ΕΠΛ 211:
Θεωρία Υπολογισμού και Πολυπλοκότητας

Επανάληψη Μαθήματος

Το Μάθημα σε μια Διαφάνεια

- Υπολογιστικά μοντέλα
  - Κανονικές Γλώσσες
    - Ντετερμινιστικά Αυτόματα
    - Μη Ντετερμινιστικά Αυτόματα
    - Κανονικές Εκφράσεις
    - Μη κανονικές γλώσσες (Λήμμα Άντλησης)
  - Ασυμφραστικές Γλώσσες
    - Ασυμφραστικές Γραμματικές
    - Αυτόματα με Στοίβα
    - Μη ασυμφραστικές γλώσσες (Λήμμα Άντλησης)
  - Μηχανές Turing
- Υπολογισιμότητα
  - Τι είναι αλγόριθμος?
  - Επιλύσιμα και Μη Επιλύσιμα προβλήματα
    - Περιγραφή αλγορίθμου
    - Απεικονιστικές Αναγωγές
- Πολυπλοκότητα Επιλύσιμων Προβλημάτων
  - Χρονική και Χωρική

Εισαγωγικά

- Μαθηματικό Υπόβαθρο
  - Σύνολα και Πράξεις Συνόλων
  - Είδη Αποδείξεων

- Γραφήματα

- Γλώσσες
  - Τι είναι το Αλφάβητο?
    - Σύνολο από σύμβολα
  - Τι είναι η λέξη?
    - Ακολουθία συμβόλων
  - Τι είναι η γλώσσα?
    - Σύνολο από λέξεις

Κανονικές Γλώσσες

- Ντετερμινιστικά Αυτόματα
- Μη Ντετερμινιστικά Αυτόματα
- Κανονικές Εκφράσεις
- Μη κανονικές γλώσσες (Λήμμα Άντλησης)

Ντετερμινιστικά Αυτόματα

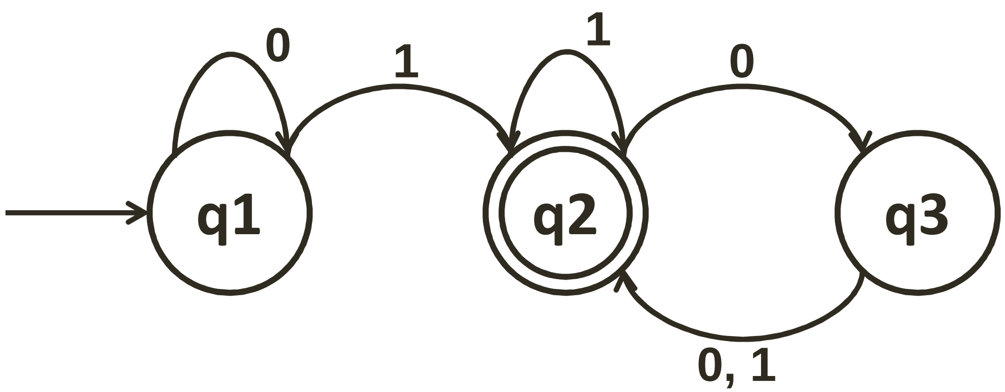
Ορισμός

***Ντετερμινιστικό Πεπερασμένο Αυτόματο* είναι μια πεντάδα** $(Q, \Sigma, \delta, q_0, F)$

- ***Q* = *πεπερασμένο* σύνολο καταστάσεων**
- Σ **= αλφάβητο**
- ***Συνάρτηση μετάβασης*** $\delta: Q \times \Sigma_e \rightarrow Q$
  Όταν το αυτόματο είναι στην κατάσταση $q \in Q$ ***και διαβάσει το σύμβολο*** $a \in \Sigma$ ***μεταβιβάζεται σε μια κατάσταση q' π.χ.* δ (q,α)=q'**
- $q_o \in Q$ **= *αρχική κατάσταση***
- $F \subseteq Q$ **= *τελικές καταστάσεις***

Από κάθε κατάσταση q προσδιορίζουμε μια μετάβαση για κάθε σύμβολο του αλφαβήτου.

