Βιοπληροφορική

Ενότητα 6: Σύγκριση αλληλουχιών – Part I

Αν. καθηγητής Αγγελίδης Παντελής
e-mail: paggelidis@uowm.gr
ΕΕΔΙΠ Μπέλλου Σοφία
e-mail: sbellou@uowm.gr

Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών

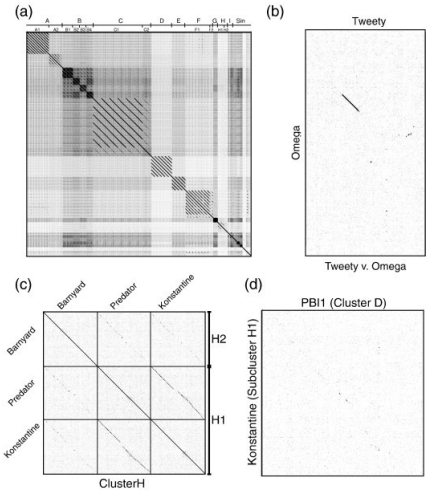
Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ψηφιακά Μαθήματα στο Πανεπιστήμιο Δυτικής Μακεδονίας**» 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.

Σύγκριση αλληλουχιών – Part I



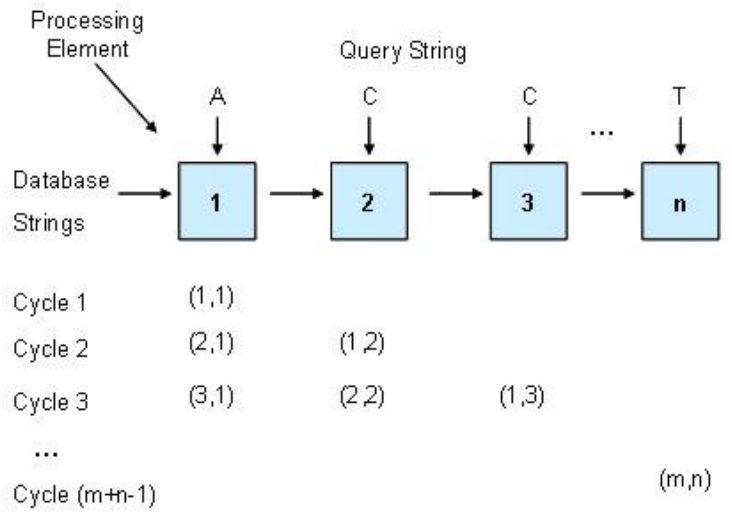
		A	T	T	C	G	T	A	C	T	T	A	G	T
	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12	-13
C	-1	-1	-2	-3	-2	-4	-6	-7	-8	-9	-10	-11	-12	-13
T	-2	-2	0	0	-2	-3	-3	-5	-7	-7	-7	-9	-11	-11
T	-3	-3	0	1	-1	-3	-2	-4	-6	-6	-6	-8	-10	-10
A	-4	-2	-2	-1	0	-2	-4	-1	-3	-5	-7	-5	-7	-9
G	-5	-4	-3	-3	-2	1	-1	-3	-2	-4	-6	-8	-4	-6
C	-6	-6	-5	-4	-2	-1	0	-2	-2	-3	-5	-7	-6	-5
T	-7	-7	-5	-4	-4	-3	0	-1	-3	-1	-1	-3	-5	-5
A	-8	-6	-7	-6	-5	-5	-2	1	-1	-3	-2	0	-2	-4
A	-9	-6	-7	-8	-7	-6	-4	1	0	-2	-4	0	-1	-3
T	-10	-8	-5	-5	-7	-8	-4	-1	0	1	1	-1	-1	0
C	-11	-10	-7	-6	-4	-6	-6	-3	0	-1	0	0	-2	-2
A	-12	-10	-9	-8	-6	-5	-7	-3	-2	-1	-2	1	-1	-3
G	-13	-12	-11	-10	-8	-5	-6	-5	-4	-3	-2	-1	2	0

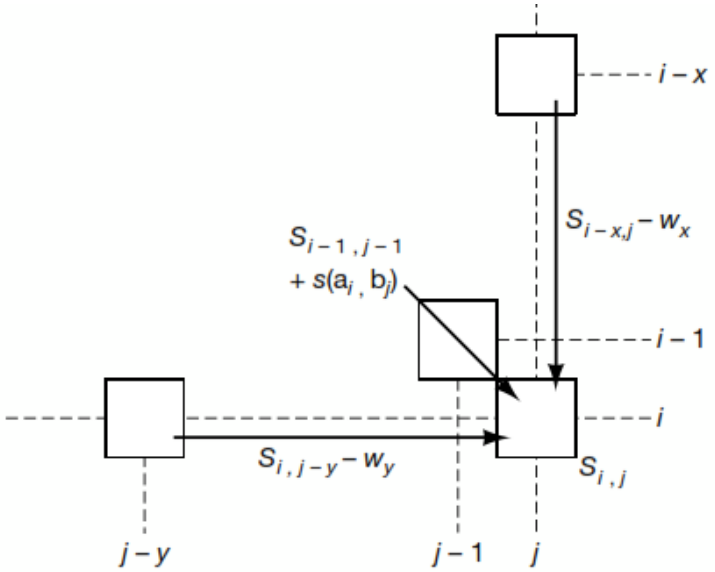
So the best alignment would be:

-- = gap
| = match

ATTCG--TACTTAGT
CTTAGCTAATCAG--

Αλγόριθμοι δυναμικού προγραμματισμού (1/2)





Βιβλιογραφία

- Βιοπληροφορική – Δυνατότητες και προοπτικές, Σοφία Κοσσίδα.
- Εισαγωγή στους αλγόριθμους Βιοπληροφορικής, Neil C. Jones & Pavel A. Pevzner.
- Bioinformatics: Sequence and Genome Analysis, David W. Mount.
- Bioinformatics Computing, Bryan Bergeron.

Μέθοδοι στοίχισης

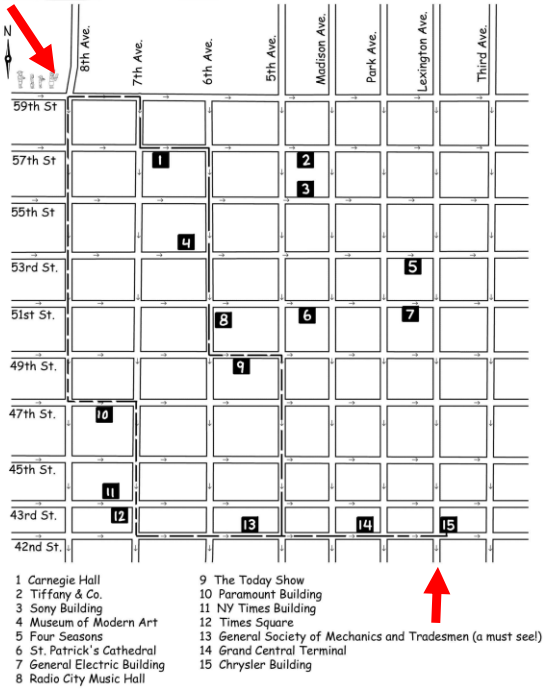
1. Οπτικοί.
2. Με αλγόριθμους δυναμικού προγραμματισμού.
3. Με ευρετικούς αλγόριθμους, που βασίζονται στην έννοια των «λέξεων».

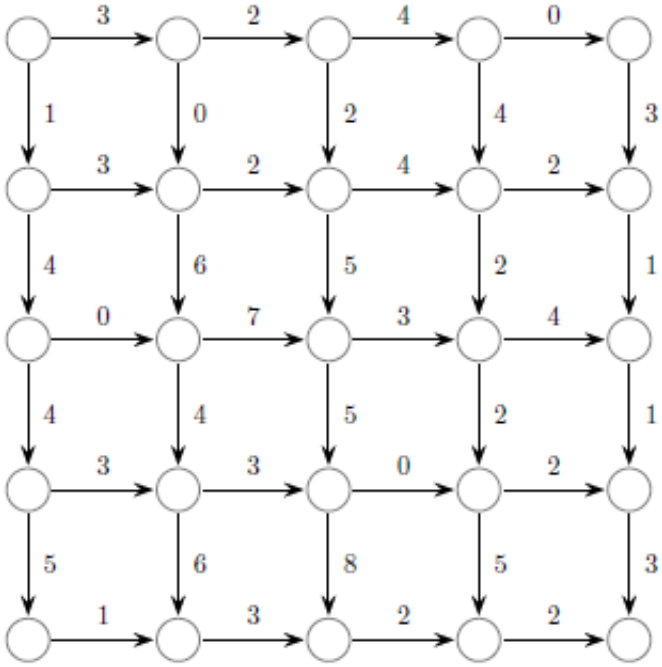
- Πρέπει να λαμβάνεται υπόψη ο **λόγος για τον οποίο γίνεται η στοίχιση αλληλουχιών.**
- Ανάλογα επιλέγεται και η μέθοδος σύγκρισης.
- **Ομοιότητα της τάξης το 90%:** Εύκολη κατασκευή της στοίχισης με όλους τους αλγόριθμους.
- **Ομοιότητα της τάξης του 25% (ζώνη του λυκόφωτος, «twilight zone of sequence alignment»):** Οριακή για ασφαλή εξαγωγή συμπερασμάτων.

Αλγόριθμοι δυναμικού προγραμματισμού (2/2)

- Συγκρίνουν δύο αλληλουχίες.
- Βασίζονται σε συστήματα βαθμονόμησης.
- Όλες οι δυνατές στοιχίσεις δύο αλληλουχιών αναπαριστώνται ως διαδρομές.
- Πίνακας βαθμολογιών όλων των δυνατών στοιχίσεων δύο αλληλουχιών.
- Προτιμάται η «διαδρομή» με το καλύτερο σκορ.

Το πρόβλημα με τους τουρίστες του Μανχάταν (1/5)

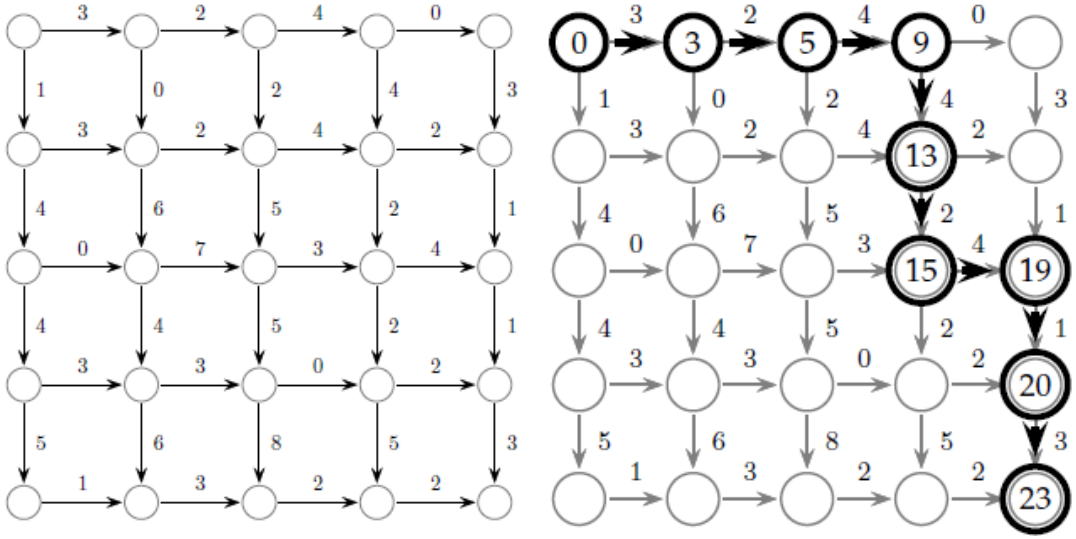




- **Σκοπός 1:** Οι τουρίστες να δουν όσο το δυνατόν περισσότερα αξιοθέατα
- **Συντελεστές στάθμισης (weights):** Πόσα αξιοθέατα υπάρχουν σε κάθε τετράγωνο

Το πρόβλημα με τους τουρίστες του Μανχάταν (2/5)

