



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

# Δομές Δεδομένων

Ιωάννης Γ. Τόλλης  
Τμήμα Επιστήμης Υπολογιστών  
Πανεπιστήμιο Κρήτης

# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



# Σημείωμα αδειοδότησης

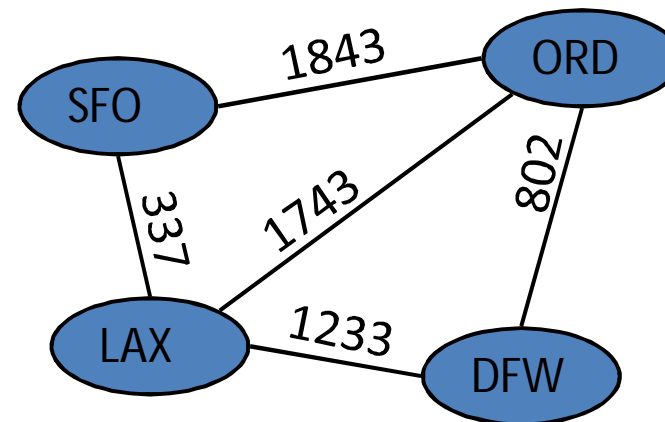
- Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση, Όχι Παράγωγο Έργο 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».

[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/>



- Ως **Μη Εμπορική** ορίζεται η χρήση:
  - που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
  - που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
  - που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο
- Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

# Γράφοι



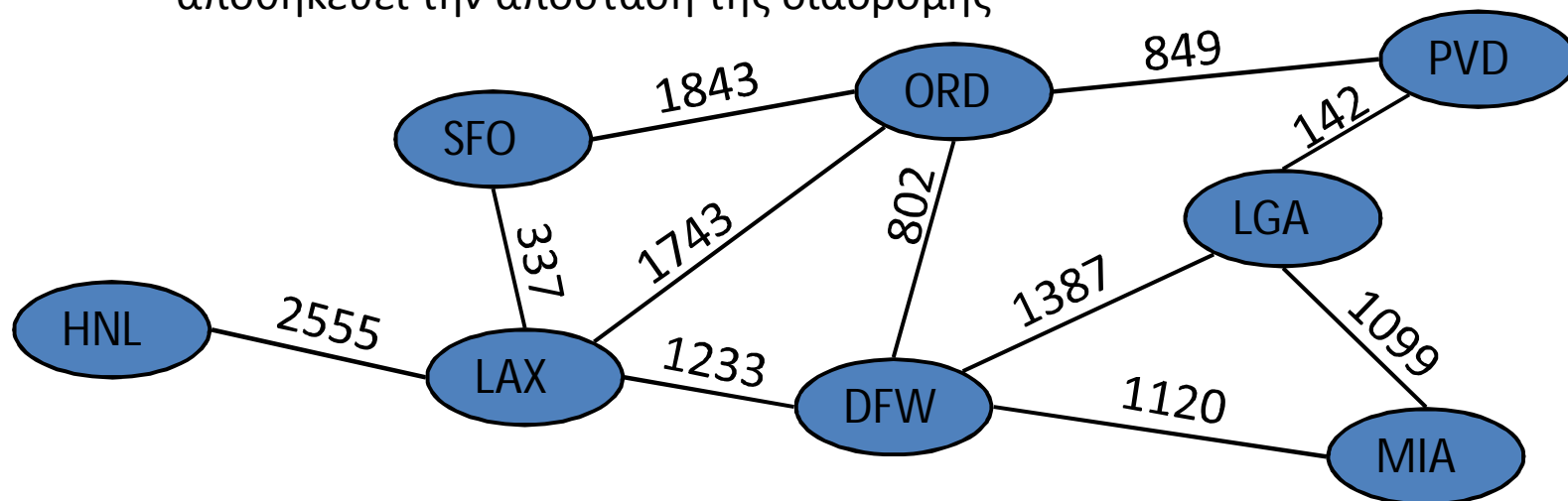
# Περιγραφή και Υλικό Ανάγνωσης

- Γράφοι (§6.1)
  - Ορισμός
  - Εφαρμογές
  - Ορολογία
  - Ιδιότητες
  - ADT
- Δομές δεδομένων για γράφους (§6.2)
  - Δομή λίστας ακμών
  - Δομή λίστας γειτνίασης
  - Δομή πίνακα γειτνίασης

