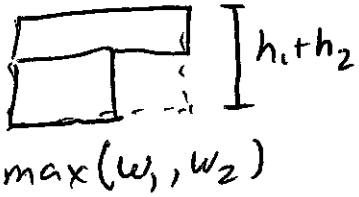
Schedule

Thurs 4/12 Guest Lecture - Dr. Richard Ewetz

→ Rectangle Packing



Tues 4/17 Dr. Ewetz →

Thurs 4/19 Travis - Final Exam Review

Tues 5/1 Final Exam 1-4pm

Today's Topics

Dynamic Programming

1. Matrix Chain Multiplication
2. Edit Distance

When coding first conceptualize **WHAT** the recursive function will return!!!

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

$\binom{n}{k}$ → # of ways to choose k items out of n

$I_0 \quad I_1 \quad I_2 \ \ldots \ I_{n-1}$

Or I could count separately:

- all sets with I_{n-1} → (Choosing from $n-1$ items, need to choose $k-1$) → $\binom{n-1}{k-1}$
- all sets without I_{n-1} → (Choosing from $n-1$ items, need to choose k) → $\binom{n-1}{k}$

f(engineer, nine)
= f(enginee, nine)

---

f(enginee, nine) =
f(engine, nin) // every match w/ last e
+ f(enigne, nine) // all other matches

MCM

$(A)(B(CD))$

$(A \cdot B) \cdot (C \cdot D)$ (matrices)

$2\times4 \quad 4\times3 \quad 3\times5 \quad 5\times1$ (dimension $r \times c$)

Product $A \times B$ exists if # cols A = # row B

# of mult to calculate $A \times B$ is (#rows A)(#cols A)(#cols B)

Result has dimensions of (#rows A) × (#cols B)

→ Cost $(A \cdot B)(C \cdot D)$

$2\times4\times3 \quad 3\times5\times1$

$24 + 15 = 39$

$\dim(AB) = 2\times3$ } 6 mult

$\dim(CD) = 3\times1$ }

Total $= 39 + 6 = \boxed{45}$

$CD \rightarrow$ 15 mult

$(B \quad (CD)) \rightarrow 4\times3\times1 = 12$ mult

$4\times3 \quad 3\times1$

$A \quad (B(CD)) \rightarrow 2\times4\times1 = 8$

$2\times4 \quad 4\times1$

Total $= 15 + 12 + 8 = \boxed{35}$

Of all possible parenthesizations the product which leads to the fewest multiplications?

Example MCM Problem

A · B · C · D E

2×4 4×3 3×5 5×1 1×2

	A	B	C	D	E
A	x	24	54	35	39
B	x	x	60	27	35
C	x	x	x	15	21
D	x	x	x	x	10
E	x	x	x	x	x

AB 2×3, CD 3×1
AC 2×5 CE 3×2
AD 2×1 DE 5×2
BE 2×2
BC 4×5
BD 4×1
BE 4×2

AB → 2×4×3=24, BC → 4×3×5=60, CD = 3×5×1=15,
DE = 5×1×2 = 10

ABC $\xrightarrow{\min}$ (AB)C, $24 + \overset{2\cdot3,\ 3\cdot5}{2\times3\times5} = 24+30 = \underline{\underline{54}}$

↘ A(BC)) $60 + 2\times4\times5 = 60+40 = \underline{\underline{100}}$

BCD → (BC)D $60 + 4\times5\times1 = 60+20 = 80$

↘ B(CD) $15 + 4\times3\times1 = 15+12 = \boxed{27}$

CDE → (CD)E $15 + 3\times1\times2 = 15+6 = \boxed{21}$

↘ C(DE) $10 + 3\times5\times2 = 10+30 = 40$

ABCD → A(BCD) $27 + 2\times4\times1 = \boxed{35}$

(AB)(CD) $24 + 15 + 2\times3\times1 = 45$

(ABC)D $54 + 2\times5\times1 = 64$

BCDE → B(CDE) $21 + 4\times3\times2 = 45$

(BC)(DE) $60 + 10 + 4\times5\times2 = 110$

(BCD)E $27 + 4\times1\times2 = \boxed{35}$

4/10/18 (5)

ABCDE →

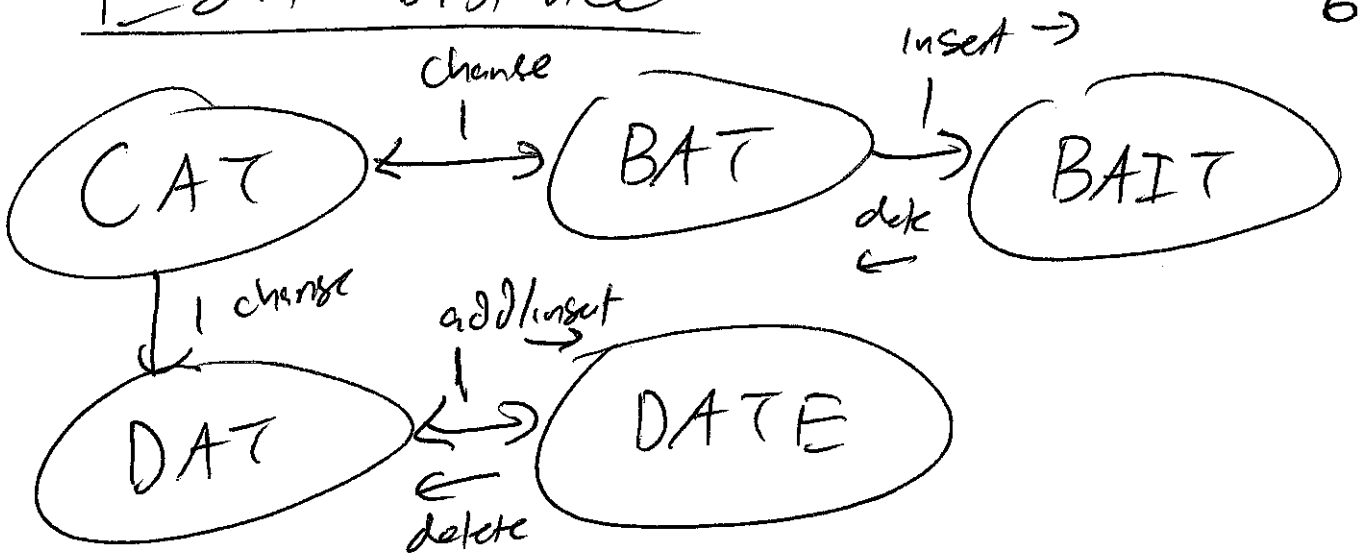
$A(BCDE) = 35 + 2 \times 4 \times 2 = 51$

$(AB)(CDE) = 24 + 21 + 2 \times 3 \times 2 = 57$

$(ABC)(DE) = 54 + 10 + 2 \times 5 \times 2 = 84$

$(ABCD)E = 35 + 2 \times 1 \times 2 = \boxed{39}$

Edit Distance



Given two strings s, t, what is the fewest # changes necessary to change s into t.

In lcs, $f(i,j) = 1 + f(i-1, j-1)$ } if last chars match

↑ greedy char match

In edit distance, $f(i,j) = f(i-1, j-1)$, if last chars match ↙ no extra cost

if $s[i] != t[j]$

then $\min($ $1 + f(i-1, j)$, // del 1 char s

$1 + f(i, j-1)$, // del 1 char t

$1 + f(i-1, j-1)$ // switch last letter in s to last letter in t. $)$