

Older generation of men  
devoted to soccer or tennis  
revenue from lotteries

consumers on pension

\* young people:  
+ young people continuity:

issuing + licensing + reuse  
recycling + reuse  
upcycling

"softener", "perfume", "deodorant"

consciousness recycling  
environmentalism

parents  
children  
influence & influence

parents  
children  
influence & influence

time - investment  
time - investment  
time - investment  
time - investment

Problems

- \* Microeconomics (monetary)
  - Interest rates
    - o nominal interest rate = real interest rate + inflation rate
    - o real interest rate = nominal interest rate - inflation rate
  - Real interest rates
    - o real interest rate = nominal interest rate - inflation rate
  - Interest rates
    - o nominal interest rate = real interest rate + inflation rate
- \* Macroeconomics
  - Interest rates
    - o nominal interest rate = real interest rate + inflation rate
    - o real interest rate = nominal interest rate - inflation rate
  - Real interest rates
    - o real interest rate = nominal interest rate - inflation rate
  - Interest rates
    - o nominal interest rate = real interest rate + inflation rate

A gain Fibonacci

fib(n: integers positive)

if  $n \leq 2$   
return 1

else

return fib(n-1) + fib(n-2)

$$T(n) = 2^n$$

1 2 3 5 ...  
logarithmic growth  $\rightarrow$  O(n) time complexity

1 1 2 3 5 ...

$$T(n) = T(n-1) + T(n-2) + O(1)$$

$$T(n) \geq 2^n \cdot T(n-2) = \dots \geq 2^{n/2}$$

fib(n):

```

dp[1,...,n]: array of nums
for i ← 1 to n do
    dp[i] ← -1

```

dp[1] ← 1

dp[2] ← 1

return fibAux(n, dp)

fibAux(n, dp)

if dp[n] == -1

return dp[n]

else if n == 2

else

res ← 1

else

res ← fibAux(n-1, dp) + fibAux(n-2, dp)

fibAux(k) - ce

choose nodes

beginning in ~~choose top~~

choose ~~bottom~~ nodes

fibAux(1), ..., fibAux(n)

= O(n)

return res

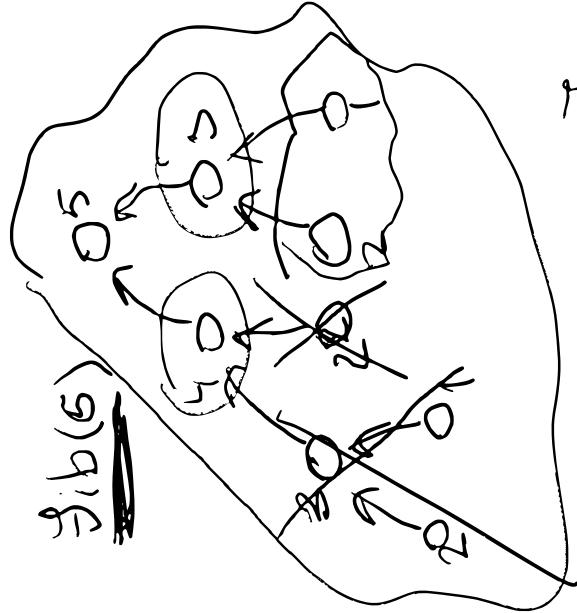
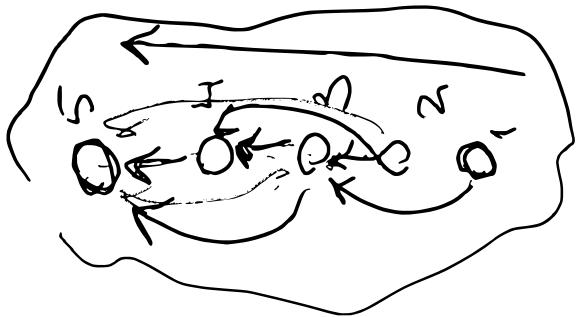
return res

return res

return res

return res

return res



$dp[1 \dots n]$   
for  $i \leftarrow 1$  to  $n$  do  
 $dp[i] \leftarrow -1$   
 $dp[1] \leftarrow 1$   
 $dp[2] \leftarrow 1$   
for  $i \leftarrow 3$  to  $n$  do  
 $dp[i] \leftarrow dp[i-1] + dp[i-2]$

```

    a, b : lists
    a ← 1, b ← 1
for i ← 2 to n do
    c ← (a + b)
    a ← b
    b ← c
return b

```

$$T(n) \approx n$$

$$\mu(n) \approx 1$$

- Rods
- LIS
- Subsets sum
- Knapsack
- Edit distance (Longest)
- LCS

hologram

- top-down: recursion + memoization
- two levels of computation:
  - top-down computation using cache
  - bottom-up computation using stack

Conducting business through a distributor.

Process or sequence of steps followed by manufacturer to bring products to market via intermediaries such as distributor, wholesaler and retailer.

