

# Εισαγωγικές Έννοιες

---

**Δημήτρης Φωτάκης**

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών

Εθνικό Μετσόβιο Πολυτεχνείο



# Υπολογιστικό Πρόβλημα

---

- **Επιθυμητός μετασχηματισμός** δεδομένων εισόδου σε δεδομένα εξόδου.
  - Μορφή δεδομένων εισόδου (έγκυρο στιγμιότυπο).
  - Μορφή και **ιδιότητες** δεδομένων εξόδου (απάντηση ή λύση).
  - Τυπικά: **διμελής σχέση** στις συμβ/ρές εισόδου, εξόδου.
  - Διαισθητικά: **ερώτηση** που αφορά στιγμιότυπα.
- **Στιγμιότυπο**: μαθηματικό αντικείμενο που ορίζεται από δεδομένα εισόδου.
  - Διατυπώνουμε **ερώτηση** και περιμένουμε **απάντηση**.
  - Άπειρο σύνολο στιγμιότυπων.
- Κατηγορίες Προβλημάτων:
  - **Βελτιστοποίησης**: λύση με βέλτιστη αντικειμενική τιμή.
  - **Απόφασης**: απάντηση **ΝΑΙ** ή **ΌΧΙ**.

# Προβλήματα Βελτιστοποίησης

---

- Πρόβλημα βελτιστοποίησης  $\Pi$ :
  - Σύνολο στιγμιότυπων  $\Sigma_{\Pi}$
  - Σύνολο αποδεκτών λύσεων:  $\forall \sigma \in \Sigma_{\Pi}, \Lambda_{\Pi}(\sigma)$
  - Αντικειμενική συνάρτηση:  $\forall \sigma \in \Sigma_{\Pi}, f_{\sigma} : \Lambda_{\Pi}(\sigma) \mapsto \mathbb{R}$
- Δεδομένου στιγμιότυπου  $\sigma$ , ζητείται  $\lambda_{\sigma}^* \in \Lambda_{\Pi}(\sigma)$ :
  - $\forall \lambda \in \Lambda_{\Pi}(\sigma), f_{\sigma}(\lambda_{\sigma}^*) \geq f_{\sigma}(\lambda)$  πρόβλημα **μεγιστοποίησης**
  - $\forall \lambda \in \Lambda_{\Pi}(\sigma), f_{\sigma}(\lambda_{\sigma}^*) \leq f_{\sigma}(\lambda)$  πρόβλημα **ελαχιστοποίησης**
  - $\lambda_{\sigma}^*$  βέλτιστη λύση και  $f_{\sigma}(\lambda_{\sigma}^*)$  βέλτιστη αντικειμενική τιμή**
- Συνδυαστικής βελτιστοποίησης: πεπερασμένο σύνολο αποδεκτών λύσεων που περιλαμβάνει βέλτιστη.
  - Προφανής αλγόριθμος: εξαντλητική αναζήτηση.

# Προβλήματα Απόφασης

---

- Πρόβλημα απόφασης  $\Pi$ :
  - Σύνολο στιγμιότυπων  $\Sigma_{\Pi}$
  - Σύνολο (αποδεκτών) λύσεων:  $\forall \sigma \in \Sigma_{\Pi}, \Lambda_{\Pi}(\sigma)$
  - Δεδομένου  $\sigma \in \Sigma_{\Pi}, \Lambda_{\Pi}(\sigma) \neq \emptyset$ ;
- Επιδέχεται μόνο δύο απαντήσεις: **ΝΑΙ ή ΌΧΙ.**

# Παραδείγματα Προβλημάτων

---

- **Πρόβλημα Προσπελασιμότητας:**
  - **Στιγμιότυπο:** Κατευθυνόμενο γράφημα  $G(V, E)$ , κορυφές  $s, t \in V$ .
  - **Ερώτηση:** Υπάρχει  $s - t$  μονοπάτι;
  
- **Πρόβλημα Συντομότερου Μονοπατιού:**
  - **Στιγμιότυπο:** Κατευθυνόμενο γράφημα  $G(V, E)$ , μήκη στις ακμές  $w: E \rightarrow \mathbb{R}$ , κορυφές  $s, t \in V$ .
  - **Ερώτηση:** Ποιο είναι το συντομότερο  $s - t$  μονοπάτι;

