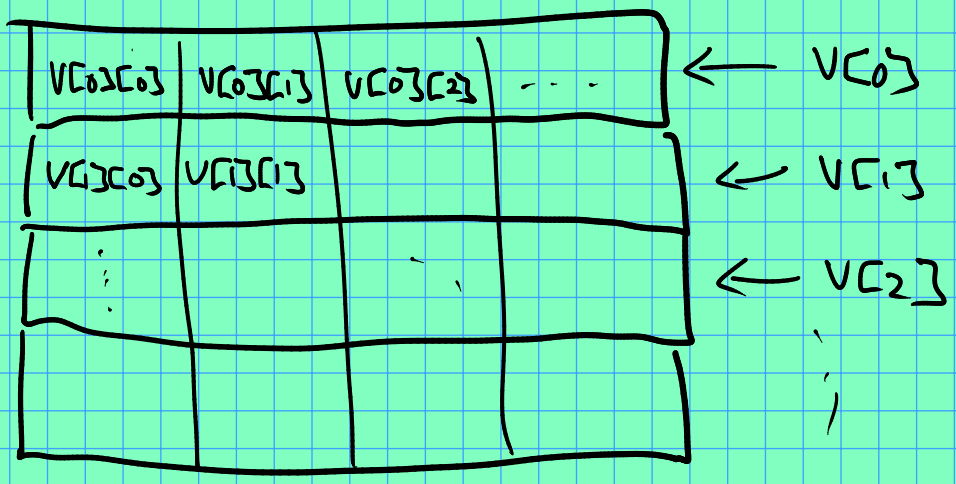
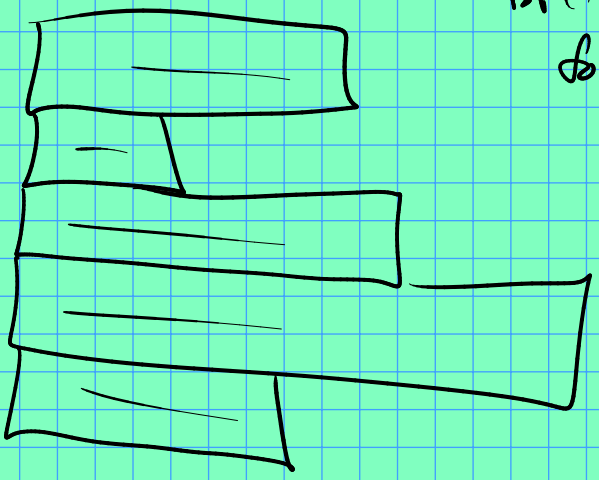
vector <vector <int>> V;





```
for(i=0; i<V.size(); i++) {
  for(j=0; j<V[i].size(); j++) {
    - - -
```

From last time: polynomial evaluation: Horner

$f(x) = \sum_{i=0}^{n} a_i x^i$, want to find value of f at some specific input x, e.g. $f(3)$.

Original method cost 2 mults per loop iteration.

Can we improve?

Say $f(x) = 4x^3 + 3x^2 + 2x + 1$

iter. 0 4

1 $4x + 3$

2 $(4x + 3)x + 2 = 4x^2 + 3x + 2$

3 $(4x^2 + 3x + 2)x + 1 = f(x)$ ✓

("Horner's Rule")

$((4x + 3)x + 2)x + 1$

Deeper look at string.

– vector<char> in a thin disguise.

vector	string
v.size()	s.length()
v[i]	s[i]
v.push_back(x)	s += x

↖ x could be char or string!

Exercise: try to write a function that takes 2 strings x, y and returns true $\iff$ x is a substring of y.

E.g. x = abc, y = defabcdef ✓

x = abc, y = aa bb cc ✗

(like ctrl-f, or /)