$$T(n) = 3T\left(\frac{n}{2}\right) + O(n^2)$$

$$\Rightarrow \quad T(n) = \boxed{O(n^2)}$$

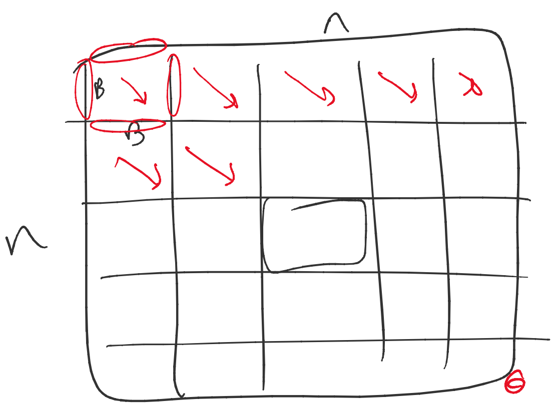
$$S(n) = S\left(\frac{n}{2}\right) + O(n)$$

$$\Rightarrow \quad S(n) = \boxed{O(n)}$$

## Time Improvement (Masek & Paterson '80)

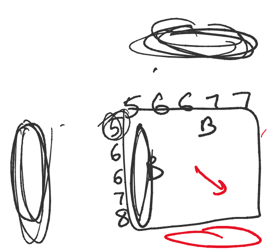
Assume $|\Sigma|$ const. $\quad m = n$

idea - precompute answers for all possible subproblems of size $B$ & store in table




$$\boxed{\left(\frac{n}{B}\right)^2}$$

by table lookup




input: 2 strings of length $B$ in $\Sigma^*$ & 1st row / 1st column

# choices $|\Sigma|^B \cdot |\Sigma|^B$ $\quad 2^B \quad 2^B$

$|\Sigma| \geq 2$

# subproblems of size $B$

$$\leq |\Sigma|^{2B} \cdot 2^{2B} \leq |\Sigma|^{4B} = n^{4\varepsilon} < \sqrt{n}$$

Set $B = \varepsilon \log_{|\Sigma|} n$

$\varepsilon = 1/8$

$O(n^2/\log^2 n)$

$\varepsilon = 1/8$

total preprocessing time $\leq O(\sqrt{n}\log^2 n)$
$= o(n)$.

final time bd
$$O\left(\left(\frac{n}{B}\right)^2\right) = \boxed{O\left(\frac{n^2}{\log^2 n}\right)} \leftarrow$$

Small improvement!

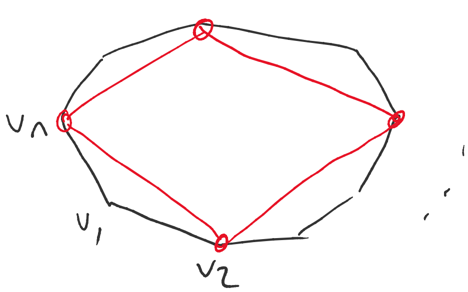
Thm (Abboud et al. '15, Bringman-Künnemann '15)
If there is an $O(n^{1.999})$ time alg'm for LCS,
then there would be "faster" alg'm for k-SAT
(Strong Exp Time Hypothesis)
(stronger than $P \neq NP$ hypothesis)

---

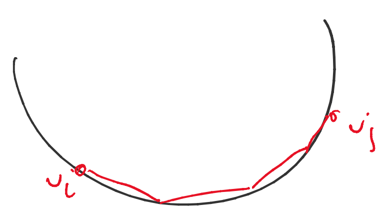
## Max-Perimeter L-gon

Given convex polygon $v_1, v_2, \ldots, v_n$ in ccw order and $L \leq n$,
find subpolygon with $L$ vertices maximizing perimeter

let $d(i,j)$ = Euclidean dist between $v_i$ & $v_j$




$L = 4$




## DP Method 1

Define subproblems:

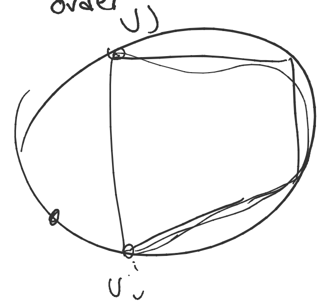
## DP Method 1

Define subproblems:

for each $1 \le i < j \le n$, $\ell \le L$,

let $D^{(\ell)}(i,j)$ = max dist of $\ell$-link path from $v_i$ to $v_j$ in ccw order