# Παραδείγματα Προβλημάτων

---

## □ Πρόβλημα κύκλου Hamilton:

- **Στιγμιότυπο:** Γράφημα  $G(V, E)$ .
- **Ερώτηση:** Υπάρχει κύκλος Hamilton στο  $G$ ;

## □ Πρόβλημα Πλανόδιου Πωλητή (TSP):

- **Στιγμιότυπο:** Σύνολο  $N = \{1, \dots, n\}$  σημείων, αποστάσεις  $d : N \times N \rightarrow \mathbb{R}_+$ .
- **Ερώτηση:** Ποια περιοδεία ελαχιστοποιεί συνολικό μήκος ή ισοδύναμα, **ποια μετάθεση  $\pi$**  του  $N$  ελαχιστοποιεί το:

$$\sum_{i=1}^{n-1} d(\pi(i), \pi(i+1)) + d(\pi(n), \pi(1))$$

# Αλγόριθμος

---

- **Σαφώς** ορισμένη διαδικασία για την **επίλυση** προβλήματος σε **πεπερασμένο** χρόνο από υπολογιστική **μηχανή**.
  - «Συνταγή» για την επίλυση υπολογιστικού προβλήματος.
  - **Σαφήνεια**: κάθε ενέργεια ορίζεται επακριβώς.
  - Είναι **μηχανιστικά αποτελεσματικός**.
  - Δέχεται ως **είσοδο στιγμιότυπο** προβλήματος και παράγει ως **έξοδο πάντα** την σωστή λύση.
  - Η λύση υπολογίζεται έπειτα από **πεπερασμένο #ενεργειών**.
- **Ορθότητα αλγόριθμου**: απαντάει πάντα σωστά.
  - **Λάθος**: αντιπαράδειγμα. **Ορθότητα**: μαθηματική απόδειξη.
- Προβλήματα λύνονται από **πολλούς σωστούς** αλγόριθμους:
  - Ποιος είναι **ο καλύτερος** (για συγκεκριμένη εφαρμογή);

# Υπολογιστική Πολυπλοκότητα

---

- Υπολογιστική πολυπλοκότητα **αλγόριθμου A**:
  - Ποσότητα υπολογιστικών πόρων που απαιτεί A ως **αύξουσα συνάρτηση μεγέθους** στιγμιότυπου εισόδου.
  - Χρόνος, μνήμη, επεξεργαστές, επικοινωνία, τυχαιότητα.
  - **Χειρότερης, μέσης, καλύτερης** περίπτωσης.
- Μέγεθος στιγμιότυπου εισόδου  **$n$** :
  - #bits για αναπαράσταση **δεδομένων εισόδου** στη μνήμη.
  - Πλήθος **βασικών συνιστωσών** που αποτελούν μέτρο μεγέθους και δυσκολίας στιγμιότυπου (π.χ. κορυφές & ακμές).
- Υπολογιστική πολυπλοκότητα **προβλήματος Π**:
  - Πολυπλοκότητα (χειρότερης περίπτωσης) **καλύτερου αλγόριθμου** που λύνει πρόβλημα Π.



