este método tiene el problema de generar mucha  
información adicional ya que genera muchos diseños  
duplicados en los mapeos la información y  
genera muchos falsos candidatos. ✓

Preguntas Extra

La cantidad de triángulos en un gráfico es un  
índicador de que tan interrelacionados están los  
modos.

Si pude ~~contar~~ ~~calcular~~ encontrar  
un cuento cont. # de triángulos/modos.

Este cuento cuando más alto nos solucionados  
están. Podrás definirte un umbral que indique  
si es o no un red social.

Ejercicio 3

a) Falso

Contreexample:

$$pt = \frac{902}{1300} = 0,69 \quad \text{Como es mayor a } 0,5 \text{ indexos no ocurren.}$$

B)

Por lo tanto  $1 - 0,69 = 0,31 \leq 0,38$

0,38 es el umbral de  $b=1$

d) Falso

Contreexample:

Unico es óptimo si  $b$  es uno. pero lo usual  $p > 0,38$

Contreexample:

Si suponemos por ejemplo 100 términos en 100 documentos donde cada término esté en 90 de estos documentos  $9 \cdot 10^3$  de estos documentos.

$$p = \frac{900 \cdot 10^6}{1000 \cdot 10^6} = 0.9$$

Como 0.9 es menor a 0.5  $\Rightarrow$  indexos no ocurren.

$$1 - 0.9 = 0.1 \leq 0.38$$

4)

	M-1	M0	M1	M2
PASO 1	A 1/3	ESC 1	AB 1/2	
CTX: NULO	B 1/3			
LETRA: A	EOF 1/3			

EMITE: ESC (1) ✓ A (1/3) ✓

	M-1	M0	M1	M2
PASO 2	A 1/3	ESC 1/2	A] ESC 1	
CTX: A	B 1/3	A 1/2		
LETRA: B	EOF 1/3			

EMITE: ESC (1) ✓ ESC (1/2) ✓ B (1/3) ✓

	M-1	M0	M1	M2
PASO 3	A 1/3	ESC 1/4	A] ESC 1/2	AB] ESC (1)
CTX: AB	B 1/3	A 1/4	B 1/2	
L : A	EOF 1/3	B 1/4	B] ESC 1	

EMITE: ESC (1) ✓ ESC (2) ✓ A (1/4) ✓

	M-1	M0	M1	M2
PASO 4	A 1/3	ESC 2/5	A] ESC 1/2	
CTX: BA	B 1/3	A 2/5	B 1/2	AB] ESC 1/2
L : B	EOF 1/3	B 2/5	B] ESC 1/2	A 1/2

EMITE: ESC (1) ✓ B (1/2) ✓

	M-1	M0	M1	M2
PASO 5	A 1/3	ESC 2/5	A] ESC 1/3	AB] ESC 1/2
CTX: AB	B 1/3	A 2/5	B 2/3	A 1/2
L : A	EOF 1/3	B 2/5	B] ESC 1/2	B 1/2

EMITE: A (1/2)

	M-1	M0	M1	M2
PASO 6	A 1/3	ESC 2/3	A) ESC 1/3 B) 2/3	AB) ESC 1/3
CTX: BA	B 1/3	A 3/3	B) ESC 1/3	A 2/3
L : A	EOF 1/3	B 2/3	A 2/3	B 1/2

EMITE : ESC(1/2) ESC(1) A(B)

	M-1	M0	M1	M2
PASO 7	A 1/3	ESC 2/3	A) ESC 2/5 B) 2/5	AB) ESC 1/3
CTX: AA	B 1/3	A 4/3	B) ESC 1/3	A 2/3
L : EOF	EOF 1/3	B 2/3	A 2/3	BA) ESC 2/4

EMITE : ESC(1) ESC(2/5) ESC(1) EOF(1) AA) esc 1

Calcula el buenisimo

A: ESC(1) A(1/3)

B: ESC(1) ESC(1/2) B(1/3)