Γράφοι

# Γράφος

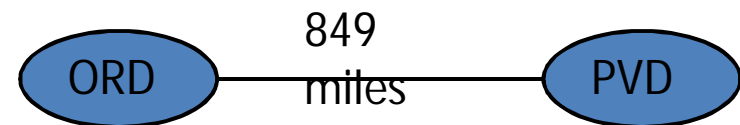
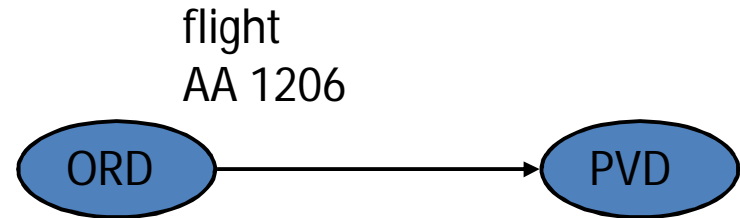
- Ένας γράφος είναι ένα ζευγάρι  $(V, E)$ , όπου
  - $V$  είναι ένα σύνολο κόμβων, ονομαζόμενων **κορυφών**
  - $E$  είναι ένα σύνολο από ζεύγη κορυφών, ονομαζόμενων **ακμών**
  - Οι κορυφές και οι ακμές είναι θέσεις και αποθηκεύουν στοιχεία
- Παράδειγμα:
  - Μια κορυφή αναπαριστά ένα αεροδρόμιο και αποθηκεύει ένα αεροπορικό κωδικό τριών γραμμάτων
  - Μια ακμή αναπαριστά μια διαδρομή μεταξύ δύο αεροδρομίων και αποθηκεύει την απόσταση της διαδρομής



