

QUESTION #1

"Best one" numero 15 poiché condinche il maggior numero di nodi con 16

	Out degree	Ref	Copy List	Extra Nodes
15	13			
16	13	1	0011111110110	9, 20, 21, 30



blocks = 5,

Copy Blocks = 0, 1, 6, 0, 1

QUESTION #2

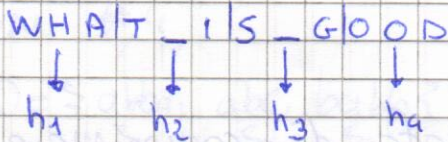
F_{old} = "WHAT_IS_GOOD"

F_{new} = "WHAT_IS_SO_GOOD"

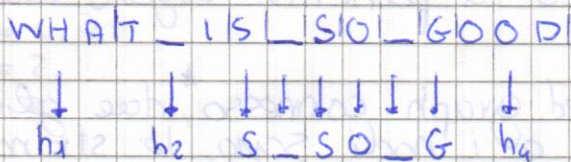
B = 3

RSync

Si presuppone una bassa capacità computazionale del client.
Il client invia al server un SET di hash ottenuti nel seguente modo:



Il server utilizzerà rolling hash per dedurre F_{old}, o meglio uno sketch di esso:

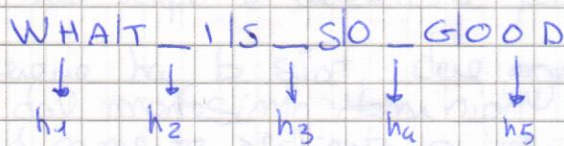


Il server invierà quindi al client gzip(h₁, h₂, S, SO, G, h₄), dove h₁, h₂, h₄ sono simboli speciali.

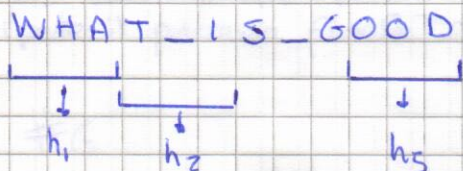
Triple emesse: <0, 0, h₁> <0, 0, h₂> <0, 0, S> <0, 0, _> <2, 1, 0> <3, 1, G> <0, 0, h₄>

ZSync

Si presuppone una buona capacità computazionale del client. Il server invia al client una sequenza di hash ottenuti nel seguente modo:



Il client utilizza rolling hash per massimizzare il numero di match.



Il client invia quindi al server il bit-array 11001

Il server comprende quindi che il client possiede i blocchi WHA, T-I e OOD, mentre S-S e O-G sono mancanti. Il server invia al client: `gzip(WHAT_I OOD | S_S O_G)`

Triple: $\langle 0, 0, 5 \rangle \langle 6, 1, 5 \rangle \langle 5, 1, _ \rangle \langle 0, 0, 6 \rangle$

Il client decodifica quindi tali triple utilizzando il prefisso WHA T-I OOD, il quale corrisponde infatti agli hash noti h_1 , h_2 e h_5 e ottiene S-S O-G.

Poiché tale stringa era stata compressa mantenendo l'ordinamento degli hash non noti h_3 e h_4 , il client è in grado di comprendere che $h_3 = \text{hash}(S-S)$, $h_4 = \text{hash}(O-G)$ e quindi può ricostruire F-new.

QUESTION #3

$S = \{s_1, s_2, s_3, s_4\}$
S = {abaco, basco, raco, vasto}

Z-delta valuta quale è il miglior modo di comprimere tali stringhe valutando i costi (# triple) emesse dalle varie compressioni $gzip(\{s_i | S_j\}) \forall i, j \neq j$

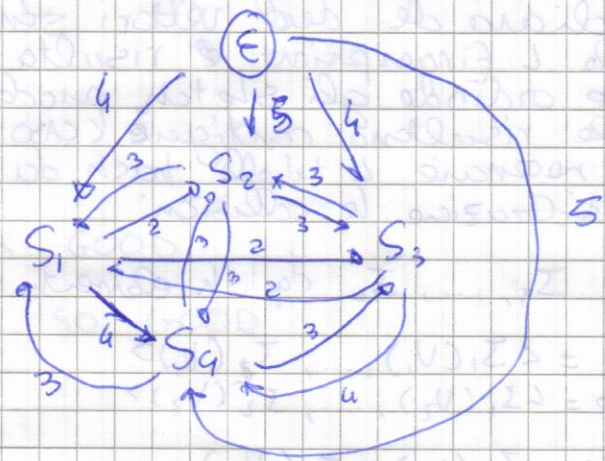
~~Per identificare il nodo di partenza vengono valutate anche $\forall i \text{ gzip}(s_i)$~~

Viene costruito un weighted directed graph connesso*, dove gli edge vengono associati coi relativi costi e i nodi sono le stringhe, al quale viene aggiunta una source fittizia per derivare la stringa di partenza.