- **Σκοπός:** Οι τουρίστες πρέπει να επιλέξουν μία διαδρομή από το **σημείο προέλευσης** προς το **σημείο απόληξης.**



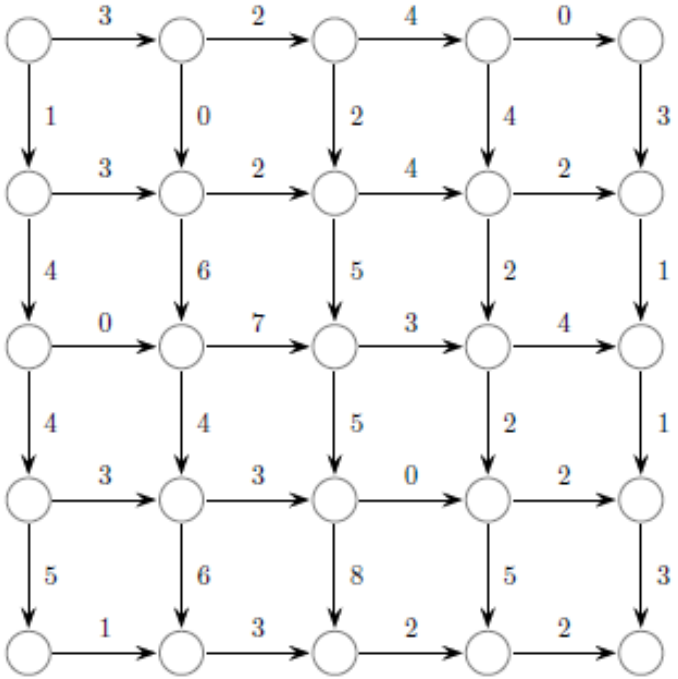
- **Κορυφή προέλευσης** (source vertex) & **κορυφή απόληξης** (sink vertex).
- **Διασταυρώσεις:** Κορυφές & **Οδοί:** Ακμές**.**
- **Συντελεστής στάθμισης κάθε διαδρομής:** Άθροισμα συντελεστών στάθμισης των ακμών της (ο συνολικός αριθμός των αξιοθέατων).

Το πρόβλημα με τους τουρίστες του Μανχάταν (3/5)

- Η κορυφή προέλευσης βρίσκεται στο (0,0) και η κορυφή απόληξης βρίσκεται στο (m,n).
- **Μέθοδος ωμής βίας:** Η καλύτερη διαδρομή μεταξύ όλων των διαδρομών στο πλέγμα -> Αδύνατο ακόμη και για μέτρια πλέγματα.
- **Γενικό πρόβλημα:** Η εύρεση της μεγαλύτερης διαδρομής **από την κορυφή προέλευσης** προς κάποια **τυχαία κορυφή (i, j),** όπου **$0 \leq i \leq n$ & $0 \leq j \leq m$.**
- **Μήκος καλύτερης διαδρομής: $s_{i,j}$.**
- **Προσοχή: ο συντελεστής στάθμισης $s_{n,m}$ είναι η λύση του προβλήματος με τους τουρίστες του Μανχάταν.**

Το πρόβλημα με τους τουρίστες του Μανχάταν (4/5)

- **Πρόβλημα με τους τουρίστες του Μανχάταν:** Μοναδικό ερώτημα. Ποιος είναι ο καλύτερος τρόπος μετάβασης από την κορυφή προέλευσης στην κορυφή απόληξης.
- **Γενικό πρόβλημα: n x m διαφορετικά ερωτήματα.** Ποιος είναι ο καλύτερος τρόπος μετάβασης από την κορυφή προέλευσης σε οποιοδήποτε σημείο.



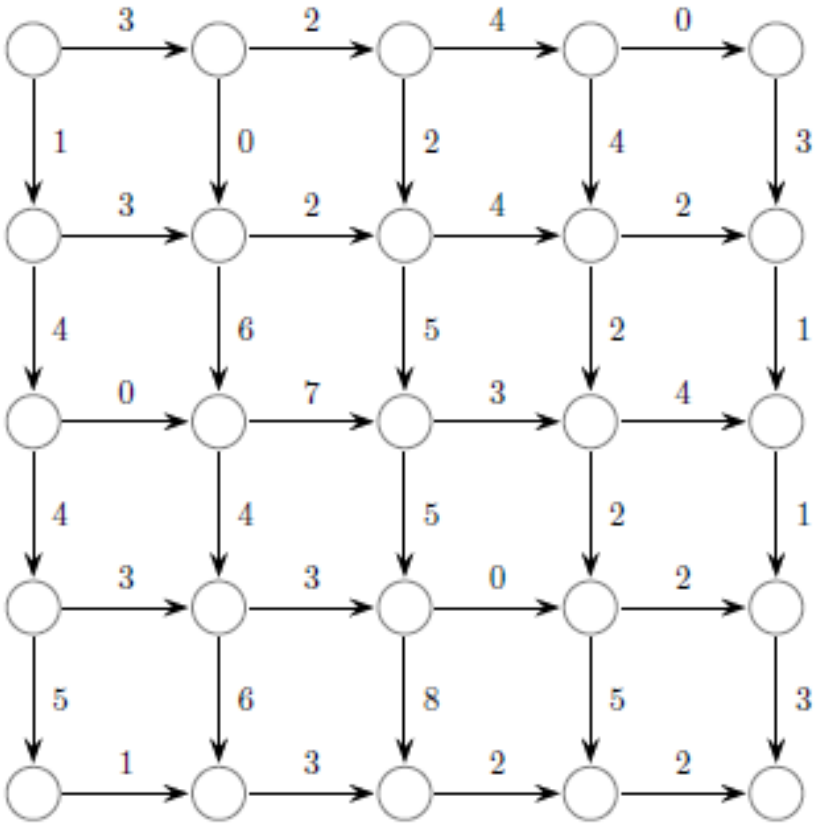
Το πρόβλημα με τους τουρίστες του Μανχάταν (5/5)

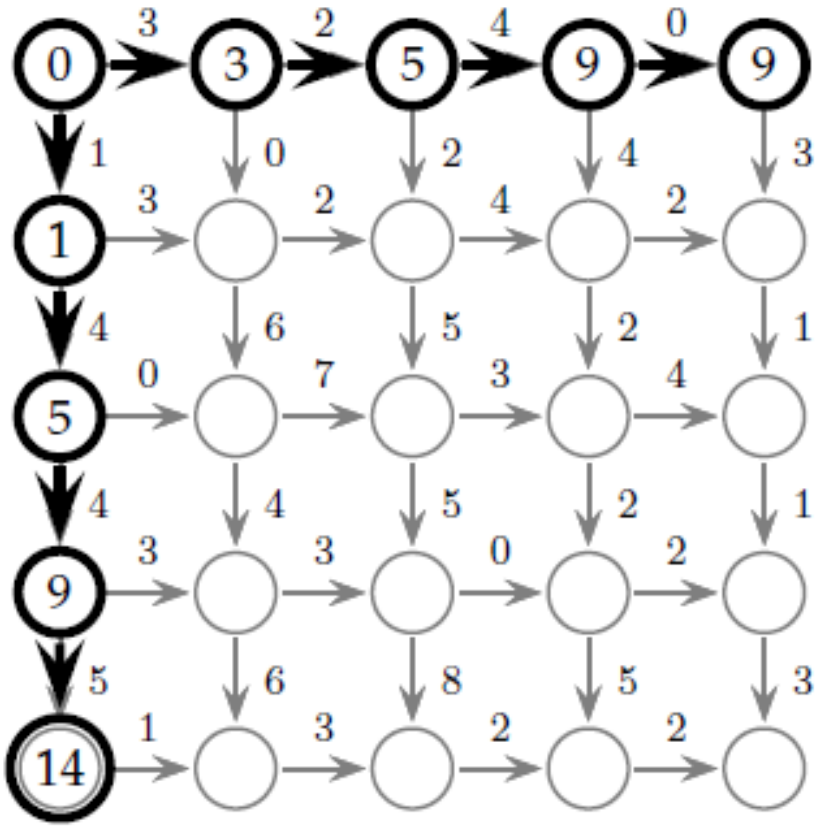
- Φαίνεται ότι έχουμε δημιουργήσει mxn προβλήματα (τον υπολογισμό της κορυφής (i,j) με $0 \le i \le n$ και $0 \le j \le m$, αντί
- 1 πρόβλημα (τον υπολογισμό του $s_{n,m}$).

Η ΒΑΣΗ ΤΟΥ ΔΥΝΑΜΙΚΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ =

Η ΕΠΙΛΥΣΗ ΤΟΥ ΓΕΝΙΚΟΥ ΠΡΟΒΛΗΜΑΤΟΣ ΕΙΝΑΙ ΤΟ ΙΔΙΟ ΕΥΚΟΛΗ ΜΕ ΤΗΝ ΕΠΙΛΥΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ ΜΕ ΤΟΥΣ ΤΟΥΡΙΣΤΕΣ ΤΟΥ ΜΑΝΧΑΤΑΝ.

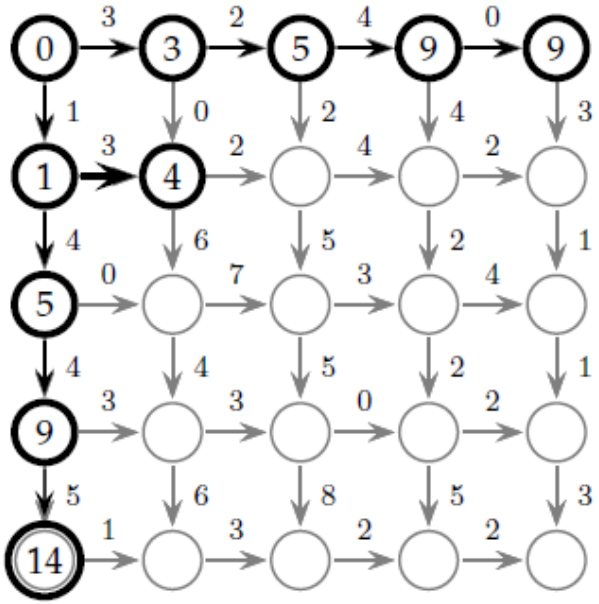
Εύρεση των $s_{0,j}$ $(0 \le j \le m)$ και $s_{i,0}$ $(0 \le i \le n)$

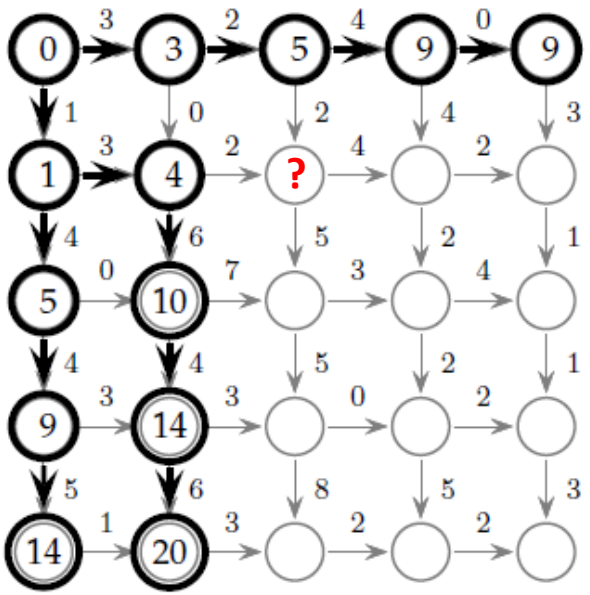




Υπολογισμός του $s_{1,1}$

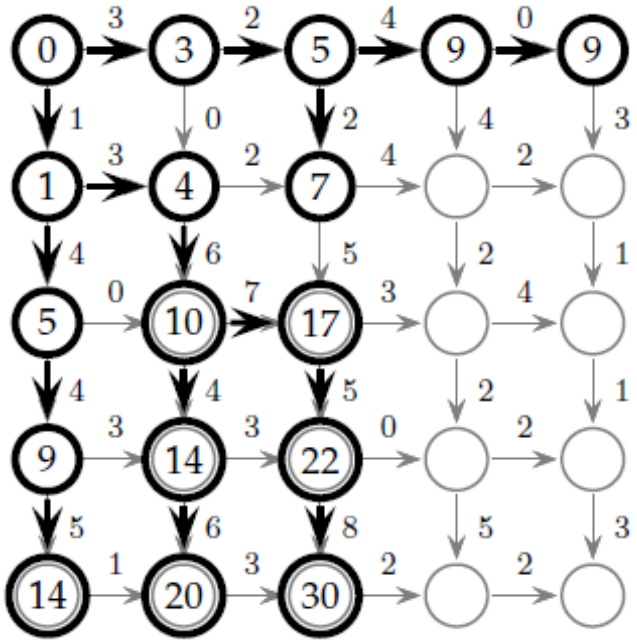
- Θέση $\mathbf{s_{1,1}}$.
- **1ος** τρόπος: Νότια από τη θέση (0,1), **2ος** τρόπος**:** Ανατολικά από τη θέση (1,0)**.**