Γράφοι

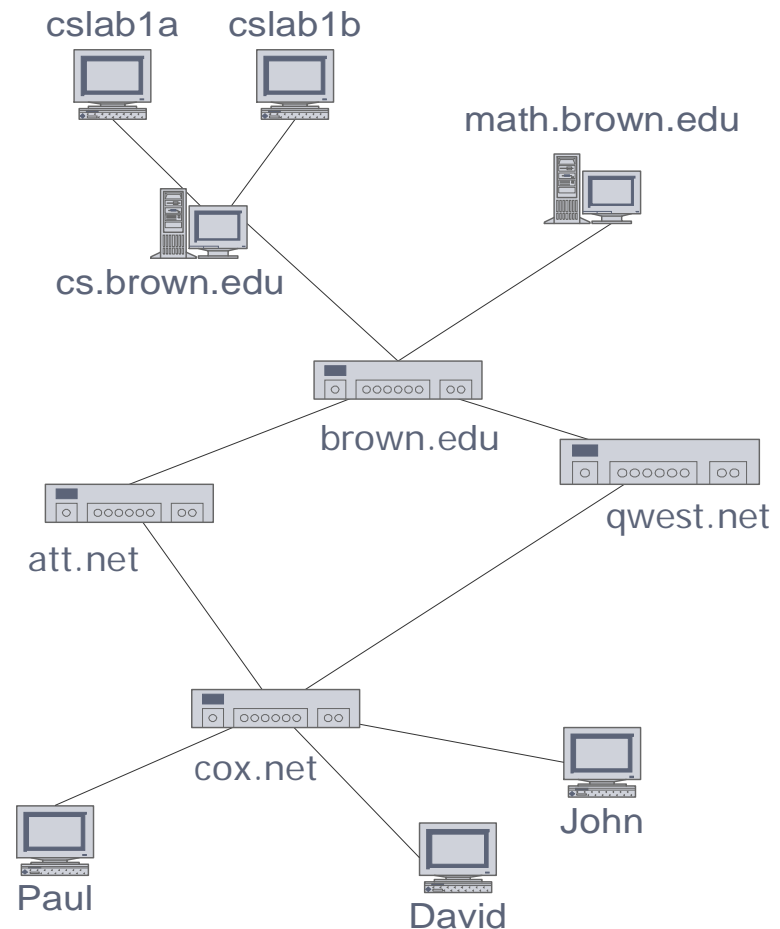
# Τύποι ακμών

- **Κατευθυνόμενες ακμές**
  - Ταξινομημένα ζευγάρια κορυφών  $(u,v)$
  - Η πρώτη κορυφή  $u$  είναι η αρχή
  - Η δεύτερη κορυφή  $v$  είναι ο προορισμός
  - π.χ., μια πτήση
- **Μη κατευθυνόμενες ακμές**
  - Μη ταξινομημένα ζευγάρια κορυφών  $(u,v)$
  - π.χ., η διαδρομή μιας πτήσης
- **Κατευθυνόμενος γράφος**
  - Όλες οι ακμές είναι κατευθυνόμενες
  - π.χ., δίκτυο πτήσεων
- **Μη κατευθυνόμενος γράφος**
  - Όλες οι ακμές είναι μη κατευθυνόμενες
  - π.χ., δίκτυο διαδρομών



# Εφαρμογές

- Ηλεκτρονικά κυκλώματα
  - Printed circuit board
  - Integrated circuit
- Δίκτυα Μεταφορών
  - Highway network
  - Flight network
- Δίκτυα Υπολογιστών
  - Local area network
  - Internet
  - Web
- Βάσεις Δεδομένων
  - Entity-relationship diagram

