

Αλεξανδροι

samaras@uom.edu.gr

- Το βιβλίο γνωρίζει Python. Γράφοντας Python, Ν. Σαμάρας

## Γράφοντας

$$n! = 1 \times 2 \times 3 \times 4 \times \dots \times (n-1) \times n$$

$$6! = 1 \times 2 \times 3 \times 4 \times 5 \times 6$$

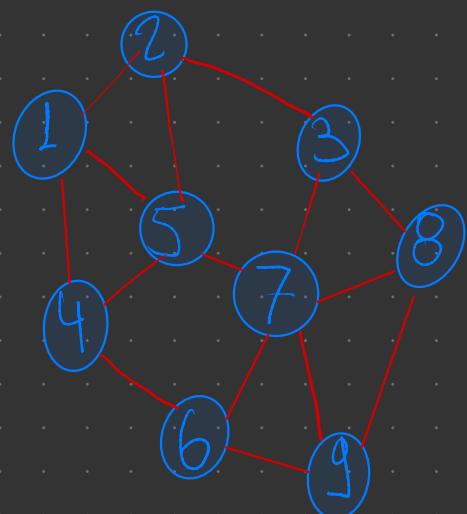
Δεν είχε λειτουργήσει  
από τη συμβατική αριθμητική

## ~ Traveling Salesman Problem

Οριζόντιος σύνδεσης που διασφαλίζει ότι όλα τα κόμβα στην πόλη παραστέλλονται μεταξύ τους

$$\text{Πλήθος λυσεών: } \frac{(n-1)!}{2}$$

## Γράφος (Graph):



- κόμβοι (nodes)
- αρμόδιοι (edges)

$$3.678 \cdot 10^5 \xrightarrow{\text{scientific form}} 3.678 \times 10^5$$

$$3.678 \cdot 10^{-5} = 3.678 \times 10^{-5}$$

# ~ Page Rank

στόχος: κατάταξη των σελίδων ανάλογα με χρονική σειρά της προσαρμογής της Google

Operation

$$\begin{aligned} a &= \frac{4}{3} \\ b &= a - L \\ c &= 3 \times b \\ e &= L - c \end{aligned}$$

Human

$$\begin{aligned} a &= \frac{4}{3} \\ b &= \frac{4}{3} - \frac{3}{3} = \frac{1}{3} \\ c &= 3 \times \frac{1}{3} = L \\ e &= 0 \end{aligned}$$

PC

$$\begin{aligned} a &= 1,333\ldots \\ b &= 0,333\ldots \\ c &= 1,000 \quad \eta \quad 0,999\ldots \\ e &= 2,2204 E-16 \quad \eta \quad 2,2204 \cdot 10^{-16} \end{aligned}$$

cylinder size  
pxarvys (Emach)

Επειώσις Εξαστίνυσ

$V = \lambda \rho \chi \nu \eta$

$\lambda = \lambda \rho \chi \nu \chi \nu$

$\bar{\lambda} = \lambda \rho \chi \nu \chi \nu \approx \pi \rho \chi \nu \nu 100$

L J floor

F T cell

# Τροπολόγηση και Στρατηγική

Υποδομική Τροπολόγηση (ΥΠΤ): Λύνονται όταν οι περιορισμοί στην παραγωγή είναι από την παραγωγή

Καταγόπιες ΥΠΤ:

~ Τροπολόγηση Αναζήτησης: Na boedei eis oroleio eiōdou tou ikavottoi

search problems

~ Τροπολόγηση Συρροής: Na peraorixiazorei q eiōdou czo wozc va ikavottoi pia idios

structurings problems

~ Τροπολόγηση Κατασκευής: Na kataskevazorei pia eiōdou tou va ikavottoi pia siōryza

constructions problems

~ Τροπολόγηση Απόρρητης: Na antypatizorei kala koo pia eiōdou ikavottoi pia siōryza

decisions problems

~ Τροπολόγηση Βελτιστοποίησης: Na boedei q kathwētai eiōdou tou ikavottoi pia siōryza

optimization problem

~ Τροπολόγηση Προσαρμογής: Na basiropothei pia siōryza ws tpos zon xpovo

adaptation problems

## Χαρακτηριστικά Αλγορίθμου

- Δεν περιέχει δομικές αντιφασεις
- Λύνει καθε στρατηγικόν του τροπολόγων
- Τελειώνει περα από πτενεράσμενο σημείο τρόπεων
- Δεν περιέχει εγγονεις του επηγεινούνται με νικητή περιού σπόντο
- Είναι πτενεράσμενος
- Κατα σημείου
- Έχει ειδούσα -εξόδου
- Είναι ανορθοδοξίους

