Εισαγωγή στην Υπολογιστική Πολυπλοκότητα

Διδάσκοντες: **Ε. Ζάχος, Δ. Φωτάκης**
Επιμέλεια διαφανειών: **Δ. Φωτάκης**

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Εθνικό Μετσόβιο Πολυτεχνείο

Θεωρία Υπολογισμού

- Γιατί κάποια προβλήματα είναι **αδύνατο** να λυθούν από υπολογιστές;
- **Hilbert** (1900): **πληρότητα** και **αυτοματοποίηση** των μαθηματικών.
- 10ο πρόβλημα : **Αλγόριθμος** για λύση Διοφαντικών εξισώσεων:
  Έχει $x^2 - 2y^2 + 3 = 0$ **ακέραιες** ρίζες;
- **Αλγόριθμος**: Διατύπωση και απόδειξη.
- **Όχι αλγόριθμος**: Ορισμός "αλγόριθμου" μέσω **υπολογιστικού μοντέλου**.
  Απόδειξη ότι "αλγόριθμος $\Rightarrow$ αντίφαση στο μοντέλο".
- **Gödel**: Μαθηματικά **δεν είναι πλήρη**!
  Υπάρχουν "αλήθειες" που δεν αποδεικνύονται.
- **Turing**: Μαθηματικά **δεν αυτοματοποιούνται**!
  Μη-επιλύσιμα προβλήματα: επίλυσή τους δεν αυτοματοποιείται (γενική περίπτωση).
- **Matijasevic** (1970): Δεν υπάρχει αλγόριθμος για Διοφαντικές εξισώσεις!
  Για κάθε αλγόριθμο $\mathcal{A}$, υπάρχει εξίσωση που ο $\mathcal{A}$ δίνει **λάθος απάντηση**!

Υπολογιστική Πολυπλοκότητα

- Γιατί κάποια προβλήματα είναι **δύσκολο** να λυθούν από υπολογιστές;
  Ανάπτυξη τελευταία 30 χρόνια (Papadimitriou, Computational Complexity, 1994).
- Ποια **επιλύσιμα** προβλήματα είναι **εύκολα** και ποια **δύσκολα**.
- **Επιλύσιμα** προβλήματα: Υπολογιστικοί πόροι;
  - Εύλογοι υπολογιστικοί πόροι $\Rightarrow$ **ευεπίλυτα** (tractable) προβλήματα.
    Πολλαπλασιασμός Πινάκων, Κλασματικό Σακίδιο, Ελάχιστο Επικαλύπτον Δέντρο, Συντομότερα Μονοπάτια.
  - Διαφορετικά, **δισεπίλυτα** (intractable).
    Ακέραιο Σακίδιο, Περιοδεύων Πωλητής, Κάλυψη Συνόλων, Δρομολόγηση, Χρωματισμός, Συντομότερα Μονοπάτια με Περιορισμούς, κλπ.
  - Επίδραση **υπολογιστικού μοντέλου** στους υπολογιστικούς πόρους.

Προβλήματα και Αλγόριθμοι

- **Αλγόριθμος** είναι λεπτομερής περιγραφή μεθόδου επίλυσης προβλήματος.
  υπολογιστική μηχανή (Turing) που τερματίζει.
- **Υπολογιστικό πρόβλημα** αποτελείται από άπειρο σύνολο στιγμιοτύπων.
  αποτελεί αντικείμενο μελέτης.
- **Στιγμιότυπο** είναι μαθηματικό αντικείμενο για το οποίο ρωτάμε **ερώτηση** και περιμένουμε **απάντηση**.
- Δύο είδη προβλημάτων:
  - **Απόφασης**: απαντήσεις **ΝΑΙ** ή **ΟΧΙ**.
  - **Βελτιστοποίησης**: καλύτερη εφικτή **λύση**.

Παραδείγματα Προβλημάτων

- **Πρόβλημα Προσπελασιμότητας**:
  - **Στιγμιότυπο**: Κατευθυνόμενο γράφημα $G(V, E)$ και διακεκριμένες κορυφές s και t.
  - **Ερώτηση**: Υπάρχει μονοπάτι από s στο t;
- **Πρόβλημα Συντομότερου Μονοπατιού**:
  - **Στιγμιότυπο**: Κατευθυνόμενο γράφημα με μήκη στις ακμές $G(V, E, w)$ και διακεκριμένες κορυφές s και t.
  - **Ερώτηση**: Ποιο είναι το συντομότερο $s - t$ μονοπάτι;