Want $\max\limits_{1 \le i < j \le n} \left( D^{(L-1)}(i,j) + d(j,i) \right)$



**Recursive formula:**

$$\boxed{D^{(\ell)}(i,j) = \max_{i \le k \le j} \left( D^{(\ell-1)}(i,k) + d(k,j) \right)}$$

**Base Case:** $D^{(1)}(i,j) = d(i,j)$.

**Evaluation order:** increasing $\ell$.

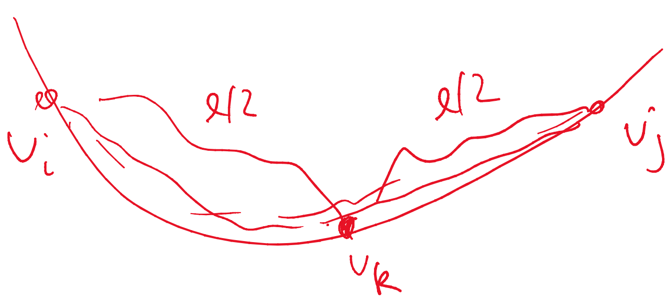
$\Rightarrow$ # table entries $= O(n^2 L)$

each entry $O(n)$ time

$\Rightarrow$ total time $\boxed{O(n^3 L)} \le O(n^4)$

---

## DP Method 2



$D^{(\ell)}(i,j) = \left( \max \; D^{(\ell/2)}(i,k) + D^{(\ell/2)}(k,j) \right)$ if $\ell$ even

$$D^{(\ell)}(i,j) = \begin{cases} \max\limits_{i \le k \le j} \left( D^{(\ell/2)}(i,k) + D^{(\ell/2)}(k,j) \right) & \text{if } \ell \text{ even} \\ \max\limits_{i \le k \le j} \left( D^{(\ell-1)}(i,k) + d(k,j) \right) & \text{if } \ell \text{ odd} \end{cases}$$

only need $O(\log L)$ values of $\ell$.

$\Rightarrow O(n^3 \log L)$ time

('repeated squaring)

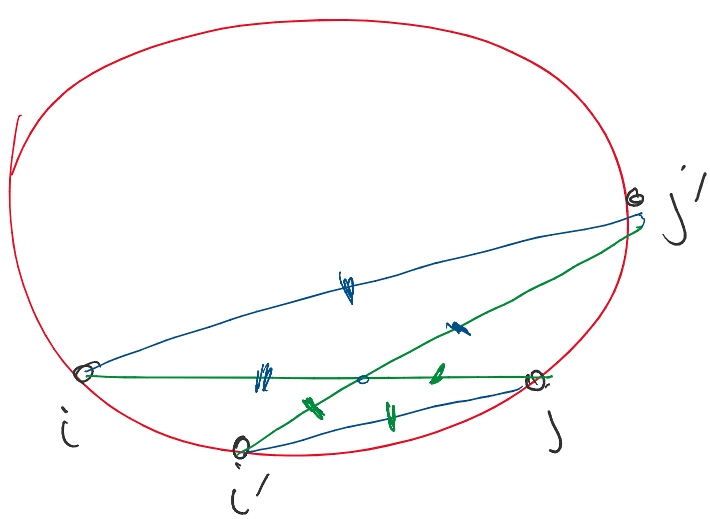
---

## Improved DP Method 2:

**Def** Function $d$ satisfies convex Monge property

if $\forall\ i \le i' \le j \le j'$,

$$d(i,j) + d(i',j') \geq d(i,j') + d(i',j)$$




for our problem, it follows by triangle ineq.

**Lemma** (F. Yao '82)

Let $D(i,j) = \max\limits_{i \le k \le j} \big( d(i,k) + d(k,j) \big)$

$K(i,j) =$ $k$ that attains this max

If $d$ is convex Monge,

then a) $K$ is monotone increas. in each row & in each column

b) $D$ is also convex Monge.

**Cor** If $d$ is convex Monge, compute $D$ in $O(n^2)$ time.