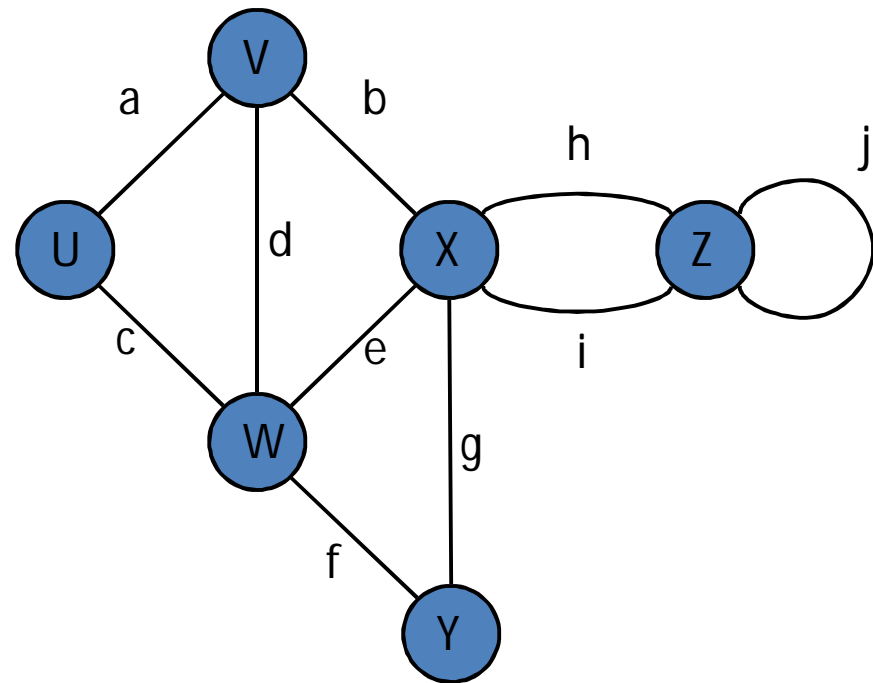


Γράφου



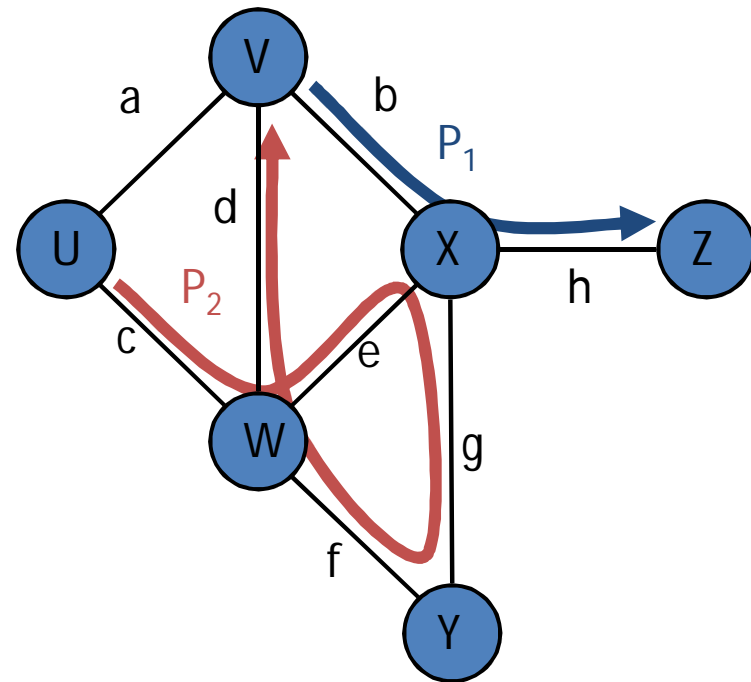
# Ορολογία

- **Τελικές κορυφές (ή τελικά σημεία (endpoints)) μιας ακμής**
  - U και V είναι τα endpoints της a
- **Ακμές επικείμενες σε μια κορυφή**
  - Οι a, d, και b είναι επικείμενες στη V
- **Γειτονικές κορυφές**
  - Οι U και V είναι γειτονικές
- **Βαθμός μιας κορυφής**
  - Η X έχει βαθμό 5
- **Παράλληλες ακμές**
  - Οι h και i είναι παράλληλες ακμές
- **Self-loop**
  - Το j είναι ένα self-loop



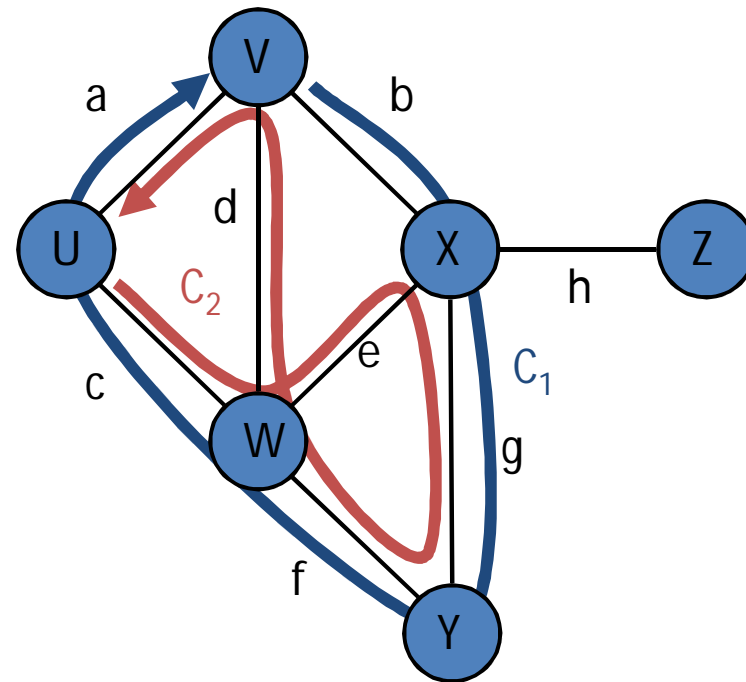
# Ορολογία (συν.)

- **Μονοπάτι**
  - Ακολουθία εναλλασόμενων κορυφών και ακμών
  - Αρχίζει με μια κορυφή
  - Τελειώνει με μια κορυφή
  - Τα endpoints κάθε ακμής προηγούνται και ακολουθούν την ακμή
- **Απλό μονοπάτι**
  - Μονοπάτι του οποίου όλες οι κορυφές και οι ακμές είναι διακριτές
- **Παράδειγματα**
  - $P_1=(V,b,X,h,Z)$  είναι ένα απλό μονοπάτι
  - $P_2=(U,c,W,e,X,g,Y,f,W,d,V)$  είναι ένα μονοπάτι που δεν είναι απλό



# Ορολογία (συν.)

- Κύκλος
  - Κυκλική ακολουθία εναλλασόμενων κορυφών και ακμών
  - Τα endpoints κάθε ακμής προηγούνται και ακολουθούν την ακμή
- Απλός κύκλος
  - Κύκλος του οποίου όλες οι κορυφές και οι ακμές είναι διακριτές
- Παραδείγματα
  - $C_1 = (V, b, X, g, Y, f, W, c, U, a, \rightarrow)$  είναι ένας απλός κύκλος
  - $C_2 = (U, c, W, e, X, g, Y, f, W, d, V, a, \rightarrow)$  είναι ένας κύκλος που δεν είναι απλός