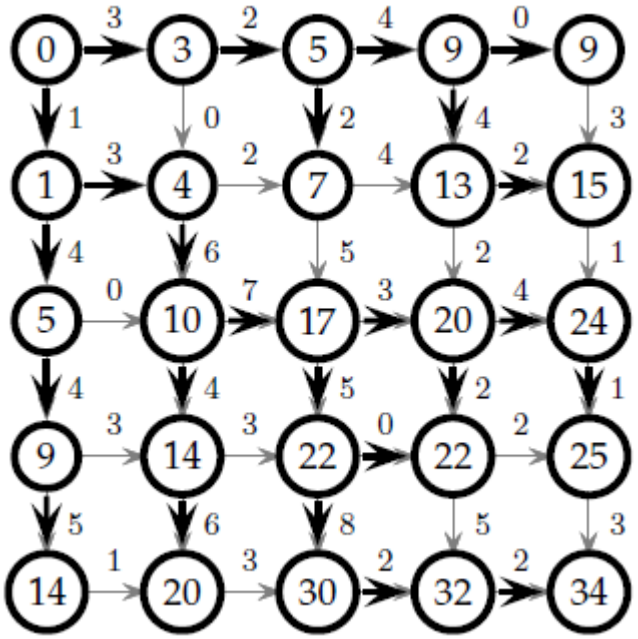




$$s_{1,2} = \max \begin{cases} s_{1,1}\text{+συντελεστής βαρύτητας της ακμής μεταξύ (1,1) και (1,2)} \\ s_{0,2}\text{+συντελεστής βαρύτητας της ακμής μεταξύ (0,2) και (1,2)} \end{cases}$$

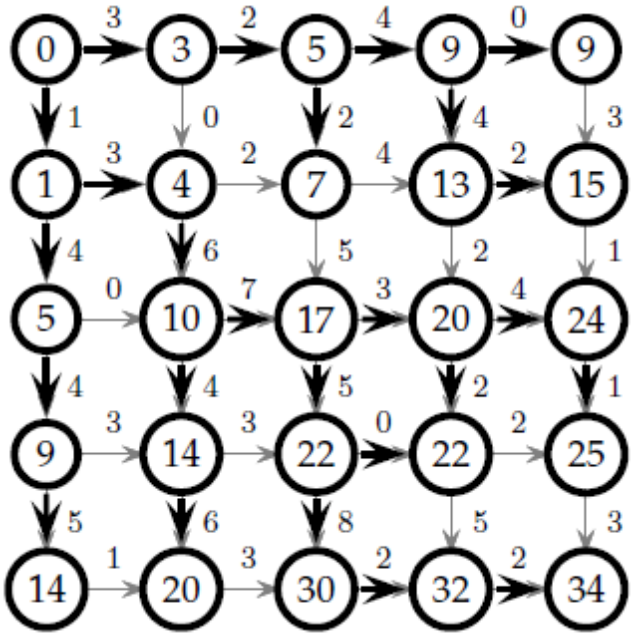
Υπολογισμός του $s_{i,j}$ (1/2)





$$s_{1,j} = \max \begin{cases} s_{i-1,j}\text{+συντελεστής βαρύτητας της ακμής μεταξύ (i-1,j) και (I,j)} \\ s_{i,j-1}\text{+συντελεστής βαρύτητας της ακμής μεταξύ (i,j-1) και (I,j)} \end{cases}$$

Υπολογισμός του $s_{i,j}$ (2/2)



MANHATTANTOURIST

1 $s_{0,0} \leftarrow 0$

2 for i ← 1 to n

3 $s_{i,0} \leftarrow s_{i-1,0} + w_{i,0}$ (↓)

4 for j ← 1 to m

5 $s_{0,j} \leftarrow s_{0,j-1} + w_{0,j}$ (→)

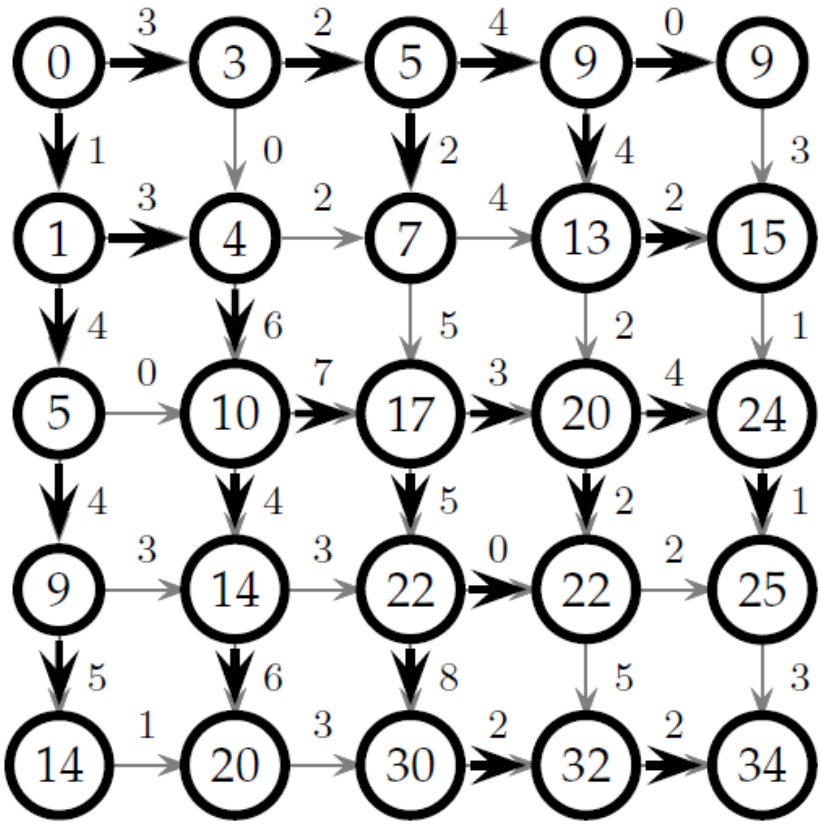
6 for i ← 1 to n

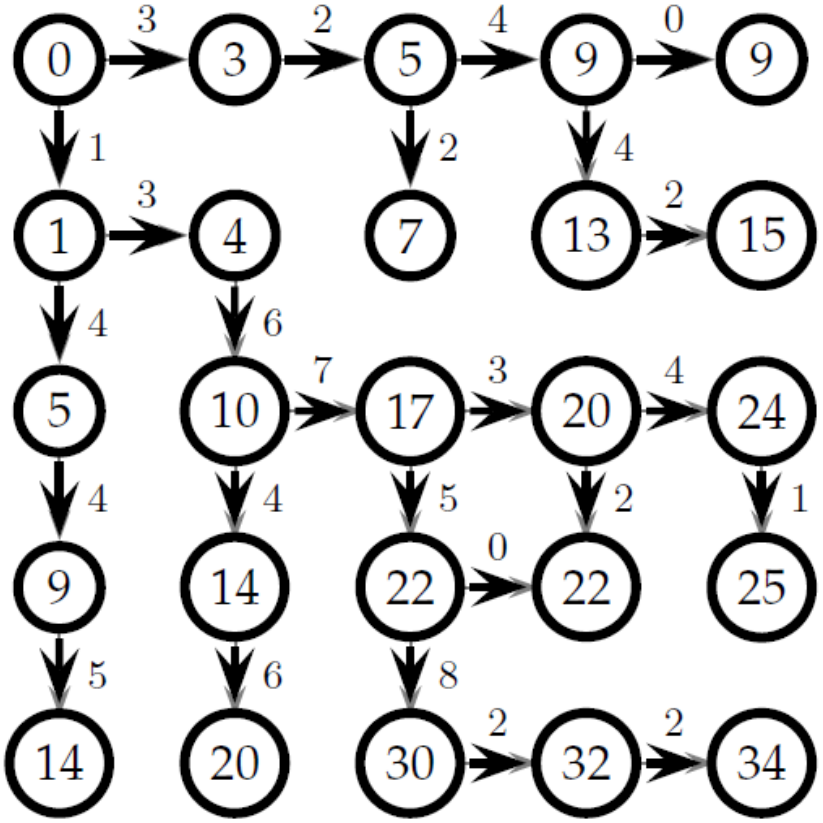
7 for j ← 1 to m

8 $s_{i,j} = \max \begin{cases} s_{i-1,j} + w_{i,j} \ (\downarrow) \\ s_{i,j-1} + w_{i,j} \ (\rightarrow) \end{cases}$

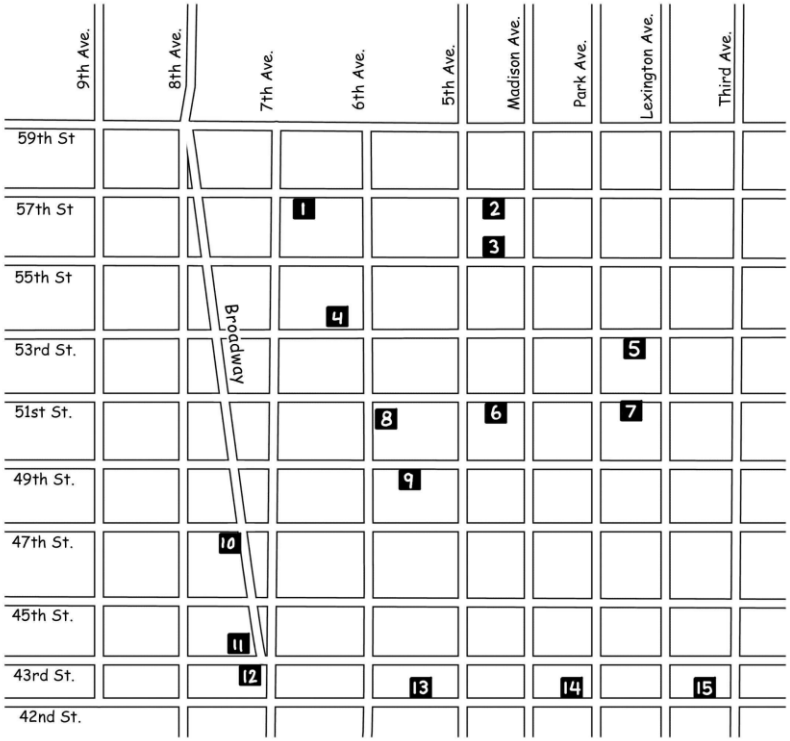
$$s_{1,j} = \max \begin{cases} s_{i-1,j} + \text{συντελεστής βαρύτητας της ακμής μεταξύ } (i-1,j) \text{ και } (l,j) \\ s_{i,j-1} + \text{συντελεστής βαρύτητας της ακμής μεταξύ } (i,j-1) \text{ και } (l,j) \end{cases}$$

