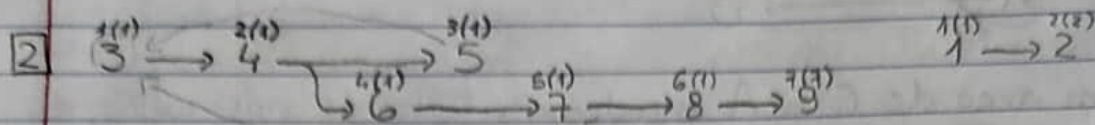
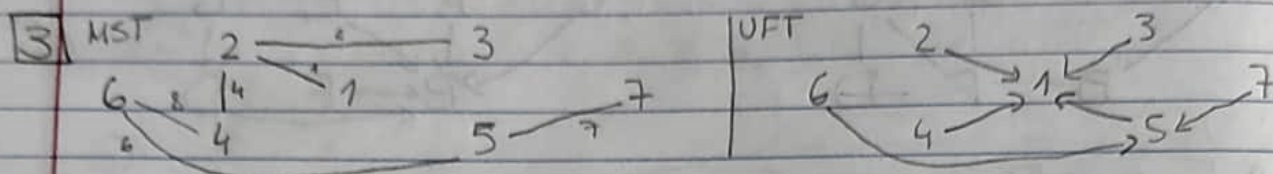


ER23

1) $a=1$ $b=2$ $c=1$ $d=1$ $\log_2 1 < 1 \rightarrow O(n)$ 

SCC: {3,4,5,6,7,8}, {9}, {1}, {2}

4) CB: $n=1 \rightarrow \text{mean}(L, 1) = L[1]$

H.I: se, para $n=k-1$, $\text{mean}(L, k-1) = \frac{L[1] + \dots + L[k-1]}{k-1}$,
então, para $n=k$, $\text{mean}(L, k) = \frac{L[1] + \dots + L[k]}{k}$

P.I:

$$\begin{aligned} \text{mean}(L, k) &= L[k]/k + \text{mean}(L, k-1) \times (k-1)/k \\ &= L[k]/k + \frac{(L[1] + \dots + L[k-1]) \times (k-1)}{(k-1)k} = \frac{L[1] + \dots + L[k]}{k} \end{aligned}$$

5) Func seq(dp):

max_val = max(DP)

A[1] = DP[1]

for $i=2 \rightarrow n$:max_idx = i if $DP[i] = \text{max_val}$

A[i] = min(DP[i], DP[i] - DP[i-1])

list = []

for $i = \text{max_idx} \rightarrow 1$:

list.push(A[i])

if $DP[i] = A[i]$: break

return list.reverse()

(assumindo que inicialmente não temos acesso a A)

6) Func TD(sums, N, idx):

if dp[sums, idx] has value: return dp[sums, idx]

if $\text{idx} = n$ { if (for $i=1 \rightarrow 3$: sums[i] = sum_N): dp[sums, idx] = True

else dp[sums, idx] = False }

dp[sums, idx] = False

for $i=1 \rightarrow 3$: { sums[i].push(N[idx])

if TD(sums, N, idx+1): dp[sums, idx] = True; break }

sums[i].pop() }

return dp[sums, idx]