CSE 101 4-30-26

Pat: ext. 2 days to Tuesday 5/5

Runtime of DFS:

$n = |V(G)|$, $m = |E(G)|$

Initialize: $\Theta(n)$

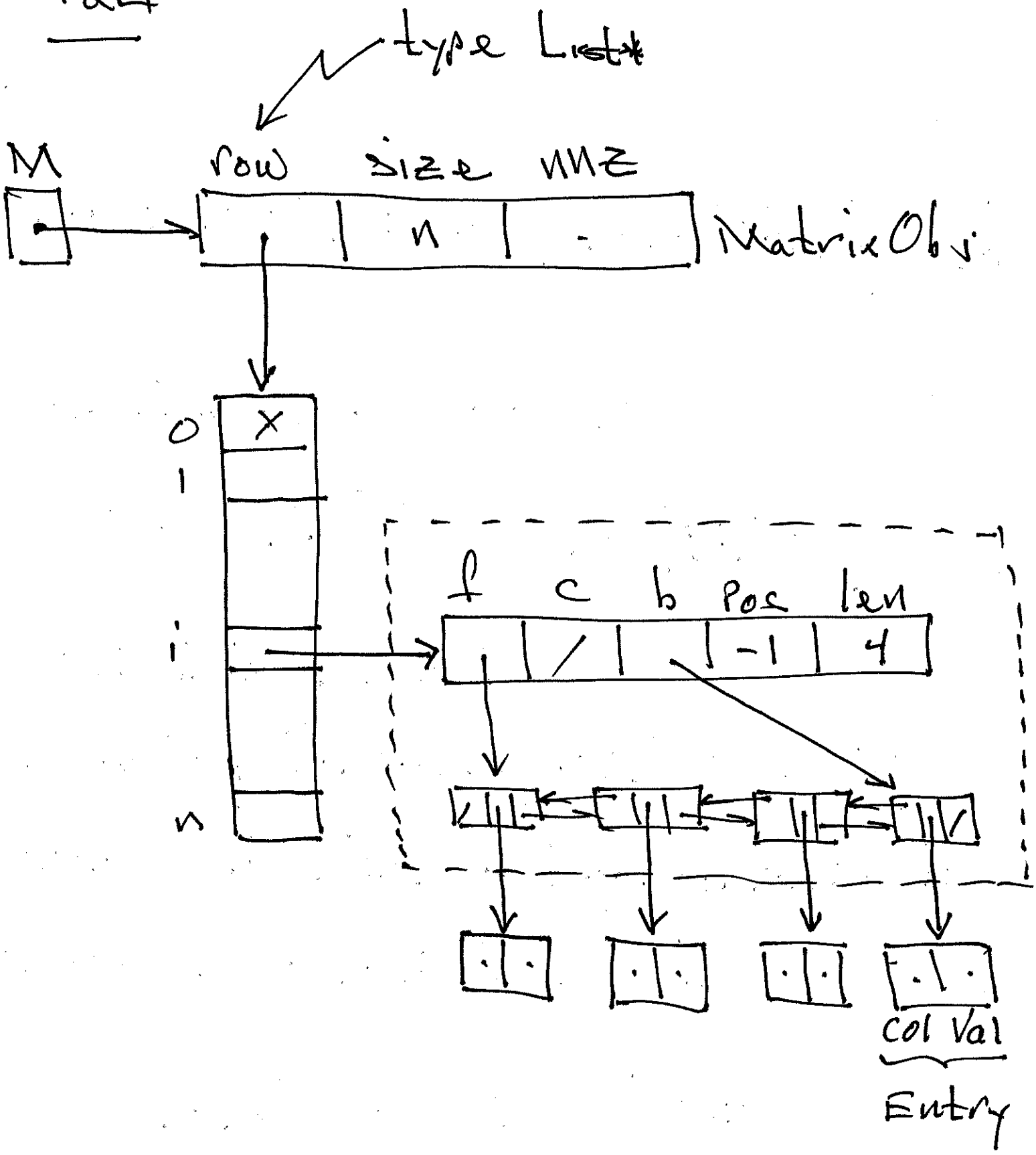
Main loop (not counting visit()): $\Theta(n)$

inside visit (not counting recursion): $\Theta(m)$

calls to visit(): $\Theta(n)$

total cost: $\Theta(n+m)$

Part


type List*
M
row
size
nnz
n
-
MatrixObj
0
1
i
n
X
f
c
b
Pos
len
-1
4
Col Val
Entry

Build Matrix ops order

- constructor
- makeZero
- destructor
- printMatrix
- changeEntry ← **!!!!

Easy
- copy(), transpose(), scalarMult()

Helper • dot(), add(), sub()

Hard • product(), sum(), diff()

changeEntry(i, j, x)

change M_{ij} to x

cases

1. $M_{ij} = 0$, $x = 0$: do nothing
2. $M_{ij} \neq 0$, $x = 0$: delete
3. $M_{ij} = 0$, $x \neq 0$: insert
4. $M_{ij} \neq 0$, $x \neq 0$: overdrite

helper fcn dot(L_1, L_2)

L_1 : (10, 1.0), (30, 2.0), (50, 3.0), (70, 7.0)

L_2 : (20, 4.0), (40, 5.0), (50, 6.0) ___

↓

+ 18.0
accumlate
sum

Product(A, B)

T = Transpose(B)

for i = 1 to n ← size

if A_i is all zeros

continue

for j = 1 to n

$C_{ij} = dot(A_i, T_j)$

~~Return~~

return C

Runtime of Product()

$$\Theta(n^2 + a \cdot b)$$

Idea (not required)

instead of array of lists,
have List of Lists

$$\Theta(n + a \cdot b)$$