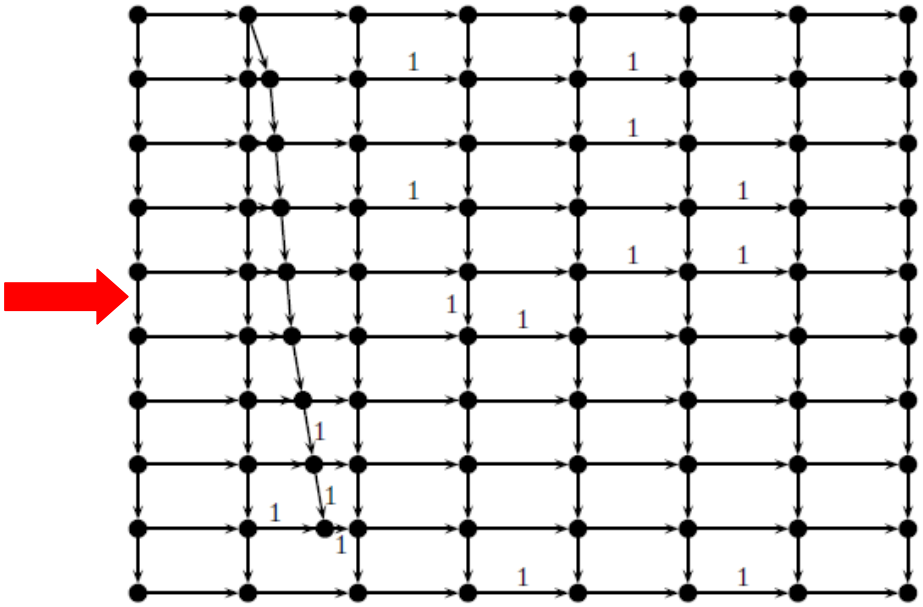
Οι πιθανές διαδρομές





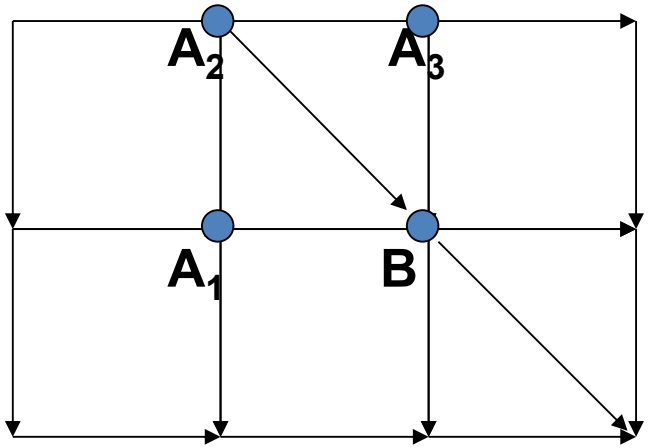
Κατευθυνόμενα ακυκλικά γραφήματα
Directed acyclic graphs, DAG (1/3)





Γράφημα G = (V,E), όπου V: κορυφές και E: ακμές του γραφήματος

Manhattan Is Not A Perfect Grid

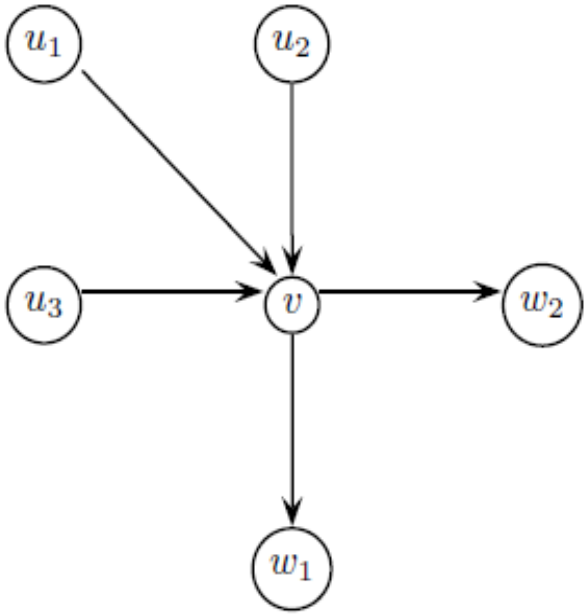


What about diagonals?

- The score at point B is given by:

$$s_B = \textbf{max of} \begin{cases} s_{A1} + \text{weight of the edge } (A_1, B) \\ s_{A2} + \text{weight of the edge } (A_2, B) \\ s_{A3} + \text{weight of the edge } (A_3, B) \end{cases}$$

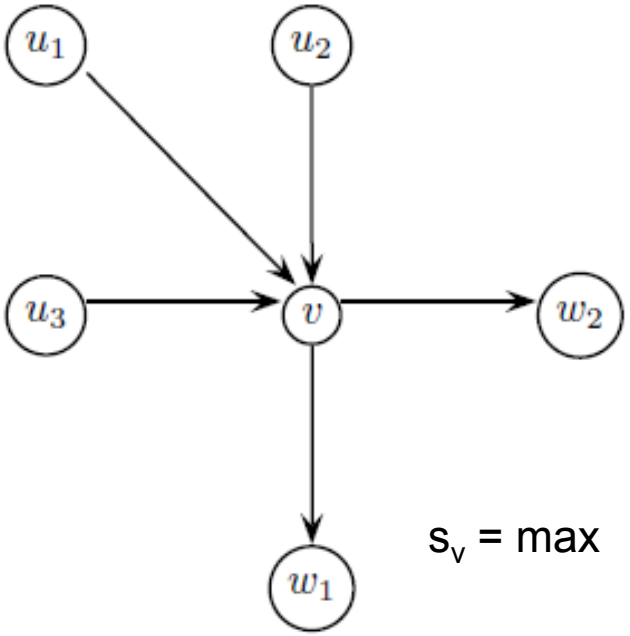
Κατευθυνόμενα ακυκλικά γραφήματα Directed acyclic graphs, DAG (2/3)



- Μία ακμή του γραφήματος G μπορεί να οριστεί σε σχέση με την κορυφή προέλευσης της u και την κορυφή προορισμού της v ως (u,v).
- **Εισερχόμενος βαθμός κορυφής:** Ο αριθμός των εισερχόμενων ακμών μιας κορυφής – πρόγονοι.
- **Εξερχόμενος βαθμός κορυφής:** Ο αριθμός των εξερχόμενων ακμών μιας κορυφής - απόγονοι
- u: πρόγονος (predecessor) της κορυφής v αν (u,v) ε E.

Γράφημα G = (V,E), όπου V: κορυφές και E: ακμές του γραφήματος

Κατευθυνόμενα ακυκλικά γραφήματα Directed acyclic graphs, DAG (3/3)



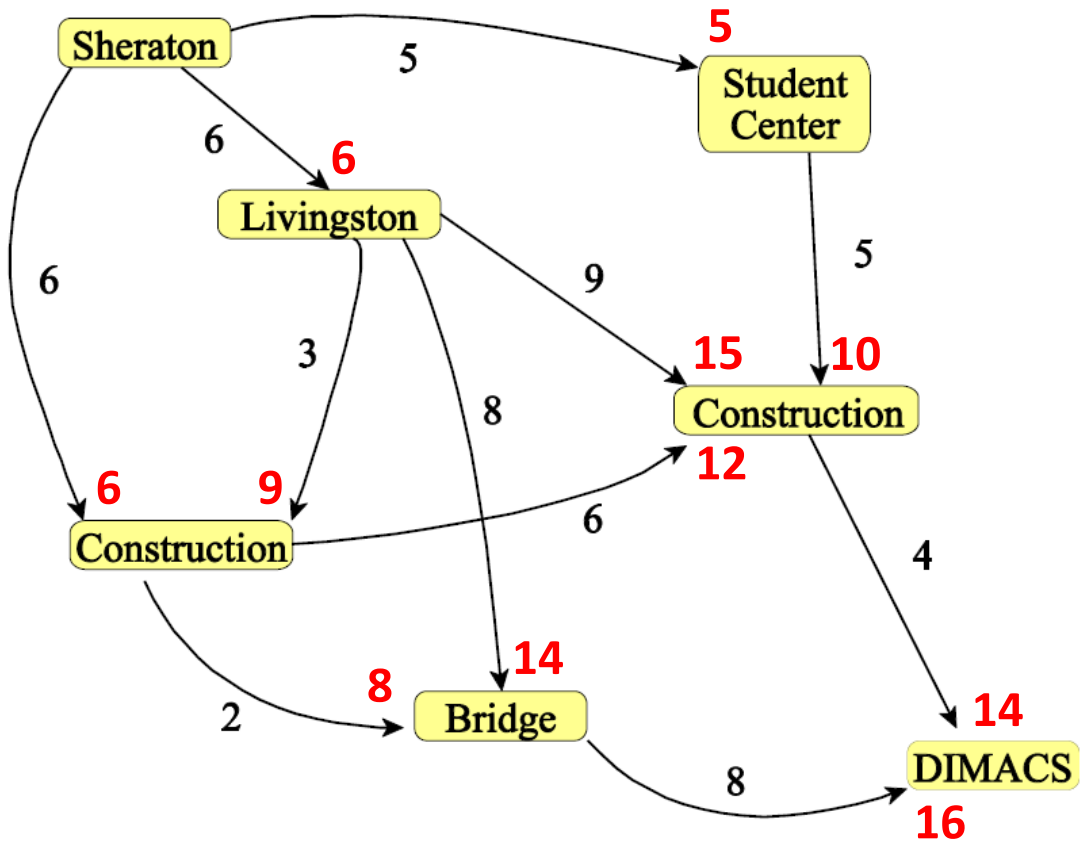
- Έστω κορυφή v με εισερχόμενο βαθμό 3 και σύνολο προγόνων {u1, u2, u3}.
- Η μεγαλύτερη διαδρομή προς τη συγκεκριμένη κορυφή είναι:

$s_v = \max$ { s_{u1} + συντελεστής βαρύτητας της ακμής μεταξύ u_1 και v ; s_{u2} + συντελεστής βαρύτητας της ακμής μεταξύ u_2 και v ; s_{u3} + συντελεστής βαρύτητας της ακμής μεταξύ u_3 και v }

ΓΕΝΙΚΑ η βαθμολογία S_v μιας κορυφής δίνεται από τη σχέση

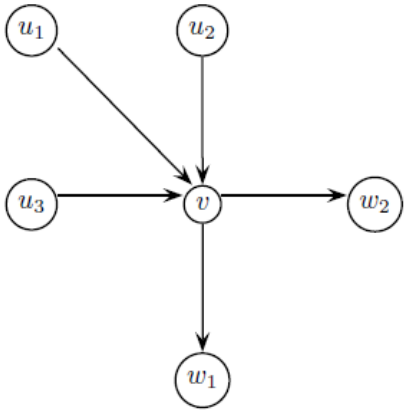
S_v = max (S_u + συντελεστής βαρύτητας της ακμής μεταξύ u και v)
u∈Predecessors(v)

Το πιο σύντομο μονοπάτι από το Sheraton στο DIMACS



Μήκος του πιο σύντομο μονοπατιού = 14km

Traveling in the Grid



- Το μοναδικό εμπόδιο είναι ότι πρέπει να αποφασίσετε τη σειρά με την οποία θα περάσετε από τις κορυφές κατά τον υπολογισμό του s.
- Η σειρά είναι σημαντική, αφού μέχρι να αναλυθεί η κορυφή v, πρέπει να έχουν υπολογιστεί οι τιμές s_u για όλους τους προγόνους τους.
- Θα πρέπει να διασχίσουμε τις κορυφές με μία σειρά.
- Πώς μπορούμε να βρούμε αυτή τη σειρά για συγκεκριμένη διάταξη;;;

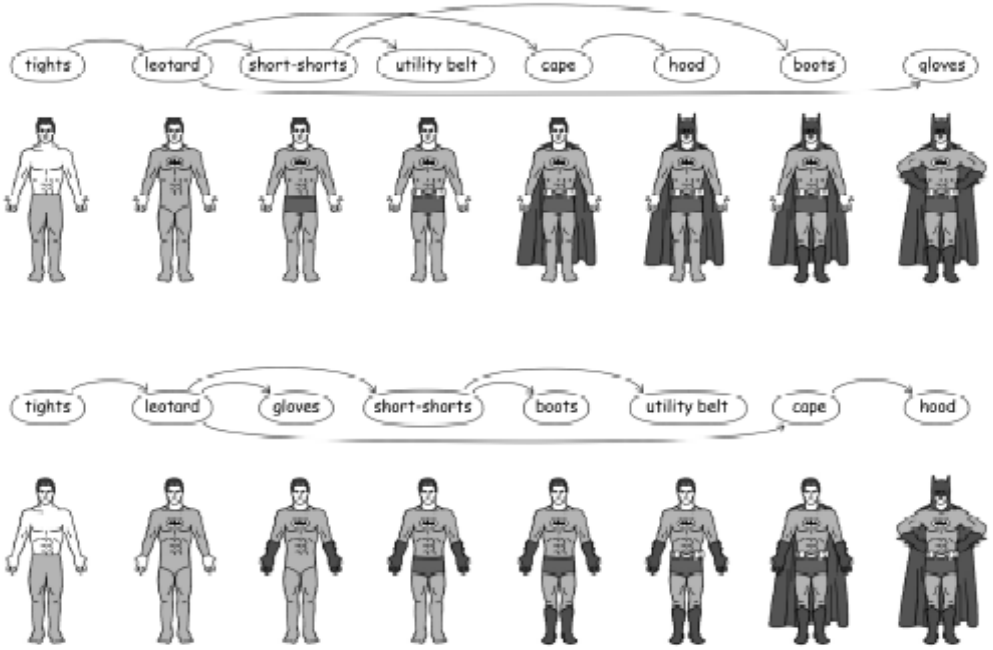
Dressing in the morning problem as DAG

- Αφού το Μανχάταν δεν είναι ένα τέλειο πλέγμα, μπορούμε να το αναπαραστήσουμε ως ένα κατευθυνόμενο ακυκλικό διάγραμμα – directed acyclic graph, DAG.
- **Παράδειγμα:** Με τον ίδιο τρόπο μπορούμε να αναπαραστήσουμε και το πρόβλημα του πρωινού ντυσίματος.

