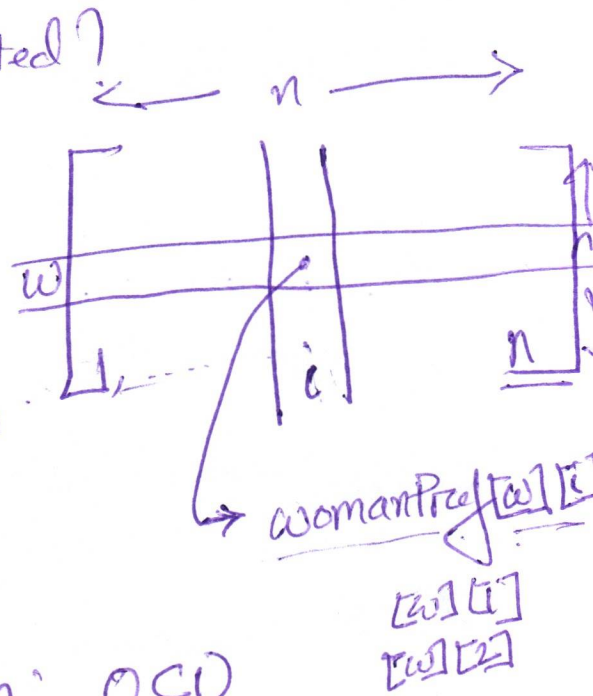


(Q0) How is the input represented?

2-D arrays

womanPref [w] [i] = ID of the i th most preferred man in L_w .

manPref [m] [i] = ID of the i th most preferred woman in L_m .



Init: _____

Query: Read a specific location: $O(1)$

Update: _____

(Q1) How to find a free woman?

Maintain a linked list free (of free women).

Init: Add all n women to free: $O(n)$

Query: Pick 1st woman w in free (+ delete w from free)

Update: (w proposed to m) $O(1)$

Case 1: m is free \Rightarrow do nothing to free.

Case 2.1: $w' > w \Rightarrow$ add w to free.

Case 2.2: $w > w' \Rightarrow$ add w' to free.

(Q2) How to find w's best unproposed man?

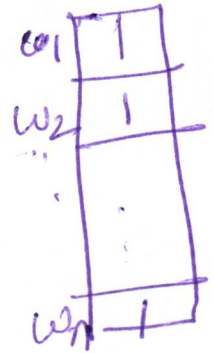
Maintain an array next of length n.

$next[w] = \text{rank of the man } m \text{ that } w \text{ should propose to.}$

Init : $next[w] = 1 \quad \forall w : O(n)$

Query : Who should w propose to?
Read $next[w] : O(1)$

Update : $next[w] = next[w+1] \quad O(1)$



(Q3) How to find who m is engaged to?

Maintain an array current of size n.

$current[m] = \begin{cases} -1 & \text{if } m \text{ is free} \\ w & \text{if } (m,w) \text{ is engaged} \end{cases}$

Init : $current[m] = -1 \quad \forall m : O(n)$

Query : Read $current[m] : O(1)$

Update : if (m,w) get engaged after w's proposal,
 $current[m] = w : O(1)$

So far : $Init : O(n) + O(n) + O(n) = O(n)$; Query/Update : $O(1)$

(Q4) Is $w' > w$?

Scan $manPref[m]$ to find ranks $i' & i$ of women $w' & w$. check: $\frac{i' > i}{O(1)}$