# Ιδιότητες

## Ιδιότητα 1

$$\sum_v \text{deg}(v) = 2m$$

Απόδειξη: κάθε endpoint μετριέται δύο φορές

## Σημείωση

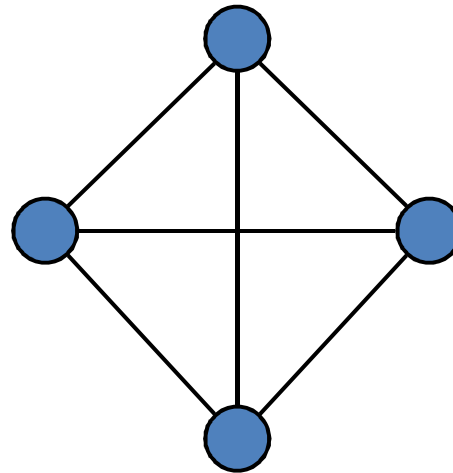
$n$  αριθμός κορυφών  
 $m$  αριθμός ακμών  
 $\text{deg}(v)$  βαθμός κορυφής  $v$

## Ιδιότητα 2

Σε ένα μη κατευθυνόμενο γράφο χωρίς self-loops και πολλαπλές ακμές (multiple edges)

$$m \leq n(n-1)/2$$

Απόδειξη: κάθε κορυφή έχει βαθμό το πολύ  $(n-1)$



## Παράδειγμα

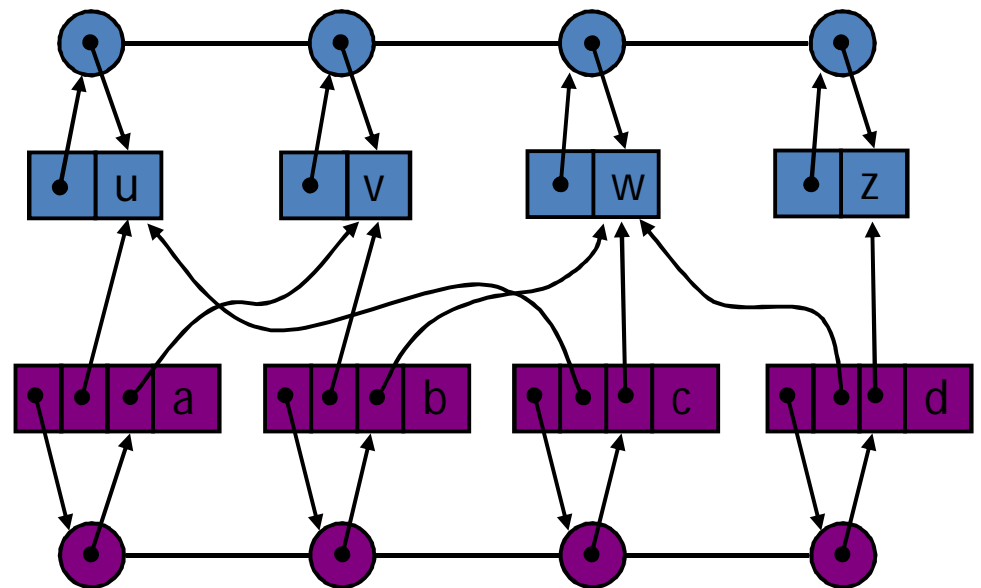
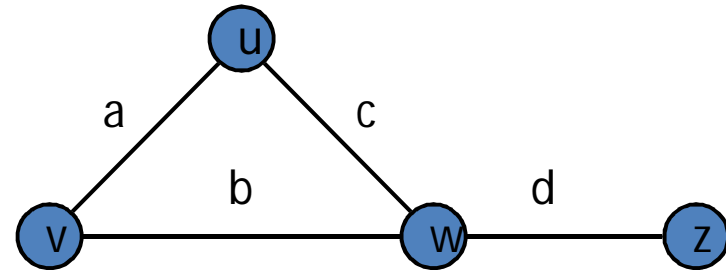
- $n = 4$
- $m = 6$
- $\text{deg}(v) = 3$