Τοπολογικές διατάξεις - Topological ordering

- Οι σειρά με την οποία διατρέχονται οι κορυφές ενός γραφήματος ονομάζονται **τοπολογικές διατάξεις** του κατευθυνόμενου ακυκλικού διαγράμματος – DAG.
- **Τοπολογική διάταξη του DAG**: Κάθε ακμή (v_i,v_j) μιας διάταξης κορυφών v_1…v_n συνδέει μία κορυφή μικρότερου δείκτη με κορυφή μεγαλύτερου δείκτη.
- Επομένως, αν οι κορυφές μπορούν να τοποθετηθούν σε μία σειρά σύμφωνα με αυξανόμενη τιμή δείκτη, τότε όλες οι ακμές πηγαίνουν από αριστερά προς τα δεξιά.

Dressing in the morning problem as DAG



- Υπάρχουν διαφορετικοί τρόποι για να ντυθεί κάποιος το πρωί.
- Ο κάθε τρόπος αντιστοιχεί σε μία διαφορετική τοπολογική διάταξη.

Longest Path in DAG Problem

- **Goal:** Find a longest path between two vertices in a weighted DAG.
- **Input:** A weighted DAG ***G*** with source and sink vertices.
- **Output:** A longest path in ***G*** from source to sink.

Longest Path in DAG: Dynamic Programming

- Suppose vertex *v* has in degree 3 and predecessors $\{u_1, u_2, u_3\}$
- Longest path to *v* from source is:

$$s_v = \textbf{max of} \begin{cases} s_{u_1} + \text{weight of edge from } u_1 \text{ to } v \\ s_{u_2} + \text{weight of edge from } u_2 \text{ to } v \\ s_{u_3} + \text{weight of edge from } u_3 \text{ to } v \end{cases}$$

In General:

$s_v = \max_u \ (s_u + \text{weight of edge from } \boldsymbol{u} \text{ to } \boldsymbol{v})$

Απόσταση μετασχηματισμού και στοιχίσεις (1/3)

Συμβολοσειρά **v** (n χαρακτήρες): **ATGTTAT**

Συμβολοσειρά **w** (m χαρακτήρες): **ATCGTAC**

A	T	-	G	T	T	A	T	-
A	T	C	G	T	-	A	-	C

5 matches

2 insertions

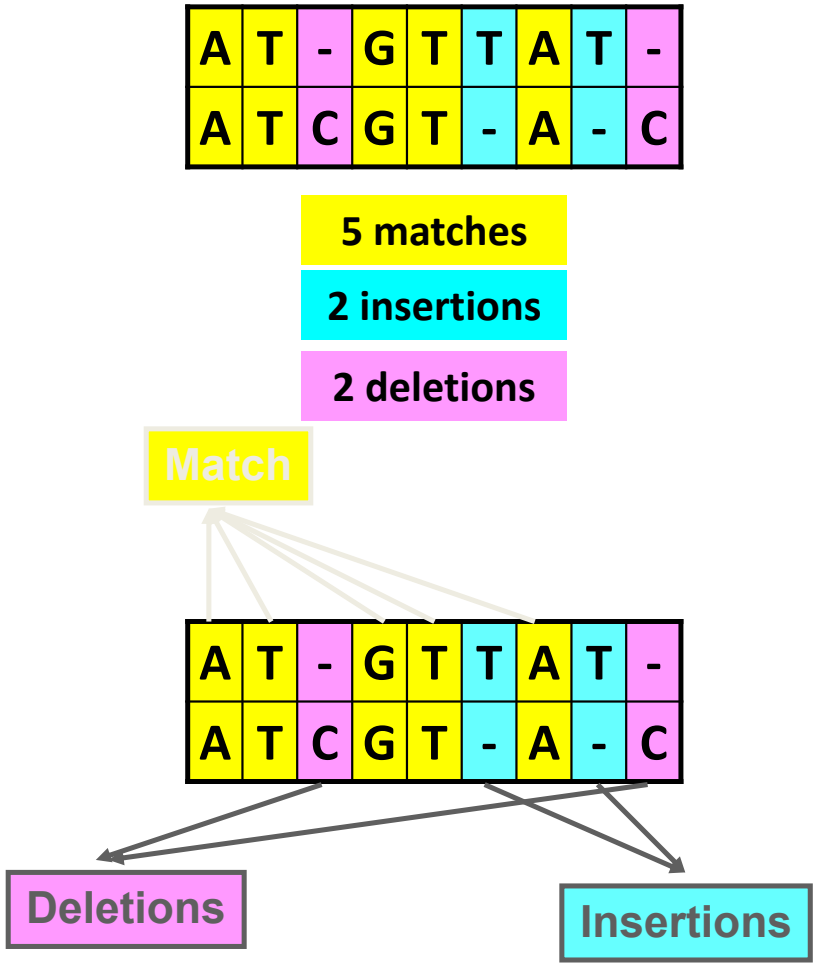
2 deletions

ΜΗΤΡΑ ΣΤΟΙΧΙΣΗΣ

Η στοίχιση των συμβολοσειρών v και w είναι μία **μήτρα** με δύο γραμμές, οι οποίες περιέχουν διατεταγμένους σε σειρά τους χαρακτήρες των συμβολοσειρών

Καμία στήλη της μήτρας στοίχισης δεν περιέχει κενά διαστήματα και στις δύο γραμμές, έτσι ώστε η στοίχιση να έχει το πολύ m+n στήλες.

Απόσταση μετασχηματισμού και στοιχίσεις (2/3)



Απόσταση μετασχηματισμού και στοιχίσεις (3/3)

A	T	-	G	T	T	A	T	-
A	T	C	G	T	-	A	-	C

- **Στήλες με ίδιο γράμμα** = ταίριασμα (match).
- **Στήλες με διαφορετικό γράμμα** = ασυμφωνία (mismatch).
- **Στήλες με κενό** = Πρόσθεση ή αφαίρεση στοιχείου (insertion – deletion).
- **Απόσταση μετασχηματισμού (edit distance):** 5 ταιριάσματα + 0 ασυμφωνίες + 4 προσθέσεις/αφαιρέσεις = **9 (m+n).**
- **Αναπαράσταση 1:**
  - **AT-GTTAT-** αναπαριστά τη γραμμή που αντιστοιχεί στην **v**=**ATGTTAT.**
  - **ATCGT-A-C** αναπαριστά τη γραμμή που αντιστοιχεί στην **w**= **ATCGTAC.**

Πλέγμα στοίχισης συμβολοσειρών

Αναπαράσταση 2:

AT- GTTAT- → 122345677

ATCGT- A-C → 123455667

A	T	-	G	T	T	A	T	-
A	T	C	G	T	-	A	-	C

$$\begin{pmatrix} 0 & 1 & 2 & 2 & 3 & 4 & 5 & 6 & 7 & 7 \\ 0 & 1 & 2 & 3 & 4 & 5 & 5 & 6 & 6 & 7 \end{pmatrix}$$

ΜΗΤΡΑ ΣΤΟΙΧΙΣΗΣ

- Κάθε στήλη της παραπάνω μήτρας είναι μία συντεταγμένη σε δισδιάστατο πλέγμα nxm.
- Ολόκληρη η στοίχιση είναι μία διαδρομή:

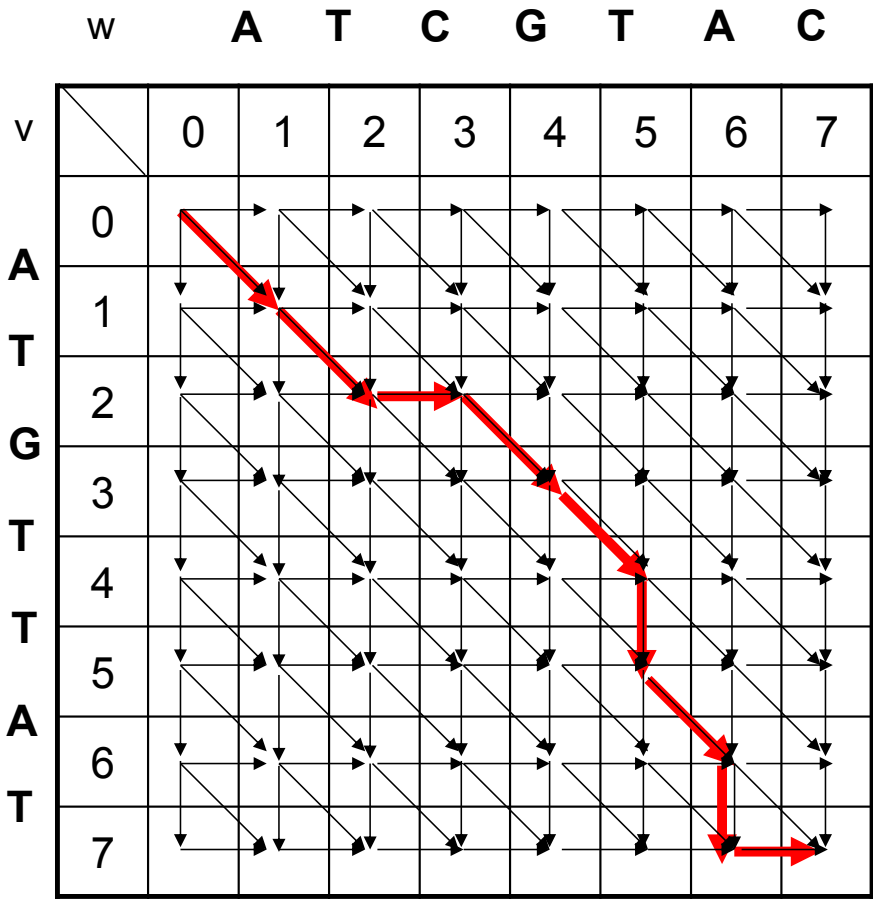
(0,0) →(1,1) →(2,2) →(2,3) →(3,4) →(4,5) →(5,5) →(6,6) →(7,6) →(7,7)

Γράφημα μετασχηματισμού (Edit graph) (1/2)

(0,0) →(1,1) →(2,2) →(2,3) →(3,4) →(4,5) →(5,5) →(6,6) →(7,6) →(7,7)

```
      0 1 2 2 3 4 5 6 7 7
v =   A T - G T T A T -
      | |   | |   |
w =   A T C G T - A - C
      0 1 2 3 4 5 5 6 6 7
```

```
↘ ↘ → ↘ ↘ ↓ ↘ ↓ →
A T - G T T A T -
| |   | |   |
A T C G T - A - C
```