# Τεχνικές Σχεδίασης Αλγορίθμων

- Οριγ. Bia - brute force
- Διαιρετική Βασικήνευ - divide conquer
- Μείωση και Βασικήνευ - decrease and conquer
- Μετασχηματισμός και βασικήνευ - transform and conquer
- Δυναμικός προγραμματισμός - dynamic programming
- Ανθεξιά - greedy

## Πολυπλοκότητα

- (1) Worst (best) case analysis
- (2) Average case analysis
- (3) Experimental analysis

Δικαία σύγκριση  
(Υπολογισμός Μετρητών)

- 1) Ιδιός προγραμματισμός
- 2) Ιδιοι υπολογισμοί προσομοιώσεων
- 3) Ιδια στρατηγική

Παλαιοί



Χρηματικοί  
Αλγόριθμοι

Σύγχρονοι



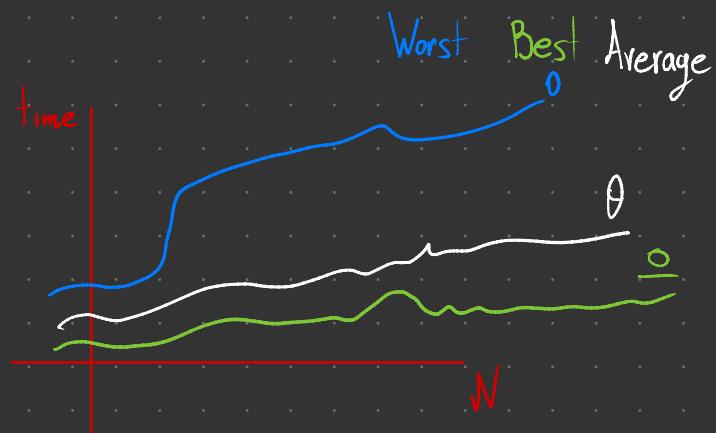
Χρηματικοί  
Αλγόριθμοι

# Συγχρησι Ηλεκτρονικών

$O(n)$  worst

$\Theta(n)$  average

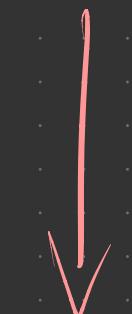
$\Omega(n)$  best



## Ηλεκτρονικά Χρόνου - Time Complexity

$\log_2 n$	Δοραγιθμική
$n$	Υπαρκή
$n \log_2 n$	$n$ Δοραγιθμική
$n^2$	Πολυωνυμικές
$n^3$	
$2^n$	Επεξεργαζόμενη
$n!$	Τριαγαρυντική

Best



Worst

Fibonacci Sequense = η σειρά αριθμών που ζεινείται από το

0 και κάθε αριθμός οπουται να το αλφοιση  
των 2 προηγουμένων

$$f(0) = 0$$

$$f(1) = 1$$

$$f(n) = f(n-1) + f(n-2)$$

Αλγόριθμος : fibl

Είσοδος : n

Έξοδος: f

1.  $f \leftarrow 0$
2.  $i \leftarrow 1$
3. for  $k \leftarrow 1$  to  $n$
4.  $f \leftarrow f+i$
5.  $i \leftarrow f-i$

## Find Second Max

1. Βρίσκω την μεγαλύτερη διαζύγιο  
και την κάνω floor

3. Τα ουρανού

2. Βρίσκω τα περιτύχεα οροικειών  
από την μεγαλύτερη και πιον καλ από  
την μεγαλύτερη πιπόστα

## Αλγορίθμος Horner

$$P \leftarrow T(1)$$

for  $i \leftarrow 2$  to  $n$

$$P \leftarrow P * r + T(i)$$

# Ψευδοτυχαιοι αριθμοι - PRNGs

$x$  (seed)

$m > 0$  max random numbers

$a > 0$  και  $< m$

$c > 0$  και  $< m$

$$r = (a \cdot x + c) \% m$$

καλεσ τιπες για

$c =$  πως αριθμος

$m =$  διαφορη του 2

S	A	U	S
E	N	M	E
Y	D	B	R
D	O	E	T
O	W	R	O

SEEDER  
GENERATOR

Σημείωση: Εάν είναι αριθμός που δεν  
εργαζεται.

Πήραμε περιοριστικά του με  $m$   
νηστεχει ουτού.