Viene poi risolto un problema simile al minimum spanning tree per ottenere il miglior ordinamento tra le stringhe, ovvero quello che porta a una miglior compressione.

* vi è un arco tra ogni due nodi

	COSTO
$g_{zip}(S_1) = g_{zip}(abaco) = \langle 0,0,a \rangle \langle 0,0,b \rangle \langle 2,1,c \rangle \langle 0,0,o \rangle$	4
$g_{zip}(S_2) = g_{zip}(basco) = \langle 0,0,b \rangle \langle 0,0,a \rangle \langle 0,0,s \rangle \langle 0,0,c \rangle \langle 0,0,o \rangle$	5
$g_{zip}(S_3) = g_{zip}(raco) = \langle 0,0,r \rangle \langle 0,0,a \rangle \langle 0,0,c \rangle \langle 0,0,o \rangle$	4
$g_{zip}(S_4) = g_{zip}(vasto) = \langle 0,0,v \rangle \langle 0,0,u \rangle \langle 0,0,s \rangle \langle 0,0,t \rangle \langle 0,0,o \rangle$	5
$g_{zip}(S_1 S_2) = g_{zip}(abaco basco) = \langle 4,2,s \rangle \langle 5,2,EOF \rangle$	2
$g_{zip}(S_1 S_3) = g_{zip}(abaco raco) = \langle 0,0,r \rangle \langle 4,3,EOF \rangle$	2
$g_{zip}(S_1 S_4) = g_{zip}(abaco vasto) = \langle 0,0,v \rangle \langle 4,1,s \rangle \langle 0,0,t \rangle \langle 5,1,EOF \rangle$	4
$g_{zip}(S_2 S_1) = g_{zip}(basco abaco) = \langle 4,1,b \rangle \langle 6,1,c \rangle \langle 5,1,EOF \rangle$	3
$g_{zip}(S_2 S_3) = g_{zip}(basco raco) = \langle 0,0,r \rangle \langle 5,1,c \rangle \langle 4,1,EOF \rangle$	3
$g_{zip}(S_2 S_4) = g_{zip}(basco vasto) = \langle 0,0,v \rangle \langle 5,2,t \rangle \langle 5,1,EOF \rangle$	3
$g_{zip}(S_3 S_1) = g_{zip}(raco abaco) = \langle 3,1,b \rangle \langle 5,3,EOF \rangle$	2
$g_{zip}(S_3 S_2) = g_{zip}(raco basco) = \langle 0,0,b \rangle \langle 4,1,s \rangle \langle 5,2,EOF \rangle$	3
$g_{zip}(S_3 S_4) = g_{zip}(raco vasto) = \langle 0,0,v \rangle \langle 4,1,s \rangle \langle 0,0,t \rangle \langle 5,1,EOF \rangle$	4
$g_{zip}(S_4 S_1) = g_{zip}(vasto abaco) = \langle 4,1,b \rangle \langle 6,1,c \rangle \langle 5,1,EOF \rangle$	3
$g_{zip}(S_4 S_2) = g_{zip}(vasto basco) = \langle 0,0,b \rangle \langle 5,2,c \rangle \langle 5,1,EOF \rangle$	3
$g_{zip}(S_4 S_3) = g_{zip}(vasto raco) = \langle 0,0,r \rangle \langle 5,1,c \rangle \langle 4,1,EOF \rangle$	3



Viewe iniato: $E \xrightarrow{4} S_3 \xrightarrow{2} S_1 \xrightarrow{3} S_2 \xrightarrow{3} S_4$ costo 11
 $g_{zip}(S_3) g_{zip}(S_3 | S_1) g_{zip}(S_1 | S_2) g_{zip}(S_2 | S_4)$

QUESTION #4

Per $D = \{ abba, abc, baab \}$ $k=2$

- $fa \rightarrow 1, 2$
- $ab \rightarrow 1, 2, 3$
- $bb \rightarrow 1, 3$
- $ba \rightarrow 1, 3$
- $bc \rightarrow 2$
- $fb \rightarrow 3$

$Q = abcc \rightarrow$ $fa \rightarrow 1, 2$ overlap simu = 2 con S_1 , 3 con S_2 , 1 con S_3
 $ab \rightarrow 1, 2, 3$
 $bc \rightarrow 2$
 $cc \rightarrow /$
 $e=1 \Rightarrow |Q| - ke = 4 - 2 = 2$
 \Rightarrow Richiesti almeno 2 bigrammi condizi
 \Rightarrow Candidati: S_1, S_2