Παραδείγματα Προβλημάτων

- **Πρόβλημα κύκλου Hamilton**:
  - **Στιγμιότυπο**: Γράφημα $G(V, E)$.
  - **Ερώτηση**: Υπάρχει κύκλος Hamilton στο G (κύκλος που διέρχεται από κάθε κορυφή ακριβώς μία φορά);
- **Πρόβλημα Περιοδεύοντος Πωλητή**:
  - **Στιγμιότυπο**: Σύνολο $= \{1, \ldots, n\}$ σημείων και αποστάσεις $d(i, j)$ μεταξύ κάθε ζεύγους διαφορετικών σημείων.
  - **Ερώτηση**: Ποια μετάθεση π του N ελαχιστοποιεί

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

Προσέγγιση

- **Κλάσεις προβλημάτων** (complexity classes) με παρόμοια "δυσκολία" (υπολογιστική πολυπλοκότητα).
- **Αναγωγή σε πλήρη** (complete) προβλήματα κάθε κλάσης: Συνοψίζουν **δυσκολία** της κλάσης.
- **Πλήρες** πρόβλημα "εύκολο" $\Rightarrow$ **Όλη** η κλάση "εύκολη".
- **Αρνητικά** αποτελέσματα $\Rightarrow$ **Όλη** η κλάση "δύσκολη".
- Προσδιορισμός **υπολογιστικού έργου** για λύση προβλημάτων στην κλάση (με παραδειγματικά υπολογιστικά προβλήματα)!
- Διαλεκτική σχέση **αλγόριθμων** και **πολυπλοκότητας**.

Κωδικοποίηση Προβλημάτων σε Τυπικές Γλώσσες

- Πρόβλημα βελτιστοποίησης $\rightarrow$ πρόβλημα απόφασης με φράγμα B.
  - Ελαχιστοποίηση: Υπάρχει εφικτή λύση με κόστος $\leq B$;
  - Μεγιστοποίηση: Υπάρχει εφικτή λύση με κέρδος $\geq B$;
- Πρόβλημα απόφασης $\rightarrow$ τυπική γλώσσα με κωδικοποίηση.
  - Στιγμιότυπο $\rightarrow$ συμβολοσειρά σε αλφάβητο Σ.
  - Πρόβλημα $\rightarrow$ γλώσσα, υποσύνολο του Σ^*.
- Πρόβλημα Π και κωδικοποίηση e: Γλώσσα $\mathcal{L}(\Pi, e)$ αποτελείται από $x \in \Sigma^*$ που προκύπτουν από την e-κωδικοποίηση των ΝΑΙ-στιγμιοτύπων του Π.

$$\mathcal{L}(\Pi, e) = \{e(x) \in \Sigma^* : x \in \Pi\}$$

- Π.χ. προβλήματα προσπελασιμότητας και κύκλου Hamilton.

Ντετερμινιστικές Μηχανές Turing

- Ντερμινιστική Μηχανή Turing (DTM) $M = (Q, \Sigma, \delta, q_0, F)$:
  - Q **σύνολο καταστάσεων**.
  - Σ **αλφάβητο εισόδου**. $\Gamma = \Sigma \cup \{\sqcup\}$ **αλφάβητο ταινίας**.
  - $q_0 \in Q$ **αρχική κατάσταση**.
  - $F \subseteq Q$ **τελικές καταστάσεις**.
  - $\delta : (Q \setminus F) \times \Gamma \mapsto Q \times \Gamma \times \{L, R, S\}$ **συνάρτηση μετάβασης**.
    (κατάσταση q, κεφαλή διαβάζει α) $\rightarrow$
    (q', κεφαλή γράφει α', κεφαλή μετακινείται L, R, ή S)

Ντετερμινιστικές Μηχανές Turing

- **Ντετερμινισμός**: $M(x)$ εξελίσσεται με **προδιαγεγραμμένο** τρόπο.
- M τερματίζει σε τελική κατάσταση: ΝΑΙ, ΟΧΙ, ΤΕΛΟΣ ή δεν τερματίζει.
  - $M(x) =$ ΝΑΙ, M **αποδέχεται** x.
  - $M(x) =$ ΟΧΙ, M **απορρίπτει** x.
  - $\mathcal{L}(M) = \{x \in \Sigma^* : M(x) = \text{ΝΑΙ}\}$.
  - $M(x) =$ ΤΕΛΟΣ και έξοδος y, M **υπολογίζει** $y = f(x)$.
- **Καθολική Μηχανή Turing**: $U(M; x) = M(x)$.
  Προσομοιώνει τη λειτουργία της μηχανής M με είσοδο x.