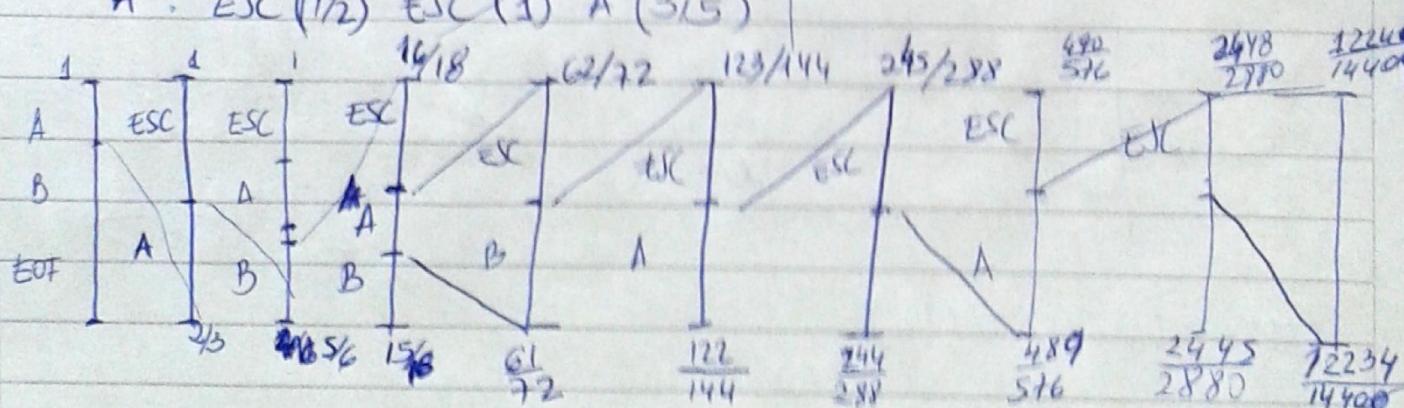
A: ESC(1) ESC(1) A(1/4)

B: ESC(1) B(1/2)

A: A(1/2)

A : ESC(1/2) ESC(1) A (3/5)

EOF : ESC(1) ESC(2/5) ESC(1) EOF(1)



$$\frac{12234}{14400} = 0.849583$$

$$\frac{12240}{14400} = 0.85$$

B

1 1 0 1 1 0 0 1 1

es el menor buenisimo al atento

NOTA

~~6) Usando método de bloques y merge de cada bloque~~

7)

Con  $m=5$  y  $a=3$  y  $b=3$

$$1 \cdot 3 + 3 = 6 \bmod 5 = 1$$

$$3 \cdot 3 + 3 = 12 \bmod 5 = 2$$

$$5 \cdot 3 + 3 = 18 \bmod 5 = 3$$

$$12 \cdot 3 + 3 = 39 \bmod 5 = 4$$

El  $m$  es el mínimo posible y  $a$  y  $b$  cumplen todos los criterios de una fórmula de hashing pedido.

Matriz de bloques

6) Utilizando método solución en bloques y un merge para consolidar.

De este manera puede operarse con distintos datos físicos por igualando posiciones entre si y sus ordenadas.

También al hacer el merge se opera de tener zonas distintas que pueden leerse y escribirse o lo que sea sin perturbar las distintas ~~posiciones~~ ~~posiciones~~ de los datos físicos.

2) a

En su primera ejecución el mapeo devolverá el par y su pertenencia.

El redudir se:

Solido Map:

$(AB)$   $(BA)$   $(BC)$   $(CB)$   $(AC)$   $(CA)$   $(AD)$   $(DA)$   $(CD)$   $(DC)$   
 $(DE)$   $(ED)$   $(DF)$   $(FD)$   $(EF)$   $(FE)$   $(AE)$   $(EA)$

El reducir recibe para cada letra todos los que se relacionan y crea un candidato ordenando los id's.

Entrada Reducción

$(A, [B, C, D, E])$	$ $	$(D, [A \subset E F])$
$(B, [A, C])$	$ $	$(E, [D, F, A])$
$(C, [A B D])$		

Sólido A:  $((A, B, C), 1)$   $((A \subset D), 1)$   $((A B D, 1))$   $((A B E), 1)$   
 $((A B D), 1)$   $((A C E, 1))$   $((A D E), 1))$

B:  $((A B C), 1)$

C:  $((A B C), 1)$   $((A, C, D), 1)$

Luego un segundo proceso de Map Reduce devolverá como resultado repetición de los dos candidatos y si son 3 hay un triángulo

Sigue →

Misiones 86644

HOJA N° 1

FECHA

1) Forma de (URL, visitas, fecha)

Salida : (URL, (visitas, fecha))

Mapper

✓ map (regid, reg)

emit (reg.URL, (reg.visitas, reg.fecha))

No acarrea los  
keys que  
usa ...

Combiner

combine (key, values)

Max visitors  
value = 0

else

✓ var max = 0, {visits: 0, fecha: 0}

foreach value in values

if (max.visits < value.visits)  
max = value

emit (key, (max.visits, max.fecha))

Reducir

reduce (key, values)

var max = 0, {visits: 0, fecha: 0}

foreach value in values

if (max.visits < value.visits)  
max = value

emit (key, (max.visitas, max.fecha))

NOTA