1.  $m$  και  $c$  δεν  
διαφορούνται

2. αριθμος  $(a-1)$  διαιρεται  
με τους πως αριθμους  
πιο παραγγελτες του  $m$

3. Αν το  $m$  διαιρεται με  
4, να διαιρεται και το  $(a-1)$

# Ταξινομηση με Επιλογή - Selection Sort ( $n-l$ )

Εργονούμε το πιο σεριο / περισσότερο σχολείο στον Τίβανα  
και το να νούμε για  $n-l$  φορες οντο καθε ψηφα βεβινε με  
θεση ανο μετα

# Ταξινόμηση με Εναλλαγή - Bubble Sort (zo τολυ n-l)

i = n  
t = -l

while i >= 2 and t ≠ 0

t = 0

for j = l to i-1

if T(j) > T(j+1)

[T(j), T(j+1)] = swap(T(j), T(j+1))

t = t + 1

i = i - 1

# Ταξινόμηση με Εισαρώση - Insertion Sort (n-l)

Από ειναι ζαξινούμενη τοτε περιουνω ανο το στοιχείο i=2 εως n  
με στο πωτο στοιχείο της θέσης

Άν δεν ειναι ναργυρω ανο το i=2 εως n και εδεχων

με τη στοιχεία j=0 εως i και το περιουνω σημ θέση του

## Σειριακή Αναζήτηση (εως n)

Επειχώ πε την ολόρο κάθε συνέχεια, σαρπάω σαν το  
βγω ή σαν ψύλω στο τελευταίο συνέχεια

## Αναζήτηση πε Βύρρα κ

Έσω γαζινόπενος ηλικίας. Επειχώ ενα βύρρα και και εξασθάνω  
κάθε κ-ορδο συνέχεια. Αν το συνέχεια σαν θέση ν είναι  
πιγούρη/μεγαλύτερη από την πλώ κ-1 θέσεις και κάνω δερδακόν

## Διαδικασία Αναζήτησης - Binary Search ( $\log_2 n$ )

Έσω γαζινόπενος ηλικίας. Βρίσκω το πιο κοντό  $\lfloor (i+j)/2 \rfloor = m$   
αν είναι μεγαλύτερο από το key, τότε  $j = m-1$ . Αν είναι  
πιγούρη ποτέ  $i = m+1$ . Και συνεχίζω αυτη η διαδικασία πέχοντας  
το  $i > j$  ή να βρεθεί το key

# Σειριακή Vs Δυαδική

~ Σε πινό πλήθος n

~ Σε στανιά χρόνου

~ Σε συχνή χρήση

~ Δεν υφίσταται σε:

- Μη ζαΐδων προσών τίτλων
- Συνδεόμενες άστρες
- Τίτλων κατακερπατισμού
- Τιτλο-διαστάζοντας τίτλων

## Τρόποι Αναζήτησης

- Ακριβές ταξιδιώρα - Exact Match
- Μερικό ταξιδιώρα - Partial Match
- Ερωτησης περιοχής - Range Query

# Δόρες Δεδομένων

Τηλεοπτική παράσταση χρόνου δορών

- Αποστολή σε διαχείριση ρυθμού
- Βελτίωση υπολογισμούς πολυπλοκούσας αλγορίθμου
- Βελτίωση θεωρίας πολυπλοκούσας αλγορίθμου
- Συγγραφή ευανάγνωσης τηγανιού κιβωτίου
- Ελλάξωση του προσαρτημένου τηγανιού κιβωτίου

Οι δόρες δεδομένων (ΔΔ) αποτελούνται από:

1. Άποψη των δεδομένων που αναπαριστά keys - κλειδιά
2. Άποψη των σύνθετων λειτουργιών - αλγορίθμων που αναπαριστά τιμές - operations
3. Άποψη περιλήψεων, διανύσματα, μέτρησης στα οποία αποθηκεύονται πληροφορίες χρήσιμες για την εντελεσθείσα πράξη.

# Τύποι Σετών ΔΔ

1. Σιαστήραση - traversal: προστέλλω / επεξεργασία κόρβων
2. αναζήτηση - search: εύρεση κόρβου για σετοφάκη
3. εισαγωγή - insertion: προσθέτω νέου κόρβου
4. διαγραφή - deletion: αφαίρεση υπορρογιας κόρβου
5. διατάξη - sorting: ταξιδιώση κόρβων σε πίστα σετοφάκης

## Σετοφάκη

Έχει πολλά εναυγενείσια σετοφάκη στην κορυφή-top OTTOU αναζανεται από προσεδανικά ορισμένα και πεινανεται οπαντανακάρια.