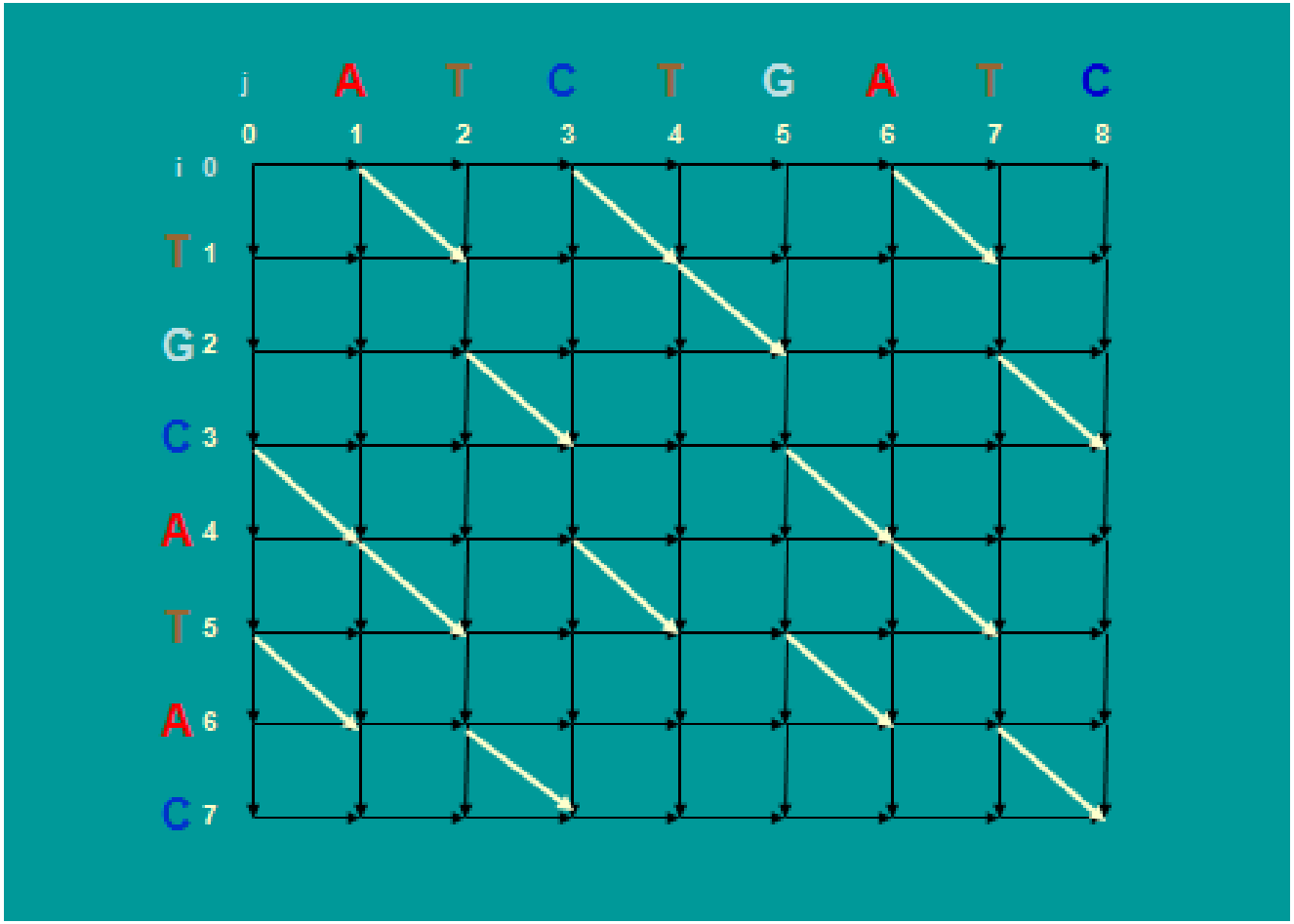
Γράφημα μετασχηματισμού (Edit graph) (2/2)

- Για δύο συμβολοσειρές υπάρχουν πολλές διαφορετικές μήτρες στοίχισης → διαφορετικές διαδρομές → διαφορετικές στοιχίσεις.
- Σύστημα βαθμονόμησης που δίνει μεγαλύτερη βαθμολογία στις στοιχίσεις με τα περισσότερα ταιριάσματα.
- Απλούστερες συναρτήσεις βαθμολόγησης:
  - Ίδια γράμματα: Θετική βαθμολογία.
  - Διαφορετικά γράμματα: Αρνητική βαθμολογία.
- Η πιο απλή συνάρτηση βαθμολόγησης:
  - Ταίριασμα: 1.
  - Διαφορετικό: 0.
- **Επόμενο πρόβλημα:** Η εύρεση της μεγαλύτερης κοινής υποαλληλουχίας.

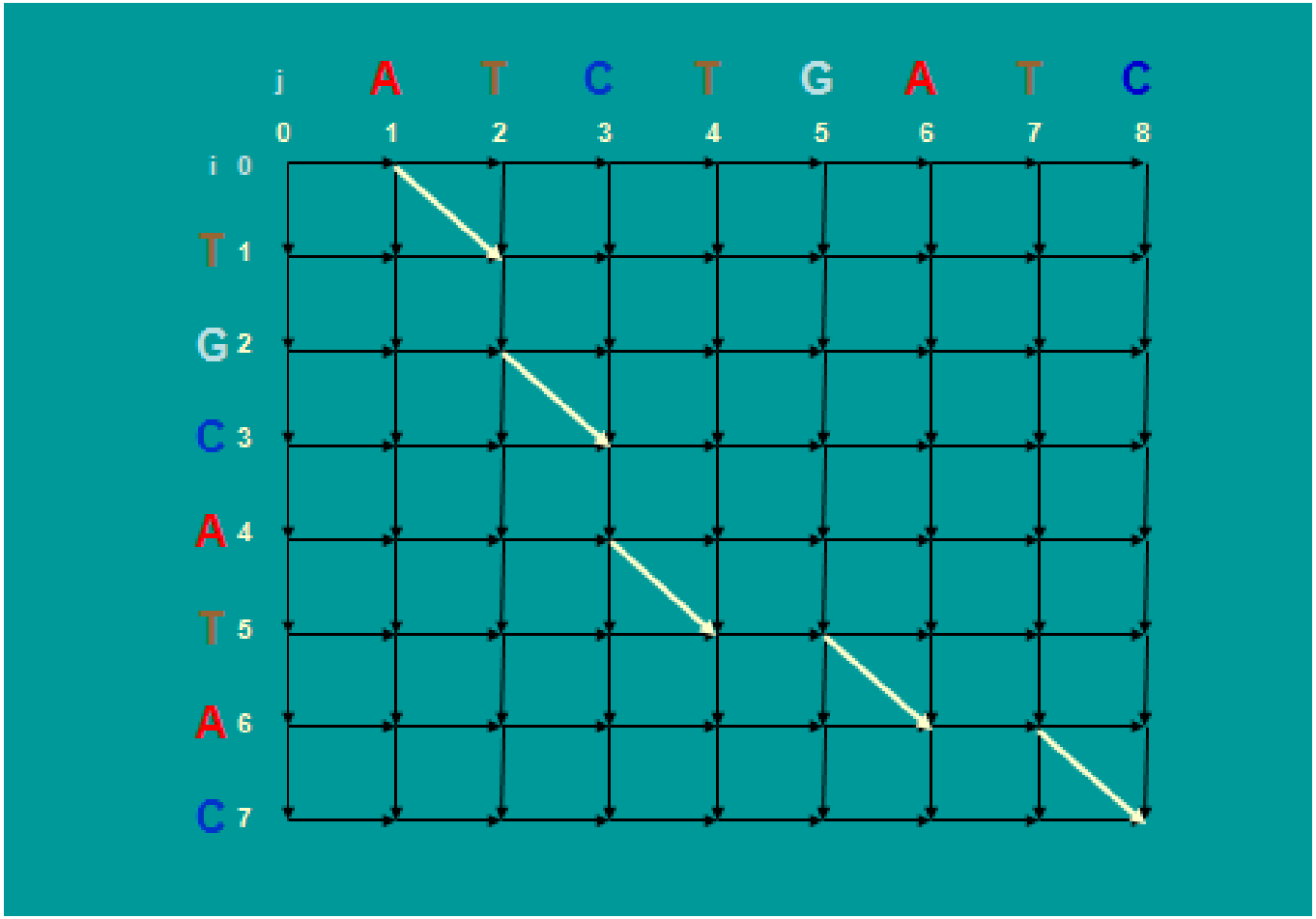
Η μεγαλύτερη κοινή υποαλληλουχία - Longest Common Subsequence (LCS)

- Η απλούστερη μορφή της ανάλυσης ομοιότητας για αλληλουχίες.
- Δεν επιτρέπεται το λάθος ταίριασμα (mismatch), μόνο οι προσθήκες και οι διαγραφές, δηλ. τα κενά και στις 2 αλληλουχίες.
- **Υποαλληλουχία συμβολοσειράς v:** Διατεταγμένη αλληλουχία χαρακτήρων (όχι απαραίτητα συνεχόμενων) από τη v.
- **Παράδειγμα:** v=ATTGCTA.
  - Υποαλληλουχίες: AGCA, ATTA.
  - Όχι υποαλληλουχίες: TGTT, TGG.
- Μία κοινή υποαλληλουχία δύο συμβολοσειρών είναι υποαλληλουχία και των δύο.
- **Παράδειγμα: v=ATCTGAT, w=TGCATA**
  - **Υποαλληλουχία: TCTA**

Edit Graph for LCS Problem (1/3)

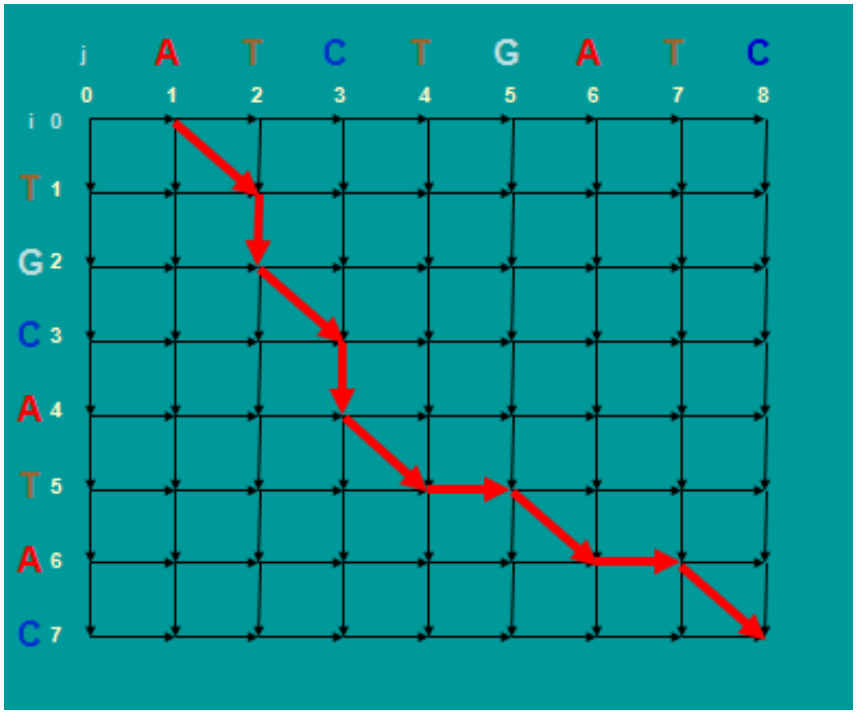


Edit Graph for LCS Problem (2/3)



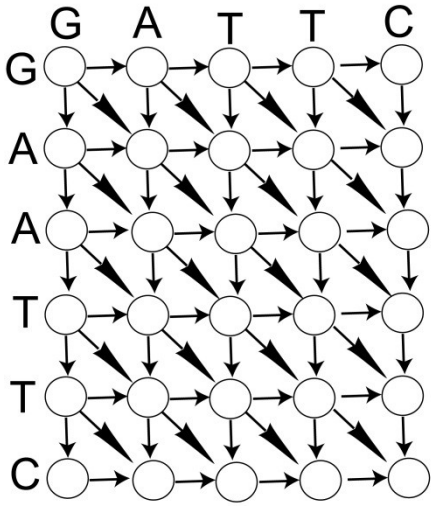
Edit Graph for LCS Problem (3/3)

- Every path is a common subsequence.
- Every diagonal edge adds an extra element to common subsequence.
- LCS Problem: Find a path with maximum number of diagonal edges.