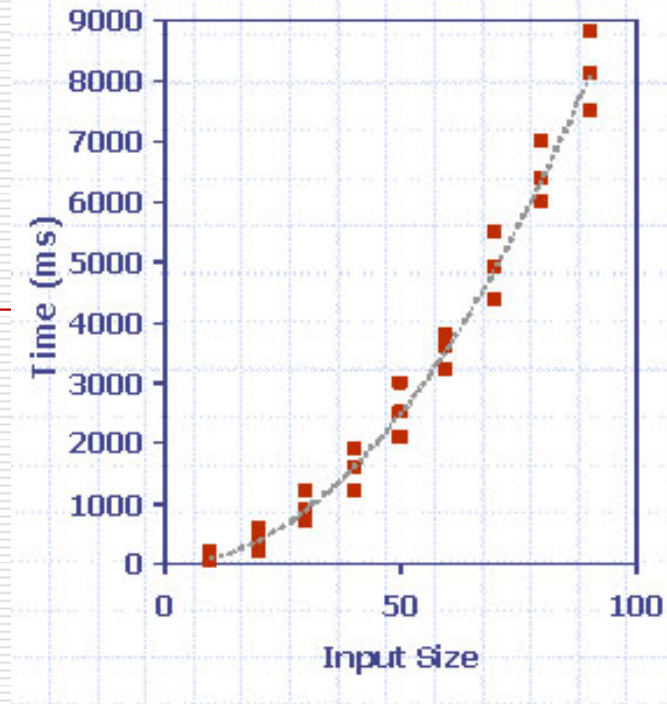
# Ανάλυση Αλγορίθμου

---

- Απόδειξη ορθότητας
  - Μερικές φορές για ένα καλώς ορισμένο υποσύνολο των στιγμιοτύπων εισόδου.
- Εκτίμηση υπολογιστικής πολυπλοκότητας.
  - **Χειρότερης, μέσης, και καλύτερης** περίπτωσης.
- Καταλληλότερη λύση ανάλογα με **απαιτήσεις εφαρμογής.**

# Πειραματική Μελέτη

- Υλοποίηση αλγόριθμου σε πρόγραμμα.
  - Δημιουργία στιγμιotypών διαφορετικού μεγέθους και σύνθεσης.
  - Επιβεβαίωση ορθότητας και καταγραφή πόρων για κάθε εκτέλεση.
  - Απεικόνιση αποτελεσμάτων σε γραφική παράσταση και εξαγωγή συμπερασμάτων.
- Περιορισμοί – Δυσκολίες:
  - Υλοποίηση χρονοβόρα και ενδεχομένως δύσκολη.
  - Αποτελέσματα όχι κατ' ανάγκη αντιπροσωπευτικά.
  - Συμπεράσματα δεν γενικεύονται κατ' ανάγκη.



# Θεωρητική Ανάλυση

---

- Δεν απαιτεί υλοποίηση αλλά σαφή **περιγραφή** αλγόριθμου.
- Καταλήγει σε **γενικά συμπεράσματα**:
  - Λαμβάνει υπόψη **όλα** τα στιγμιότυπα.
  - Προσδιορίζει υπολογιστική πολυπλοκότητα ως συνάρτηση μεγέθους εισόδου, ...
  - ... αλλά **ανεξάρτητα** από υπολογιστικό **περιβάλλον**.
  - Εστιάζει στις **εγγενείς ιδιότητες** του αλγόριθμου.
- Συμπεράσματα **επιβεβαιώνονται** εύκολα.
- Μαθηματικό υπόβαθρο: Διακριτά Μαθηματικά.
  - Γραφήματα, μαθηματική λογική, επαγωγή, αναδρομικές σχέσεις, συνδυαστική, πιθανότητες, ...

# Υπολογιστικό Μοντέλο

---

- Μηχανή Άμεσης Προσπέλασης Μνήμης (Random Access Machine, RAM).
  - **Ιδεατό μονο-επεξεργαστικό** σύστημα που ακολουθεί **αρχιτεκτονική von Neumann**.
  - Ένας επεξεργαστής, ακολουθιακή εκτέλεση εντολών
  - Απεριόριστες θέσεις μνήμης προσπελάσιμες με **διεύθυνση**.
  - Στοιχειώδη υπολογιστικά **βήματα** εκτελούνται σε **μοναδιαίο χρόνο**:
    - Ανάγνωση από / εγγραφή σε θέση μνήμης, αριθμητικές και λογικές πράξεις, συγκρίσεις, εντολές ελέγχου ροής, ...

# Ασυμπτωτική Εκτίμηση

---

- Χρόνος εκτέλεσης αλγόριθμου  $A$ :
  - Αύξουσα συνάρτηση του  $T(n)$  που εκφράζει σε πόσο χρόνο ολοκληρώνεται ο  $A$  όταν εφαρμόζεται σε στιγμ. μεγέθους  $n$ .
- Ενδιαφέρει η τάξη μεγέθους  $T(n)$  και όχι ακριβής εκτίμηση  $T(n)$ .
  - Ακριβής εκτίμηση είναι συχνά δύσκολη και εξαρτάται από υπολογιστικό περιβάλλον, υλοποίηση, ...
  - Τάξη μεγέθους είναι **εγγενής ιδιότητα** του αλγόριθμου.
    - Δυαδική αναζήτηση έχει λογαριθμικό χρόνο.
    - Γραμμική αναζήτηση έχει γραμμικό χρόνο.
- **Ασυμπτωτική εκτίμηση** αγνοεί σταθερές και εστιάζει σε τάξη μεγέθους χρόνου εκτέλεσης.