# Κύριες μέθοδοι του ΑΤΔ του Γράφου

- Κορυφές και ακμές
  - Είναι θέσεις
  - Αποθηκεύουν στοιχεία
- Accessor methods
  - `aVertex()`
  - `incidentEdges(v)`
  - `endVertices(e)`
  - `isDirected(e)`
  - `origin(e)`
  - `destination(e)`
  - `opposite(v, e)`
  - `areAdjacent(v, w)`
- Update methods
  - `insertVertex(o)`
  - `insertEdge(v, w, o)`
  - `insertDirectedEdge(v, w, o)`
  - `removeVertex(v)`
  - `removeEdge(e)`
- Generic methods
  - `numVertices()`
  - `numEdges()`
  - `vertices()`
  - `edges()`

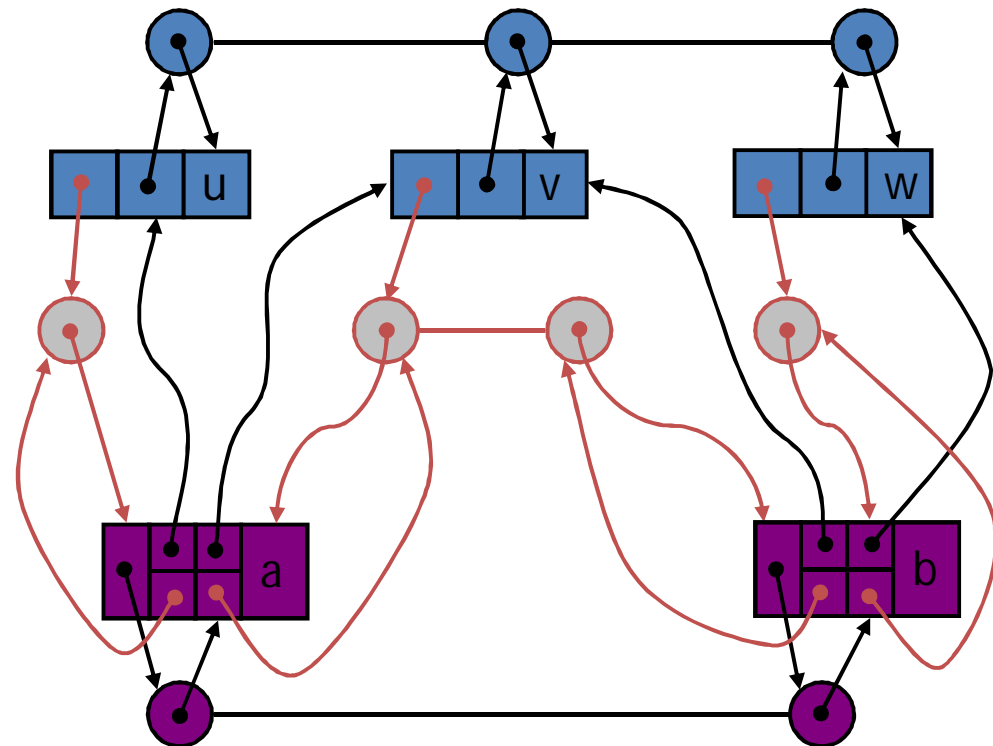
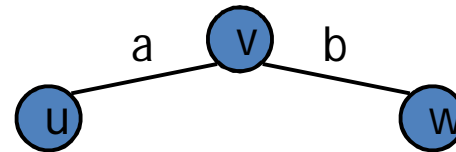
# Δομή Λίστας Ακμών

- **Αντικείμενο Κορυφής**
  - στοιχείο
  - Αναφορά στη θέση στην ακολουθία κορυφών
- **Αντικείμενο ακμής**
  - στοιχείο
  - Αρχής αντικείμενο Κορυφής
  - Προορισμός αντικείμενο Κορυφής
  - Αναφορά στη θέση στην ακολουθία ακμών
- **Ακολουθία κορυφών**
  - Ακολουθία των αντικειμένων Κορυφής
- **Ακολουθία ακμών**
  - Ακολουθία των αντικειμένων ακμών



