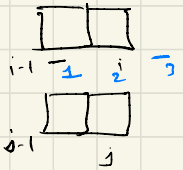
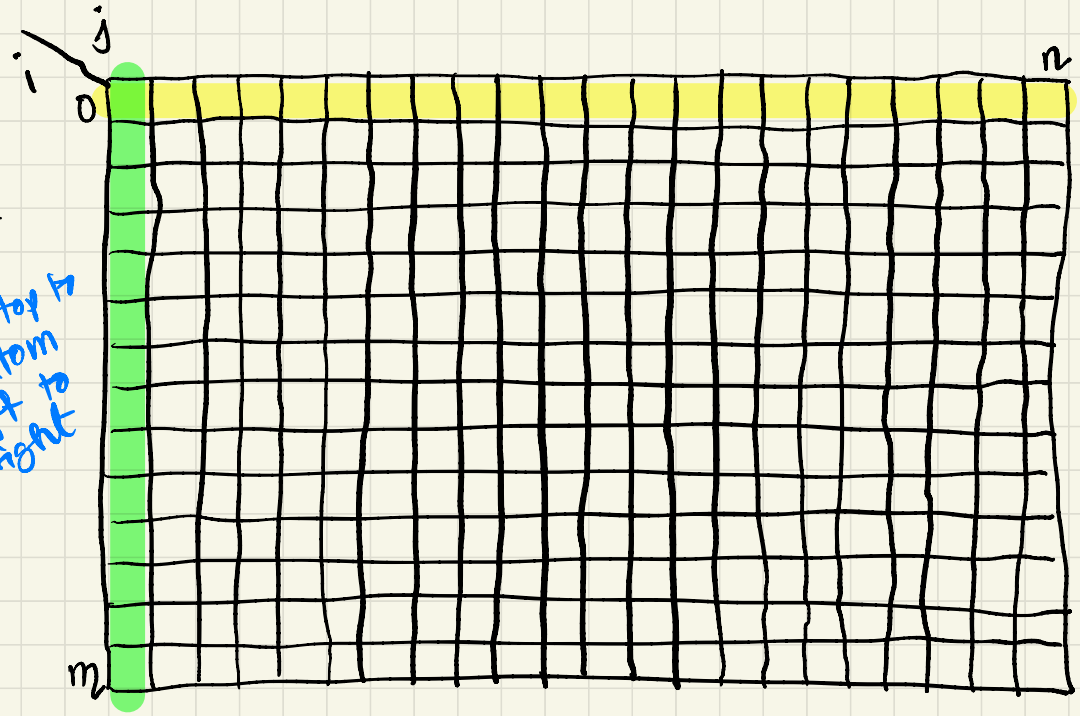
cost(i,j) = cost of the least expensive operation sequence that transforms $x[1 \ldots i]$ to $y[1 \ldots j]$. (At termination the idx ptrs for x & y would be at ith & jth respectively)

cost(i,j) =

- cost(i-1, j-1) + copy from i to j — only if $x[i] == y[j]$
- cost(i-1, j-1) + replace at j
- cost(i-1, j) + delete at i
- cost(i, j-1) + insert at j
- cost(i-1, j-1) + twiddle i,j — only if $i,j > 2$ & $x[i] == y[j-1]$ & $x[i-1] = y[j]$





Fill top to bottom left to right

$cost(0, j)$ $0 \le j \le n$

$= j$ inserts

x

y

1 2 . . . n

$cost(i, 0)$ $0 \le i \le m$

$= i$ deletes

```
for j in range(0, n+1):
    cost(0, j) = j * cost(insert)

for i in range(0, m+1):
    cost(i, 0) = i * cost(delete)

for i in range(1, m+1)
    for j in range(1, n+1)
        cost(i, j) = min( cost(i-1, j-1) + cost(copy)   if x[i] == y[j] else inf,
                          cost(i-1, j-1) + cost(replace),
                          cost(i-1, j)   + cost(delete),
                          cost(i, j-1)   + cost(insert),
                          cost(i-1, j-1) + cost(twiddle) if i > 2 & j > 2
                                                        & x[i] == y[j-1]
                                                        & x[i-1] == y[j]
                                                        else inf )
```