- Using intermediaries to sell products

Non-resale route

- Convenience stores and gasoline stations
- DACs or discount outlets
- Trade shows or exhibitions

Rely

- Bottom-up: Auto parts and bulky household items
- Top-down: Computer parts like AGS - Q

Pensando per percus +  
percussione + nozionale

passing.

Ats no sonar percutir no sonar  
nótono, to an un espíritu

percutir ignorante en "brute  
force" no sonar

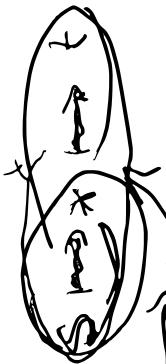
"balancieno"

brute force, no sonar

un percusseur!



S → t



P → P → P → P → P

S →

Balances between input-output and output-import

Imports - Exports = Trade balance

Trade balance = Exports - Imports

5 stages

5 phases of growth and development:

- ① Deposition phase:  
- Ovarian ripening + recruitment of follicles + selection of dominant follicle
- ② Neovascularization phase:  
- Endometrial thickening + vascularization of endometrium
- ③ Hormonal phase:  
- Increase in blood supply + increase in size of endometrium
- ④ Secretory phase:  
- Secretion of mucus + increase in thickness of endometrium
- ⑤ Decidual phase:  
- Decidualization of stromal cells + formation of decidua

⑥ Reversing opuscularization

(above) Top bottom inner outer longitudinal transverse cross oblique vertical horizontal depth length width height

## Poly cutting

Kjære alle sammen fra minden hjørne?  
Dagens tema er poly cutting.  
Det kommer nippelose, pentagon og rombe  
Vi har nippelos. Det kommer romme  
Gammelt maned med en del spennende  
Kjære alle sammen kanskje du vil ha med meg i  
Konserten om natt? Du  
Vil du ikke komme med meg til min  
Konsert? Du kan ikke komme med meg til min  
Konsert?

is known as  
leaching normal to surface

so when we encounter

anions like  $\text{Cl}^-$  or  $\text{SO}_4^{2-}$  etc.