Στάδια δυναμικού προγραμματισμού

- Initialization.
- Matrix Fill (scoring).
- Traceback (alignment).



Αλγόριθμοι δυναμικού προγραμματισμού – Παράδειγμα (1/4)

$1^{η}$ αλληλουχία: A C G T, m = 4

$2^{η}$ αλληλουχία: A G T, n = 3

Ολική στοίχιση – Αλγόριθμος Needleman-Wunsch (1/2)

- **Ολική στοίχιση** δύο αλληλουχιών. Προσπάθειες να συγκρίνουμε **όλα τα κατάλοιπα των δύο αλληλουχιών.**
- Δύο αλληλουχίες:
  - $\mathbf{x_1, x_2 \ldots x_n}$
  - $\mathbf{y_1, y_2 \ldots y_n}$
- Κατασκευάζεται ο πίνακας $\mathbf{F(i, j), 0 \leq i \leq n, 0 \leq j \leq m.}$
- Διατρέχουμε τον πίνακα από πάνω αριστερά προς κάτω δεξιά τοποθετώντας τη βαθμολογία που προκύπτει από:

Ολική στοίχιση – Αλγόριθμος Needleman-Wunsch (2/2)

1. Να στοιχηθεί το x_i με το y_j ⟶
2. Να στοιχηθεί το xi με το κενό ⟶ **MAX**
3. Να στοιχηθεί το y_j με το κενό ⟶

$$\begin{cases} F(i,j) = F(i-1,j-1) + (x_i, y_j) \\ F(i,j) = F(i-1,j) - d \\ F(i,j) = F(i,j-1) - d \end{cases}$$

όπου

- **$s(x_i,y_i)$:** η βαθμολογία για τη στοίχιση των καταλοίπων x_i με y_j
- **d:** ποινή για το κενό

Για την πρώτη γραμμή: F(i,0) = -id

Για την πρώτη στήλη: F(0,j) = -jd

Αλγόριθμοι δυναμικού προγραμματισμού – Παράδειγμα (2/4)

m+2 σειρές, n+2 στήλες

		A	C	G	T
A					
G					
T					

$$\text{MAX}\begin{cases} F(i,j) = F(i-1,j-1) + (x_i, y_j) \\ F(i,j) = F(i-1,j) + d \\ F(i,j) = F(i,j-1) + d \end{cases}$$

$s(x_i, y_j)$ = +2 αν $x_i = y_j$ (βαθμός όμοιου καταλοίπου)
$s(x_i, y_j)$ = 0 αν $x_i \neq y_j$ (βαθμός διαφορετικού καταλοίπου)
d = -1 (ποινή κενού)

Initialized Matrix – Αλγόριθμος Needleman-Wunsch (1/4)

		A	C	G	T
	GAP				
A					
G					
T					

Βαθμοί
Όμοιο: +2
Ανόμοιο: 0
Κενό: -1

Για την πρώτη γραμμή: F(i,0) = jd
Για την πρώτη στήλη: F(0,j) = id

Initialized Matrix – Αλγόριθμος Needleman-Wunsch (2/4)

		A	C	G	T
	GAP	-1	-2	-3	-4
A	-1				
G	-2				
T	-3				

Βαθμοί
Όμοιο: +2
Ανόμοιο: 0
Κενό: -1

Για την πρώτη γραμμή: F(i,0) = jd
Για την πρώτη στήλη: F(0,j) = id

Αλγόριθμοι δυναμικού προγραμματισμού – Παράδειγμα (3/4)

m+2 σειρές, n+2 στήλες

		A	C	G	T
	GAP	-1	-2	-3	-4
A	-1	2	1	0	-1
G	-2	1	2	3	2
T	-3	0	1	2	5

→ : κενό στην κάθετη

↓ : κενό στην οριζόντια

```
A   C   G   T
|       |   |
A   -   G   T
```

Επαλήθευση στοίχισης:
3·2 + 1·(-1) = 5

$s(x_i, y_j)$ = +2 (όμοιο κατάλοιπου)
$s(x_i, y_j)$ = 0 (διαφορετικό κατάλοιπο)
d = -1 (ποινή κενού)

Αλγόριθμοι δυναμικού προγραμματισμού – Παράδειγμα (4/4)

1η αλληλουχία: ASRFALFF; M = 8

2η αλληλουχία: ASIRVVFALF; N = 10

Initialized Matrix – Αλγόριθμος Needleman-Wunsch (3/4)

M+2 σειρές, N+2 στήλες

		A	S	R	F	A	L	F	F
	GAP								
A									
S									
I									
R									
V									
V									
F									
A									
L									
F									

Βαθμοί
Όμοιο: +2
Ανόμοιο: 0
Κενό: -1

Για την πρώτη γραμμή: F(i,0) = -id
Για την πρώτη στήλη: F(0,j) = -jd

Ποινή κενού (d): -1

Initialized Matrix – Αλγόριθμος Needleman-Wunsch (4/4)

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1								
S	-2								
I	-3								
R	-4								
V	-5								
V	-6								
F	-7								
A	-8								
L	-9								
F	-10								

Βαθμοί

Όμοιο: +2

Ανόμοιο: 0

Κενό: -1

Για την πρώτη γραμμή: F(i,0) = -id

Για την πρώτη στήλη: F(0,j) = -jd

Matrix fill – Αλγόριθμος Needleman-Wunsch (1/4)

$$\text{MAX}\begin{cases} F(i,j) = \ ?(i-1,j-1) + \ (x_i, y_j) \\ F(i,j) = \ ?(i-1,j) - \ ! \\ F(i,j) = \ ?(i,j-1) - \ ! \end{cases}$$

F(1,1) = MAX [$F_{0,0}$+2, $F_{1,0}$-1, $F_{0,1}$-1] = MAX[0+2, -1-1, -1-1] = MAX[2, -2, -2] = **2**

Όμοιο κατάλοιπο: +2
Ανόμοιο κατάλοιπο: 0
Κενό: -1

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2							
S	-2								
I	-3								
R	-4								
V	-5								
V	-6								
F	-7								
A	-8								
L	-9								
F	-10								

Matrix fill – Αλγόριθμος Needleman-Wunsch (2/4)

$$\text{MAX}\begin{cases} F(i,j) = F(i-1,j-1) + (x_i, y_j) \\ F(i,j) = F(i-1,j) - \\ F(i,j) = F(i,j-1) - \end{cases}$$

F(1,2) = MAX [$F_{0,1}$+2, $F_{0,2}$-1, $F_{1,1}$-1] = MAX[-1+2, -2-1, 2-1] = MAX[1, -3, -1] = **1**

Όμοιο κατάλοιπο: +2
Ανόμοιο κατάλοιπο: 0
Κενό: -1

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2	1						
S	-2								
I	-3								
R	-4								
V	-5								
V	-6								
F	-7								
A	-8								
L	-9								
F	-10								

Matrix fill – Αλγόριθμος Needleman-Wunsch (3/4)

$$\text{MAX}\begin{cases} F(i,j) = F(i-1,j-1) + (x_i, y_j) \\ F(i,j) = F(i-1,j) - \\ F(i,j) = F(i,j-1) - \end{cases}$$

F(2,1) = MAX [$F_{1,0}$+0, $F_{1,1}$-1, $F_{2,0}$-1] = MAX[-1+0, 2-1, -2-1] = MAX[-1, 1, -3] = **1**

Όμοιο κατάλοιπο: +2
Ανόμοιο κατάλοιπο: 0
Κενό: -1

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2	1						
S	-2	1							
I	-3								
R	-4								
V	-5								
V	-6								
F	-7								
A	-8								
L	-9								
F	-10								

Matrix fill – Αλγόριθμος Needleman-Wunsch (4/4)

$$\text{MAX}\begin{cases} F(i,j) = F(i-1,j-1) + (x_i, y_j) \\ F(i,j) = F(i-1,j) - \\ F(i,j) = F(i,j-1) - \end{cases}$$

F(2,2) = MAX [$F_{1,1}$+2, $F_{1,2}$-1, $F_{2,1}$-1] = MAX[+2+2, 1-1, 1-1] = MAX[+4, 0, 0] = **4**

Όμοιο κατάλοιπο: +2
Ανόμοιο κατάλοιπο: 0
Κενό: -1

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2	1						
S	-2	1	4						
I	-3								
R	-4								
V	-5								
V	-6								
F	-7								
A	-8								
L	-9								
F	-10								

Matrix traceback – Αλγόριθμος Needleman-Wunsch

$$\text{MAX}\begin{cases} F(i,j) = F(i-1,j-1) + (x_i, y_j) \\ F(i,j) = F(i-1,j) - \\ F(i,j) = F(i,j-1) - \end{cases}$$

F(2,2) = MAX [$F_{1,1}$+2, $F_{1,2}$-1, $F_{2,1}$-1] = MAX[+2+2, 1-1, 1-1] = MAX[+4, 0, 0] = **4**

Όμοιο κατάλοιπο: +2
Ανόμοιο κατάλοιπο: 0
Κενό: -1

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2	1	0	-1	-2	-3	-4	-5
S	-2	1	4	3	2	1	0	-1	-2
I	-3	0	3	4	3	2	1	0	-1
R	-4	-1	2	5	4	3	2	1	0
V	-5	-2	1	4	5	4	3	2	1
V	-6	-3	0	3	4	5	4	3	2
F	-7	-4	-1	2	5	4	5	6	5
A	-8	-5	-2	1	4	7	6	5	6
L	-9	-6	-3	0	3	6	9	8	7
F	-10	-7	-4	-1	2	5	8	11	10

Έλεγχος της βαθμολογίας

		A	S	R	F	A	L	F	F
	GAP	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2	1	0	-1	-2	-3	-4	-5
S	-2	1	4	3	2	1	0	-1	-2
I	-3	0	3	4	3	2	1	0	-1
R	-4	-1	2	5	4	3	2	1	0
V	-5	-2	1	4	5	4	3	2	1
V	-6	-3	0	3	4	5	4	3	2
F	-7	-4	-1	2	5	4	5	6	5
A	-8	-5	-2	1	4	7	6	5	6
L	-9	-6	-3	0	3	6	9	8	7
F	-10	-7	-4	-1	2	5	8	11	10

: κενό στην κάθετη

: κενό στην οριζόντια

Όμοιο κατάλοιπο: +2

Ανόμοιο κατάλοιπο: 0

Κενό: -1

A S - R - - F A L F F

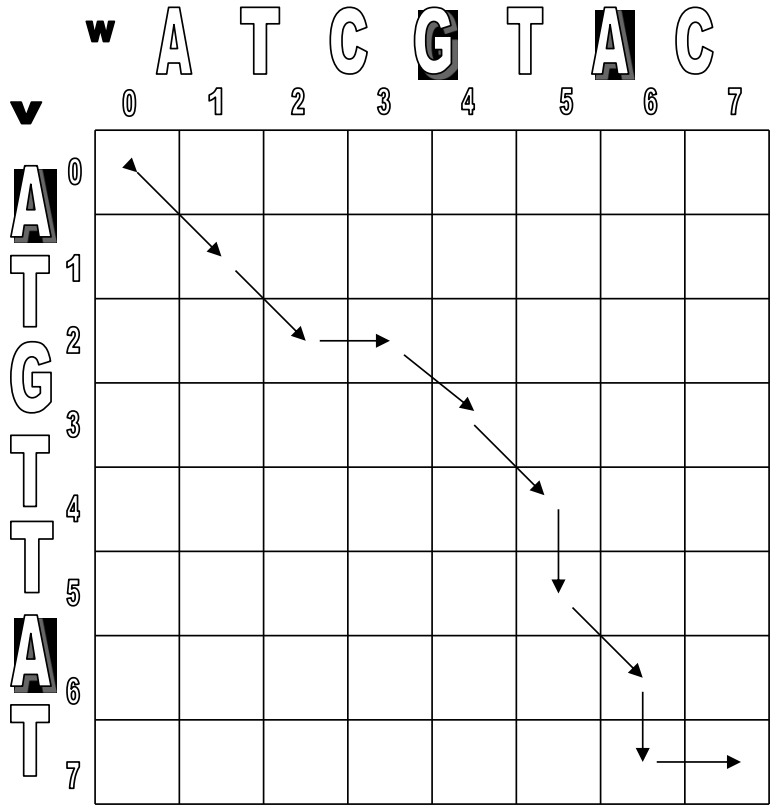
| | | | | | | |

A S I R V V F A L F -

Έλεγχος της βαθμολογίας:

(7x2) – (4x1) = 14-4 = 10

Alignment as a Path in the Edit Graph (1/3)

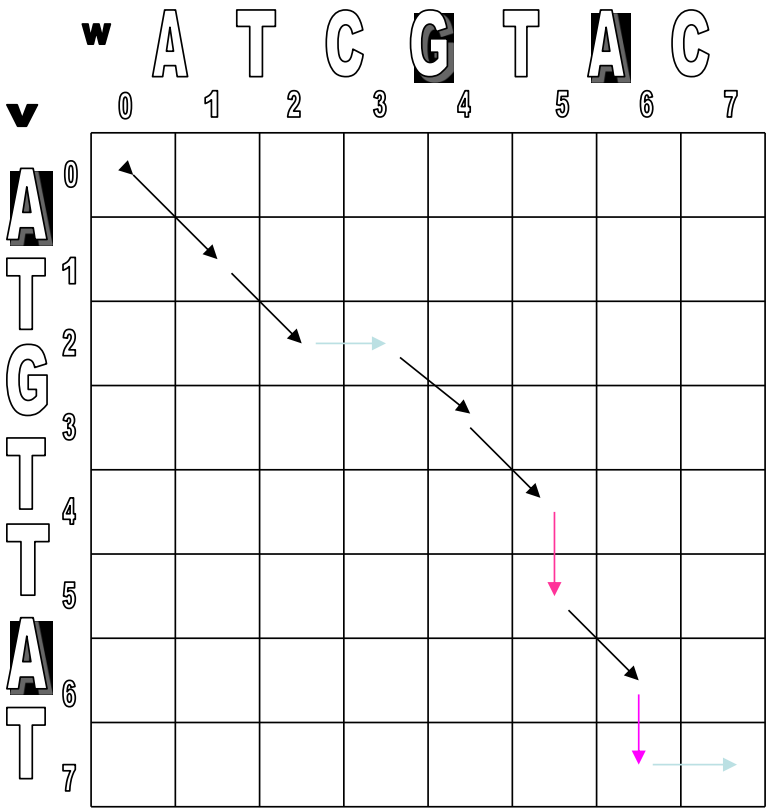


0 1 2 2 3 4 5 6 7 7
A T G T T A T
A T C G T A C
0 1 2 3 4 5 5 6 6 7

- Corresponding path -

(0,0), (1,1), (2,2), (2,3), (3,4), (4,5), (5,5), (6,6), (7,6), (7,7)

Alignment as a Path in the Edit Graph (2/3)

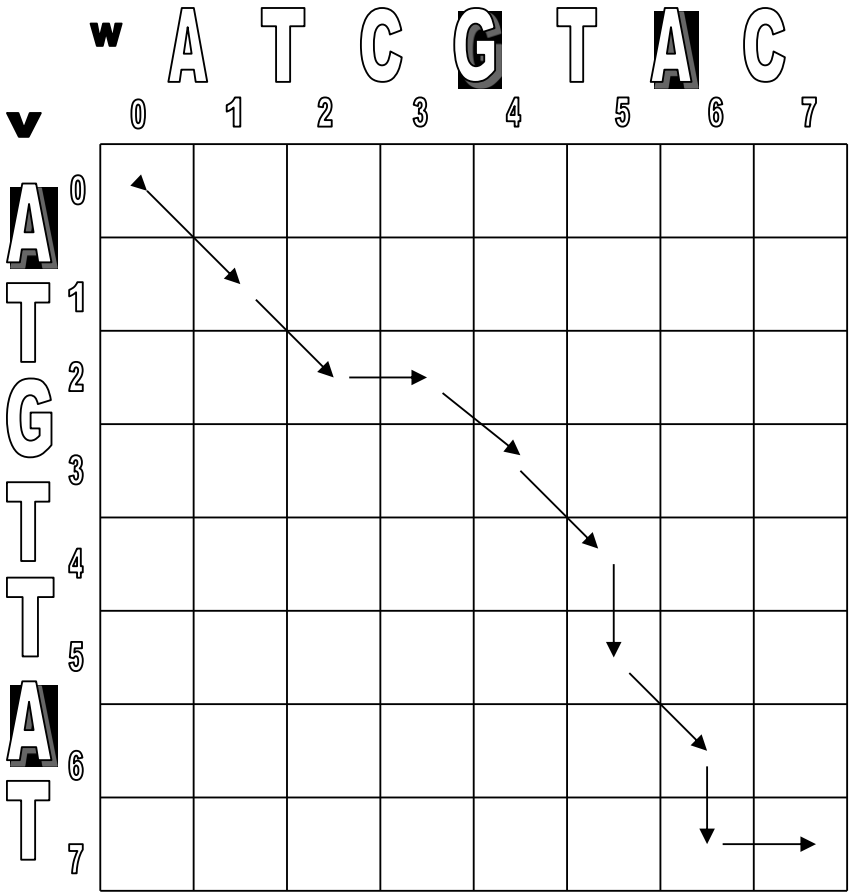


↓ and → represent indels in **v** and **w** with score 0.

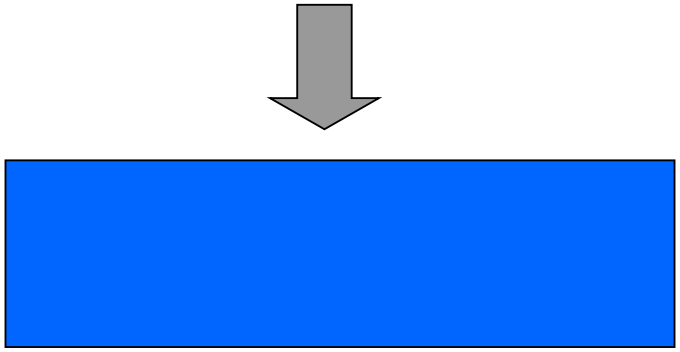
↘ represent matches with score 1.

- The score of the alignment path is 5.

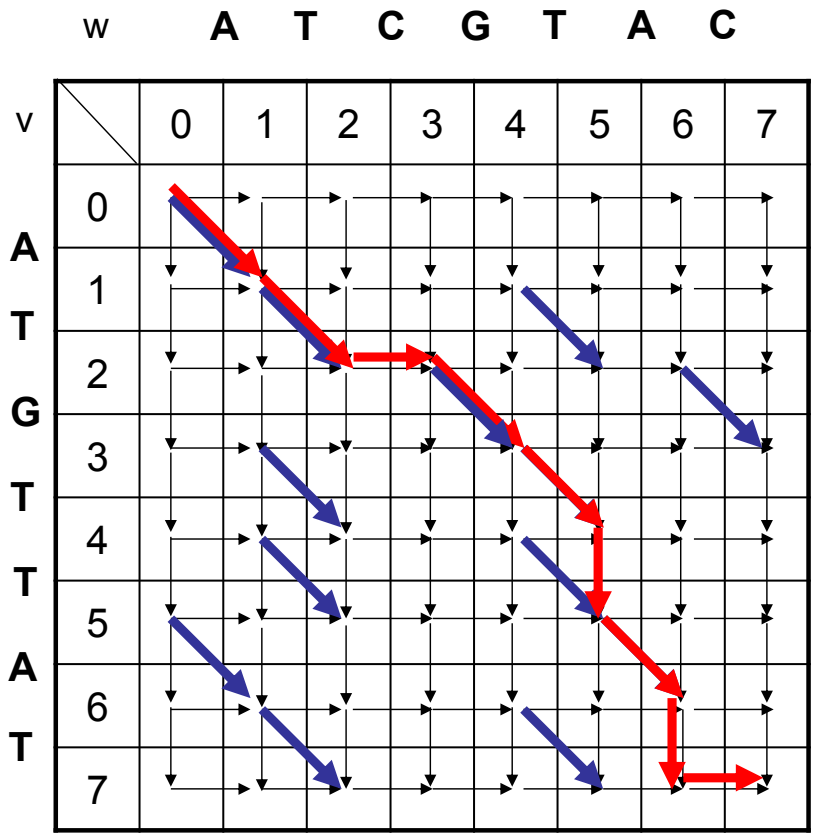
Alignment as a Path in the Edit Graph (3/3)



Every path in the edit graph corresponds to an alignment:

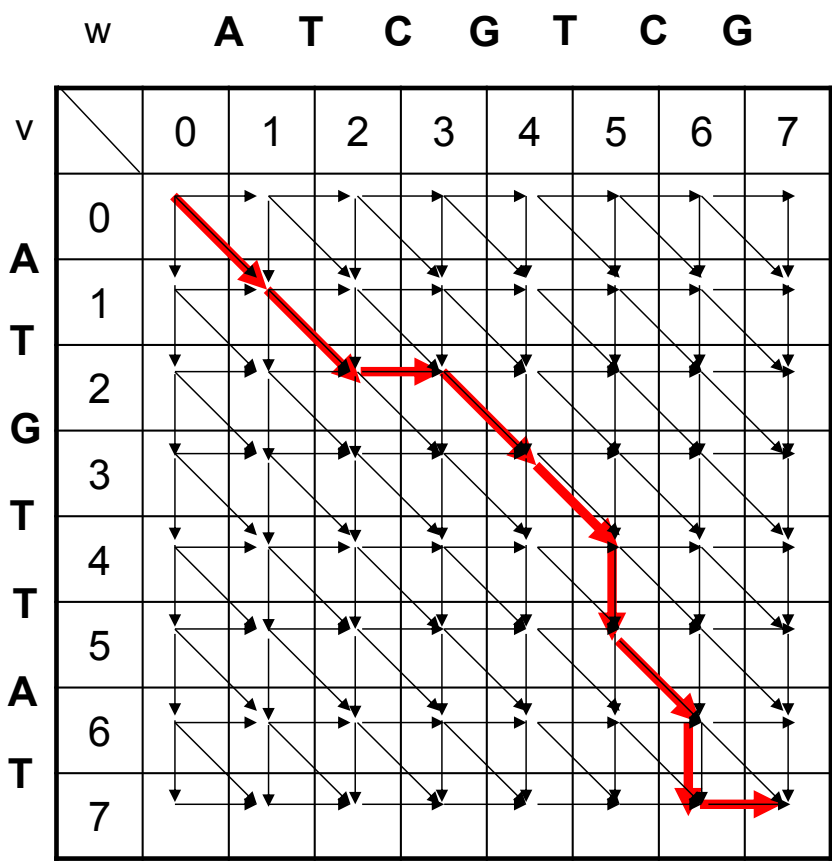


Edit Graph for LCS Problem



LCS Problem: *Find a path with maximum number of diagonal edges*

LCS Problem as Manhattan Tourist Problem



- Κάθε κοινή υποαλληλουχία αντιστοιχεί σε στοίχιση χωρίς ασυμφωνίες (διαγώνιος).

Τέλος Ενότητας

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Σημείωμα Αναφοράς

- Copyright Πανεπιστήμιο Δυτικής Μακεδονίας, Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών, Αγγελίδης Παντελής. «**Βιοπληροφορική**». Έκδοση: 1.0. Κοζάνη 2015. Διαθέσιμο από τη δικτυακή διεύθυνση:

  https: //eclass.uowm.gr/courses/ICTE102/

Σημείωμα Αδειοδότησης

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.