Υπολογισιμότητα

- **Ημιαποφασίσιμη** $\mathcal{L}$: $\forall x \in \mathcal{L}$, $M(x) =$ ΝΑΙ.
  $\forall x \notin \mathcal{L}$, $M(x) \neq$ ΝΑΙ (μπορεί να μην τερματίζει).
- **Αποφασίσιμη** $\mathcal{L}$: $\forall x \in \mathcal{L}$, $M(x) =$ ΝΑΙ.
  $\forall x \notin \mathcal{L}$, $M(x) =$ ΟΧΙ.
- **Υπολογίσιμη** f: $\forall x \in \Sigma^*$, $f(x) = y \Rightarrow M(x) = y$.
  $f(x)$ δεν ορίζεται $\Rightarrow M(x)$ δεν τερματίζει.
- **Αξίωμα Church - Turing**: Υπολογίσιμο $\Leftrightarrow$ Turing αποφασίσιμο / υπολογίσιμο!
- Να αποδείξετε τις ακόλουθες προτάσεις:
  1. Κάθε αποφασίσιμη γλώσσα είναι και ημιαποφασίσιμη.
  2. $\mathcal{L}$ αποφασίσιμη $\Rightarrow$ συμπλήρωμα $\overline{\mathcal{L}}$ αποφασίσιμο.
  3. $\mathcal{L}$ αποφασίσιμη $\Leftrightarrow \mathcal{L}$ και $\overline{\mathcal{L}}$ ημιαποφασίσιμες.

Μη-Υπολογισιμότητα

- Μη-αποφασίσιμες γλώσσες (προβλήματα που **δεν λύνονται**)
  γιατί γλώσσες **πάρα πολλές** και μηχανές Turing **πολλές**.
- **Πρόβλημα Τερματισμού**: $M(x)$ τερματίζει;
- Το πρόβλημα Τερματισμού είναι **μη-αποφασίσιμο**!
- **Απόδειξη**: DTM H τ.ω. $H(M; x)$ **αποφασίζει** αν $M(x)$ τερματίζει.
  $H(M; x) =$ ΝΑΙ αν $M(x)$ τερματίζει.
  $H(M; x) =$ ΟΧΙ αν $M(x)$ δεν τερματίζει.
- Θεωρούμε DTM $D(M)$:
  if $H(M; M) =$ ΝΑΙ then **run forever** else **halt**
- $D(M)$ **τερματίζει** ανν $M(M)$ **δεν τερματίζει**!
- $D(D)$ τερματίζει **ανν** $D(D)$ δεν τερματίζει! **Άτοπο!**
- Πολλά άλλα προβλήματα δεν λύνονται!!!

Χρονική Πολυπλοκότητα

- **Χρονική Πολυπλοκότητα** $M \equiv$ αύξουσα $t : \mathbb{N} \mapsto \mathbb{N}$:
  $\forall x,\ |x| = n,\ M(x)$ τερματίζει $\leq t(n)$ βήματα.
- **Χρονική Πολυπλοκότητα** $\Pi \equiv$ Πολυπλοκότητα γρηγορότερης M που λύνει Π.
- $\mathbf{DTIME}[t(n)] \equiv \{\Pi : \Pi$ λύνεται σε χρόνο $\mathrm{O}(t(n))\}$
- **Ιεραρχία Κλάσεων** Χρονικής Πολυπλοκότητας:
  - $\mathbf{DTIME}[t(n)] \subset \mathbf{DTIME}[\omega(t(n)\log t(n))]$
  - $\mathbf{DTIME}[n] \subset \mathbf{DTIME}[n^2] \subset \mathbf{DTIME}[n^3] \subset \cdots$
- **Πολυωνυμικός Χρόνος**: $\mathbf{P} \equiv \cup_{k\geq 0}\mathbf{DTIME}[n^k]$.
- **Εκθετικός Χρόνος**: $\mathbf{EXP} \equiv \cup_{k\geq 0}\mathbf{DTIME}[2^{n^k}]$.

$$\mathbf{P} \subset \mathbf{EXP}$$

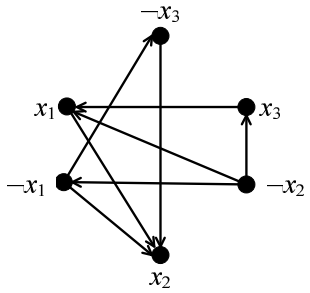
Ευεπίλυτα και Δυσεπίλυτα Προβλήματα

- **Κλάση P**: προβλήματα που λύνονται σε πολυωνυμικό χρόνο.
- **Αξίωμα Cook - Karp**: P ταυτίζεται με ευεπίλυτα προβλήματα.
- **Υπέρ**:
  - Δεν εξαρτάται από υπολογιστικό μοντέλο!
  - Συνήθως **μικρά** πολυώνυμα (π.χ. $n, n^2, n^3, \ldots$).
  - Διπλασιασμός υπολογιστικής ισχύος $\Rightarrow$ **σημαντική αύξηση** (π.χ. $2, \sqrt{2}, 2^{1/3}, \ldots$) μεγέθους στιγμιοτύπων.
- **Κατά**:
  - **Ακραίες περιπτώσεις**: Αλγόριθμος με χρόνο n^{100} δεν είναι πρακτικός ενώ αλγόριθμος με χρόνο $2^{n/100}$ είναι!
  - **Γραμμικός Προγραμματισμός**: Simplex εκθετικός στη θεωρία αλλά ταχύτερος στην πράξη!