Είναι άδειο και  
δεκτικό στοιχείο σετοφάκης  
δεσμού

Εισαγωγή push(x)

Εξαγωγή pop(x), αν δεν αριθμούνται x  
μερικές ζωντανές ορισμένες

pop

done <- 0

if top ≥ 2

top <- top - 1

x <- S(top)

done <- 1

push

done <- 0

if top ≥ 2

S(top) <- a

top <- top + 1

done <- 1

Oupes

Έχει δύο σειρές εναντίον αρχή - front και  
έναν ορο τελος - rear. Συγχρόνως φεύγει το  
παλαιότερο οροίχειο.

enqueue : εισαγωγή ← ορο rear

dequeue : εξαγωγή ← ορο front

αδεια : front = rear

τεράζη : rear = n

μια θεοί γρια το γρια

## Συνδεσμένες Αισθήσεις

- Οποειδή συστημάτων : Ζηργή = πληροφορία, Σεική
- Δυνατική Δορύ : προσθαυτισμός συστημάτων
- Γραπτή Δορύ : κάθε συστημάτιο έχει ένα ενόπευο και ένα προγραμματικό, εκτός από το πινώσι και το σεκουντ.
- Αποδημευτική Συστηματική σε Συναντήσεις : Οι πράξεις προονταν
- Αισθητική Συστημάτων : ότι αλλάζει την άποψη ενός Σεική

Τύπασης : αναζητώντας, εισαγωγή, διαγράψη, και αλλαγή

## Αντανακλαστικές Αισθήσεις

Αντανακλαστικές Αισθήσεις : Τύπασης προς την ενόπευτη σύσταση.

Διατανακλαστικές Αισθήσεις : Τύπασης για την ενόπευτη προστασία

# ΑΓΓΛΙΚΗ ΟΥΝДЕΣΕΡΒΕΣ

1 → 1 2 3 4 5 6 7

next = 0 , zedevzaio

L → 6 4 5 8  
next → 2 0 3 L

free = 0 , σεν υπάρχουν δεοεις

Σεν είσαι για να κάνω το πρότερο ενδευταίο στοιχείο και  
σεικνει στο index του κατανούμενου, και το next του  
κατανούμενου σεικνει το index του zedevzaioυ

Διαγραφή

1. Διαγραφή πρώτου

2. Διαγραφή zedevzaioυ }  
3. Διαγραφή endevzaioυ } prev περιήλιξη

1.

objekti το πρώτο στοιχείο, περιγενετας σε free και  
το next του σεικνει το προηγούμενο free

2.

Έχουμε μια περιήλιξη prev που είναι το  
πρότερο ενδευταίο στοιχείο. Διαγραφώ το zedevzaio

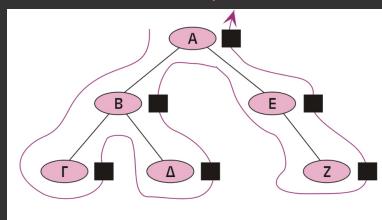
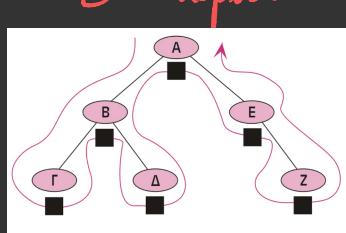
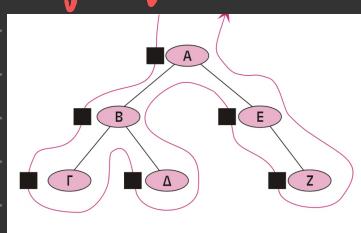
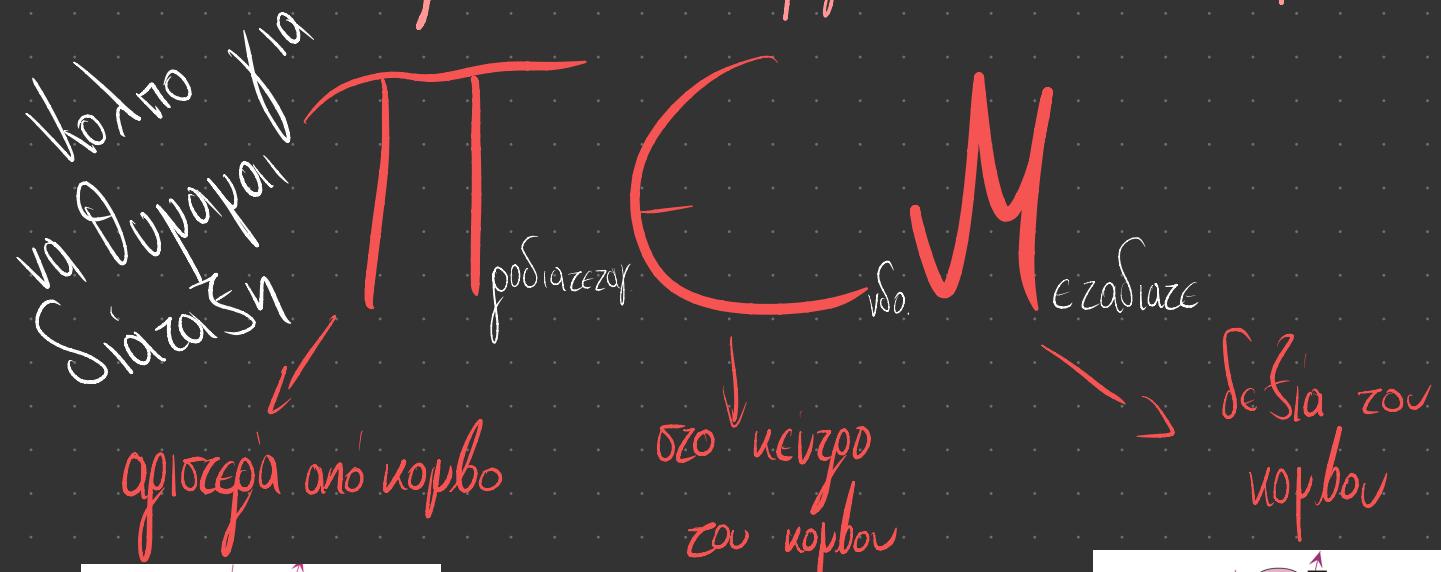
zo περισπένω σε free, του ονοτου zo  
next δείχνει στο npoγγουπένο free. και την  
αντικαθίστω zo next του prev pe 0