they will be adsorbed on surface

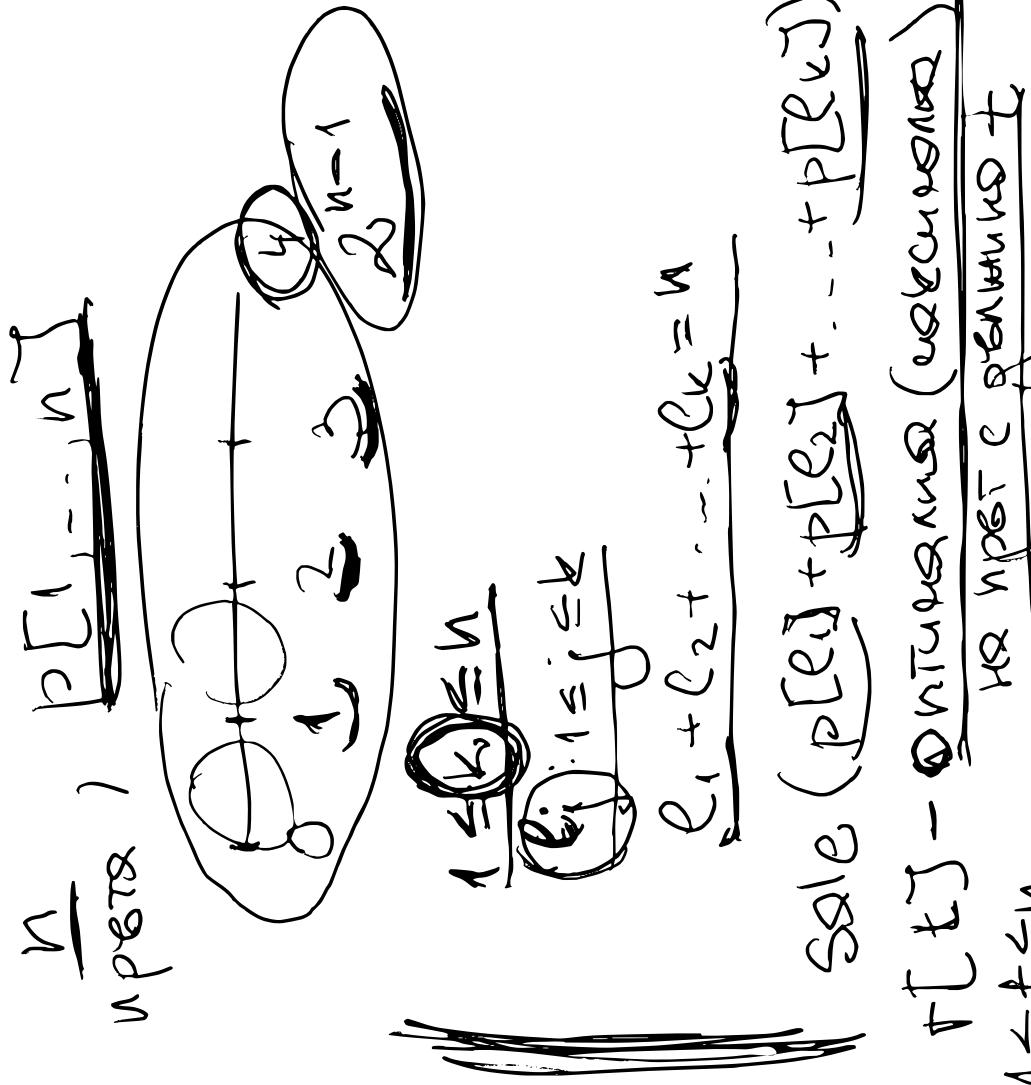
and released on downward flow

so when cations come down downwards

they will be adsorbed on surface

so when cations come down downwards





$$c_1 + c_2 + \dots + c_k = n$$

~~$\sum_{k=1}^n \binom{n}{k} = 2^n$  (incorrect)~~

~~$\sum_{k=1}^n \binom{n}{k} = \max(p[e_1] + p[e_2] + \dots + p[e_n])$~~

~~$\sum_{k=1}^n \binom{n}{k} = \max(p[e_1] + p[e_2] + \dots + p[e_n])$~~

~~$\sum_{k=1}^n \binom{n}{k} = \max(p[e_1] + p[e_2] + \dots + p[e_n])$~~

~~$\sum_{k=1}^n \binom{n}{k} = \max(p[e_1] + p[e_2] + \dots + p[e_n])$~~

~~$\sum_{k=1}^n \binom{n}{k} = \max(p[e_1] + p[e_2] + \dots + p[e_n])$~~

Ansatz:  $r[0] \leftarrow 0$

(i..n) : (max) ~~as record~~  $i < l$   
 $r[j] \leftarrow 0$  zuher.

(st+cp)  $\leftarrow \max(p[i], \max(r[i-1] + r[i-2], \dots, r[l-1] + r[l-2]))$ .



$\leftarrow r[i] \leftarrow \max(r[i-1], r[i-2], \dots, r[l-1], r[l-2])$

(o)

$r[i] \leftarrow \max(r[i-1], r[i-2], \dots, r[l-1], r[l-2])$

$r[i] \leftarrow \max(r[i-1] + r[i-2], \dots, r[l-1] + r[l-2])$



```

CutPad(p[i])
if i = 0
    return 0
res ← -∞
for j ← 1 to i do
    res ← max(res, p[i] + CutPad(p[j]))
return res

```

CutPad(p[i])

$$\begin{aligned}
 T(0) &= O(1) \\
 T(n) &= \sum_{j=1}^{n-1} T(j) + O(1) \\
 &\quad \vdots \\
 T(n) &\in \Theta(2^n)
 \end{aligned}$$

```

CutRod(p, n):
    r[0, ..., n]: array of ints
    for i ← 0 to n do
        r[i] ← -∞
    return cutRodAux(p, n, r)

cutRodAux(p, i, r)
    if r[i] = -∞
        return r[i]
    else if i = 0
        res ← 0
    else
        res ← -∞
        for j ← 1 to i do:
            res ← max(res, p[j] + cutRodAux(p, i-j, r))
    return res

```

CutRodIterative( $\underline{p}$ , n)

$dp[0, \dots, n]$ : array of ints  
 $dp[0] \rightarrow 0$

for  $i \leftarrow 1$  to  $n$  do:

$\maxRev \leftarrow -\infty$

for  $j \leftarrow 1$  to  $i$  do:

$\maxRev \leftarrow \max(\maxRev, p[i-j] + dp[j])$

$O(n^2) = O(n^2)$

$dp[i] \leftarrow \maxRev$

return  $dp[n]$

- honest increasing subsequence.
- recursive solution  $\Theta(2^n)$
- memoized recursive solution  $\Theta(n^2)$
- dynamic programming solution  $\Theta(n^2)$

honest increasing subsequence.

Recursive solution

Let  $i_1, i_2, \dots, i_k = 1, 2, \dots, n$ .  
Let  $a_{i_1}, a_{i_2}, \dots, a_{i_k}$  be a sequence of length  $k$  such that  $a_{i_1} < a_{i_2} < \dots < a_{i_k}$ .

Let  $S$  be the set of all such sequences.

Let  $f(i)$  be the number of sequences in  $S$  of length  $i$ .

Then  $f(i) = f(i-1) + f(i-2) + \dots + f(1)$ .

Initial conditions:  $f(1) = 1$ ,  $f(0) = 1$ .

Algorithm:

1. Initialize  $f(0) = 1$ .
2. For  $i = 1$  to  $n$ :
  - 3.     Initialize  $f(i) = 0$ .
  - 4.     For  $j = 1$  to  $i$ :
    - 5.          $f(i) = f(i) + f(j-1)$ .

## Subset sum

Aazen bue verurs or  
nonobut enu yeneren a A[1,..,n]  
u nonouten ugen eucos.  
Denpcte coudcubya au  
 $(\exists T \subseteq \{1, \dots, n\}) \left[ \sum_{i \in T} A[i] = S \right]$  (?