2-Ικανοποιησιμότητα $\in$ P

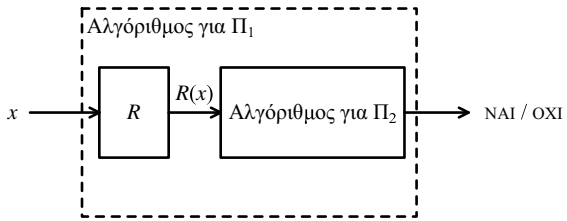
- Λογική πρόταση ϕ σε k-**Συζευκτική Κανονική Μορφή** (k-ΣΚΜ):
  $\phi = c_1 \wedge \ldots \wedge c_m,\ c_i = \ell_{i_1} \vee \ldots \vee \ell_{i_k},\ \ell \in \{x_i, \neg x_i\}$.

  Π.χ. $k = 2$: $\quad (x_1 \vee x_2) \wedge (x_1 \vee \neg x_3) \wedge (\neg x_1 \vee x_2) \wedge (x_2 \vee x_3)$
- k-**Ικανοποιησιμότητα**: ϕ σε k-ΣΚΜ ικανοποιήσιμη;
- $\ell_i \vee \ell_j \equiv (\neg\ell_i \to \ell_j) \wedge (\neg\ell_j \to \ell_i)$.
- Γράφημα G_ϕ με κορυφές $\{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, \neg x_n\}$.
  Όρος $\ell_i \vee \ell_j$: ακμές $(\neg\ell_i, \ell_j)$ και $(\neg\ell_j, \ell_i)$.
- G_ϕ συμμετρικό: $(x, x') \Leftrightarrow (\neg x', \neg x)$.
- $\ell_i \to \ell_j$ ψευδής $\Leftrightarrow \ell_i = 1$ και $\ell_j = 0$.
- ϕ μη ικανοποιήσιμη ανν μονοπάτι $x - \neg x$ και μονοπάτι $\neg x - x$.



Αναγωγή και Πληρότητα

- Π_1 **ανάγεται πολυωνυμικά** σε Π_2: $\exists$**πολυωνυμικά υπολογίσιμη** συνάρτηση $R : \Sigma^* \mapsto \Sigma^*$, ώστε $\forall x \in \Sigma^*,\ x \in \Pi_1 \Leftrightarrow R(x) \in \Pi_2$ ($\Pi_1 \leq_P \Pi_2$).
  R ονομάζεται **πολυωνυμική αναγωγή**.
- $\Pi_1 \leq_P \Pi_2$ "δηλώνει" ότι το Π_2 είναι τουλ. **τόσο δύσκολο** όσο το Π_1.
- $\mathbf{C}$ κλάση προβλημάτων. Π είναι $\mathbf{C}$-**δύσκολο** αν $\forall \Pi' \in \mathbf{C}$ ανάγεται στο Π.
  Αν Π είναι $\mathbf{C}$-δύσκολο και $\Pi \in \mathbf{C} \Rightarrow \Pi$ είναι $\mathbf{C}$-**πλήρες**.
- Τα $\mathbf{C}$-**πλήρη** προβλήματα **"συνοψίζουν"** τη δυσκολία της κλάσης $\mathbf{C}$.
- Κλάση $\mathbf{C}$ είναι **κλειστή** ως προς (πολυωνυμική) αναγωγή αν $\forall \Pi_1, \Pi_2,\ \Pi_1 \leq_P \Pi_2$ και $\Pi_2 \in \mathbf{C} \Rightarrow \Pi_1 \in \mathbf{C}$.



Κάποιες Παρατηρήσεις

- Πολυωνυμική αναγωγή είναι μεταβατική (σύνθεση αναγωγών).
- $\mathbf{P}$ κλειστό ως προς πολυωνυμική αναγωγή.
- $\Pi_1 \leq_P \Pi_2$ και $\Pi_2 \in \mathbf{P}$ ($\mathbf{NP}$) $\Rightarrow \Pi_1 \in \mathbf{P}$ ($\mathbf{NP}$).
- (Κλειστές) κλάσεις με **κοινό πλήρες πρόβλημα ταυτίζονται**!
- Κλάσεις $\mathbf{C}, \mathbf{C}'$ κλειστές ως προς αναγωγή.
  Αν έχουν κοινό πλήρες πρόβλημα, $\mathbf{C} = \mathbf{C}'$.