3. Εξουπερ πια περιλήγει prev nou exei zo  
npoγγουπένο στοιχείο από αυτο nou Οιδουπέ  
να διαγράψουμε. Του αντικαθιστούμε zo next  
pe αυτο του στοιχείου nou Οιδουπέ να διαγράψουμε  
διαγράψουμε το ειδικό στοιχείο & zo κανουμε free  
nou δείχνει στο npoγγουπένο free

$\Delta \text{EVrgo}$  = Εινας γράφος

- Καθε συν κορβοι-nodes και ουδετεραι pe πια  
ακρι - edge
- Ενα σεντρο exei zo οδηγιστων συν γραμμα
- Δενρgo pe n nodes exei n-1 edges

- Διαδορού<sup>path</sup> ανο είναι κόρβα  $N_x$  σε είναι  $N_y$ , ονομάζουμε την αναβολική κοδική για  $N_x/N_y$
- Μήκος<sup>length</sup> είναι το path ονομάζουμε τον αριθμό των κόρβων του
- Βάθος<sup>depth</sup> ή Επίπεδο<sup>level</sup> είναι το μήκος της παραδίκης διαδορού του κόρβου ανο την πίσια  $\gamma_{\text{root}} = \text{Bathos Level} - \text{Bathos}$
- Ύψος<sup>height</sup> είναι διεύρυνση ονομάζουμε τον αριθμό κόρβων της παραδίκης
- Βαθός<sup>degree</sup> είναι το μήκος ονομάζουμε τον αριθμό των πατέρων της κόρβας
- Βαθός διεύρυνσης είναι ο μηδενός βαθός κόρβων



$\sum_{j \in \text{wpos}} -\text{Heap}$ : Οι αριθμοί απόθηκεν πριν ήταν πρώτες  
θέσεις ενός διανυόματος Η έτσι ως:

$$H(i) \geq \max\{H(2i), H(2i+1)\} \quad \forall 2i+1 \leq n$$

Δεσμευτική Μνήμη = stack

Δυναμική Μνήμη = heap

Ταξινόμηση Μεταγονών

for  $i \leftarrow L$  to  $n$

$$C(S(i)) \leftarrow C(S(i)) + L$$

for  $i \leftarrow L$  to  $m$

$$C(i) \leftarrow C(i) + C(i-L)$$

for  $i \leftarrow n$  to  $1$  with step  $-L$

$$T(C(S(i))) \leftarrow S(i)$$

$$C(S(i)) \leftarrow C(S(i)) - L$$

# Ta3ivopyoy Radix

for  $i \leftarrow l$  to  $n$

$$C(S[i]) \leftarrow C(S[i]) + L$$

for  $i \leftarrow l$  to  $m$

$$C(i) \leftarrow C(i) + C(i - l)$$

for  $i \leftarrow n$  to  $l$  with step  $-1$

$$T(C(S[i])) \leftarrow S[i]$$

these is  $[C(S[i])] \leftarrow 8$

$$C(S[i]) \leftarrow C(S[i]) - l$$