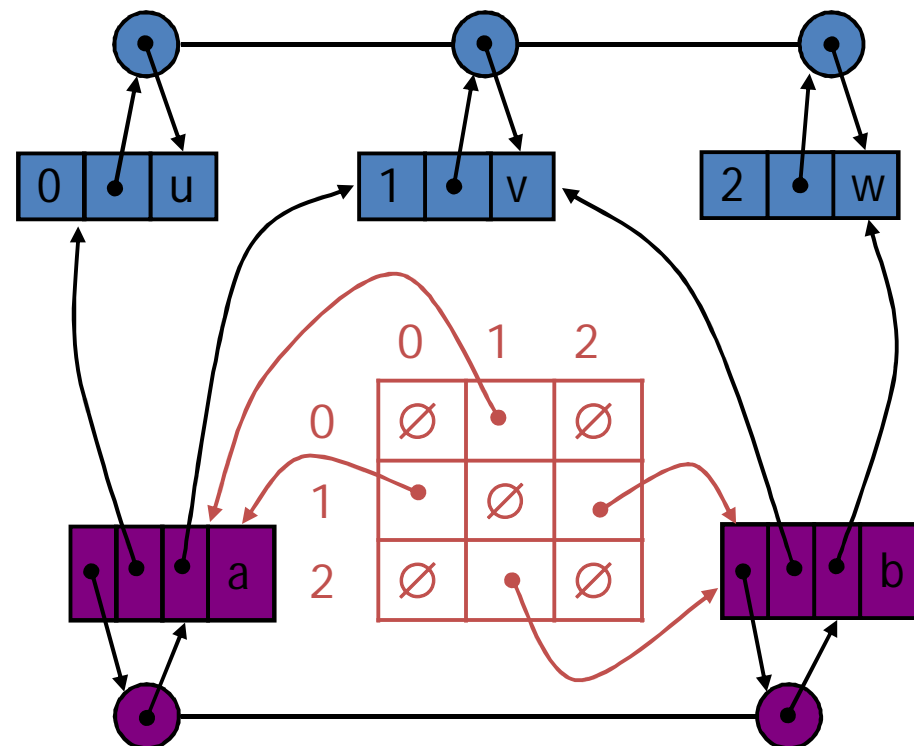
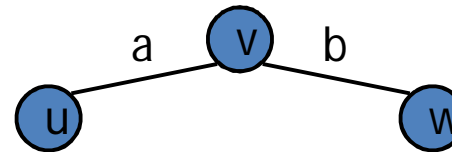
# Δομή Λίστας Γειτνίασης

- Δομή λίστας ακμών
- Ακολουθία πρόσπτωσης για κάθε κορυφή
  - Ακολουθία από αναφορές σε αντικείμενα κορυφών προσπίπτωντων ακμών
- Augmented edge objects
  - Αναφορές σε συνδεόμενες θέσεις σε ακολουθίες προσπτώσεως για τελικές κορυφές



# Δομή Πίνακα Γειτνίασης

- Δομή λίστας ακμών
- Augmented vertex objects
  - Ακέραιο κλειδί (index) συνδεδεμένο με κάθε κορυφή
- Διδιάστατος πίνακας γειτνίασης
  - Αναφορά σε αντικείμενα ακμών για γειτονικές κορυφές
  - Null για μη γειτονικές κορυφές





# Απόδοση

<ul style="list-style-type: none"> <li>◆ <math>n</math> κορυφές</li> <li>◆ <math>m</math> ακμές</li> <li>◆ χωρίς παράλληλες ακμές</li> <li>◆ χωρίς self-loops</li> </ul>	Edge List	Adjacency List	Adjacency Matrix
Space	$n + m$	$n + m$	$n^2$
incidentEdges( $v$ )	$m$	deg( $v$ )	$n$
areAdjacent ( $v, w$ )	$m$	min(deg( $v$ ), deg( $w$ ))	1
insertVertex( $o$ )	1	1	$n^2$
insertEdge( $v, w, o$ )	1	1	1
removeVertex( $v$ )	$m$	deg( $v$ )	$n^2$
removeEdge( $e$ )	1	1	1

# Τέλος Ενότητας



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης