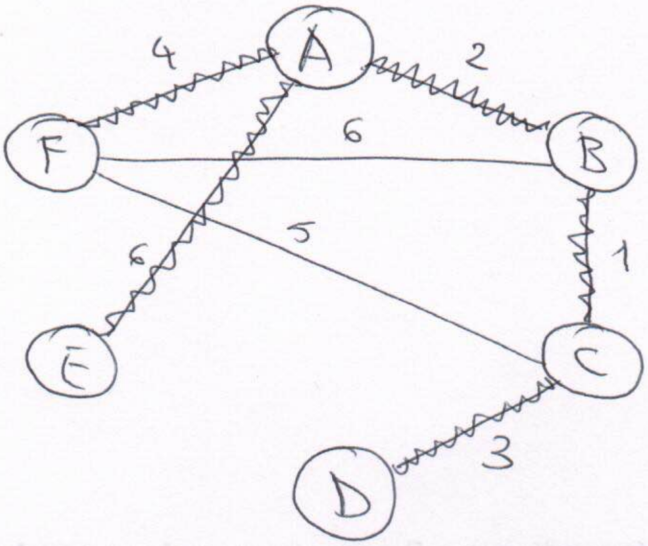
Exercise 1



~~~ MST edge

PRIM's Algorithm uses a priority queue indexed by the minimum-cost edge incident in a node. Everything starts from the root A.

ops	A	B	C	D	E	F
	0_A	∞	∞	∞	∞	∞
Extract A	•	2_A	∞	∞	6_A	4_A
Extract B (A,B)		•	1_B	∞	6_A	4_A
Extract C (B,C)			•	3_C	6_A	4_A
Extract D (C,D)				•	6_A	4_A
Extract F (A, F)					6_A	•
Extract E (A,E)						

If $n < M$ the PQ fits in internal memory and every node extraction takes 0 I/Os but the update of the PQ needs to scan on disk (possibly) the adjacency list of u $\lceil \frac{adj(u)}{B} \rceil$ I/Os. If we sum over all nodes u, it is $|V| + |E|/B$ I/Os
~~~

Exercise 2

1.

		L
amaca$		$amaca
maca$a		a$amac
aca$am	sort $\longrightarrow$	aca$am
ca$ama		amaca$
a$amac		ca$ama
$amaca		maca$a

MTF(acmaa) with $'s position = 4

$\mathcal{L} = \{ \overset{1}{a}, \overset{2}{c}, \overset{3}{m} \}$

a c m a a

↓ ↓ ↓ ↓ ↓

1 2 3 3 1

$\mathcal{L} = \{ \overset{1}{c}, \overset{2}{a}, \overset{3}{m} \}$

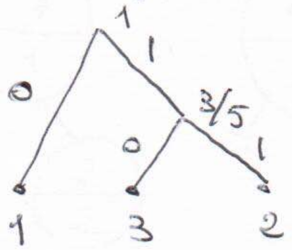
$\mathcal{L} = \{ m, c, a \}$

$\mathcal{L} = \{ a, m, c \}$

RLE1(1 2 3 3 1) $\longrightarrow$ 1 2 3 3 1

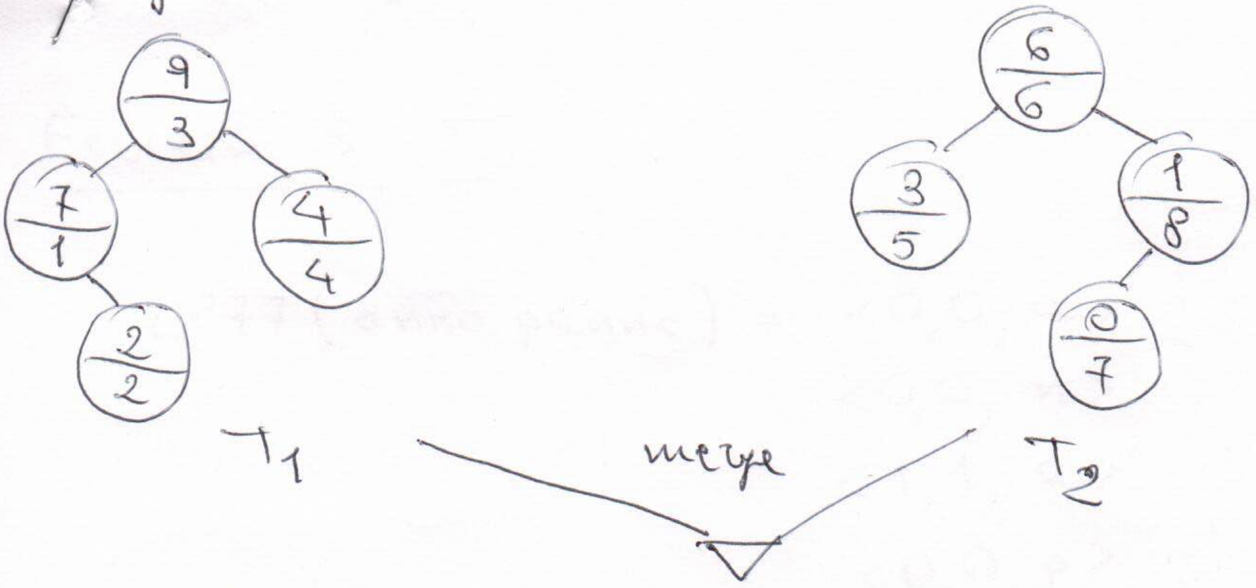
Huffman(1 2 3 3 1) $\Longrightarrow$

		cw
$f(1) = 2/5$	$\longrightarrow$	0
$f(2) = 1/5$	$\longrightarrow$	10
$f(3) = 2/5$	$\longrightarrow$	11

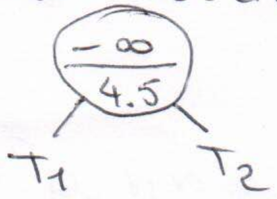


Exercise 3

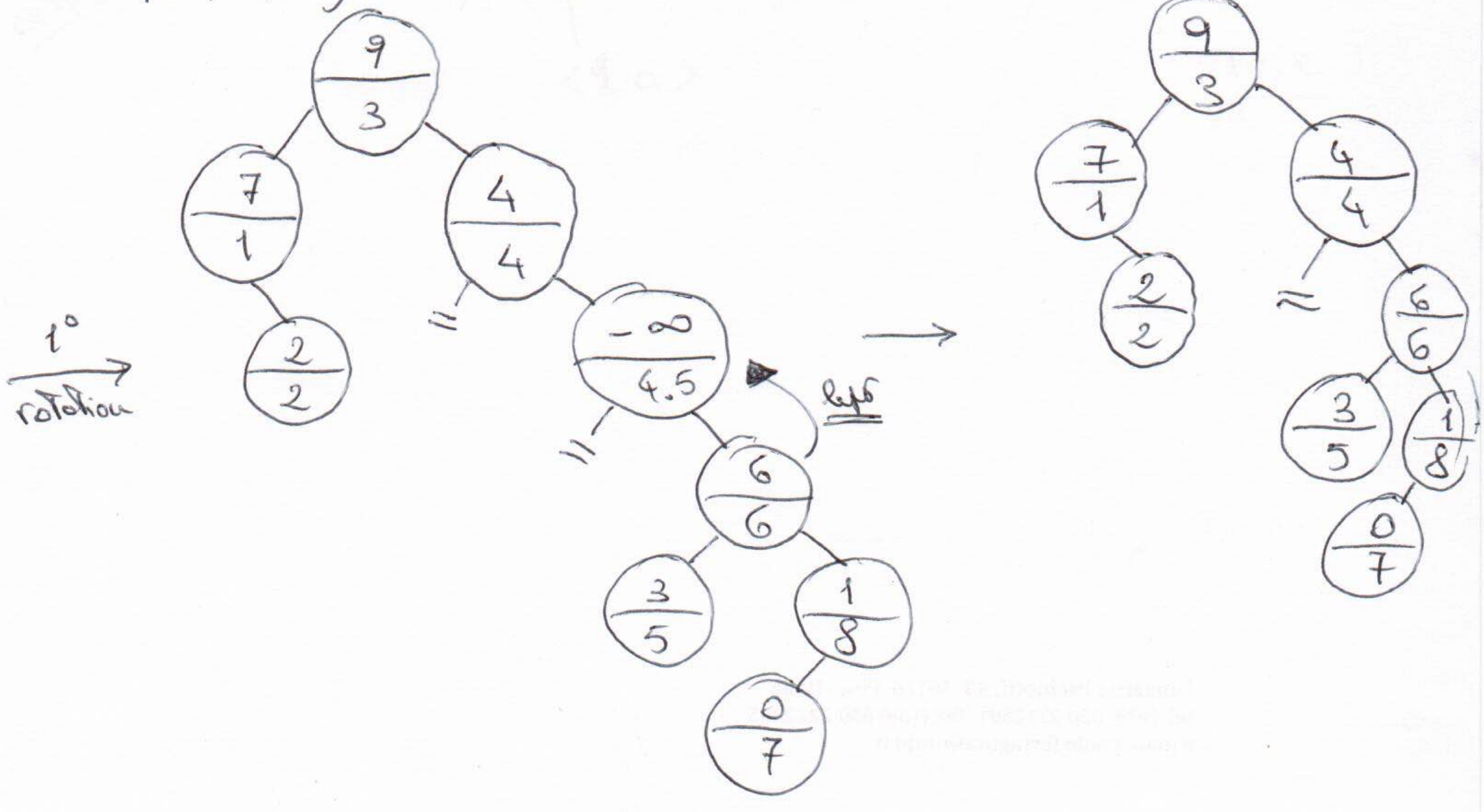
they are maximum heaps and so:



we create a dummy root with priority $-\infty$ and value between $\max(T_1) = 4$ and $\min(T_2) = 5$ so that the treap



is correct in terms of key distribution, that means it is a binary search tree. Then we make the dummy root percolate down to reestablish also the heap-property on the priorities based on rotations.



Exercise 4

see notes

Exercise 5

$LZ77(\text{annepenne}) = \langle 0,0,a \rangle$ — substring: a

$\langle 0,0,n \rangle$ — n

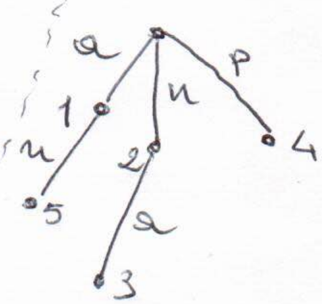
$\langle 1,1,a \rangle$ — na

$\langle 0,0,p \rangle$ — p

$\langle 5,4,EOF \rangle$ — anna

LZ78 (a n n a p a n n a)

output: $\langle 0,a \rangle$ $\langle 0,n \rangle$ $\langle 2,a \rangle$ $\langle 0,p \rangle$ $\langle 1,n \rangle$ $\langle 3,EOF \rangle$



Trie