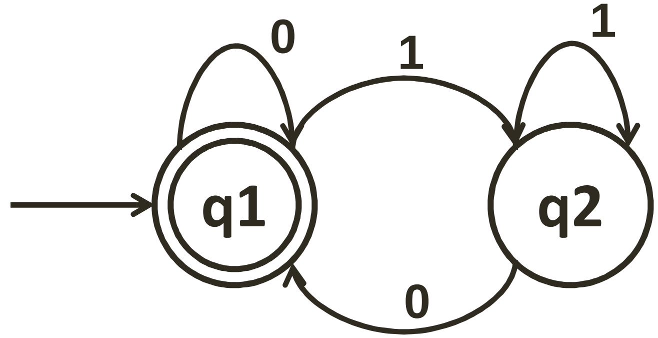
Τυπική Περιγραφή Αυτομάτου



- $M_1 = (Q, \Sigma, \delta, q_0, F)$
  - Q = {q1, q2, q3}
  - Σ = {0, 1}
  - Η συνάρτηση μεταβάσεων δ *περιγράφεται από τον πίνακα*
  - q1 η εναρκτήρια κατάσταση
  - F = {q2}

	0	1
q1	q2	q2
q2	q3	q2
q3	q2	q2

Παράδειγμα



- Τι γλώσσα αναγνωρίζει το αυτόματο M3=({q1, q2}, {0,1}, δ, q1, {q1})
  - L(M3)={w | w η κενή λέξη ή τελειώνει σε 0}

Παράδειγμα 2

- L={w | η w περιέχει το πολύ δύο 0}

Τυπικός Ορισμός του NFA

Ορισμός

Μη Ντετερμινιστικό Πεπερασμένο Αυτόματο είναι μια πεντάδα $(Q, \Sigma, \delta, q_0, F)$

Q = πεπερασμένο σύνολο καταστάσεων

Σ *= αλφάβητο*

Συνάρτηση μετάβασης $\delta : Q \times \Sigma_\varepsilon \rightarrow P(Q)$

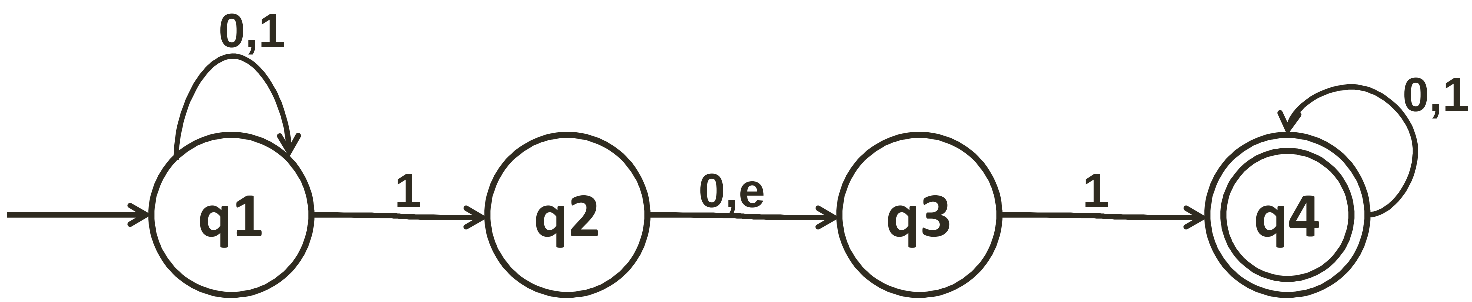
Όταν το αυτόματο είναι στην κατάσταση $q \in Q$ *και διαβάσει το σύμβολο* $a \in \Sigma$ *μεταβιβάζεται σε ένα υποσύνολο καταστάσεων π.χ.* δ (q,α)={q’, q’’}

$q_o \in Q$ *= αρχική κατάσταση*

$F \subseteq Q$ *= τελικές καταστάσεις*

Από κάθε κατάσταση q προσδιορίζουμε καμία ή περισσότερες μεταβάσεις για κάθε σύμβολο του αλφαβήτου.

DFA vs NFA



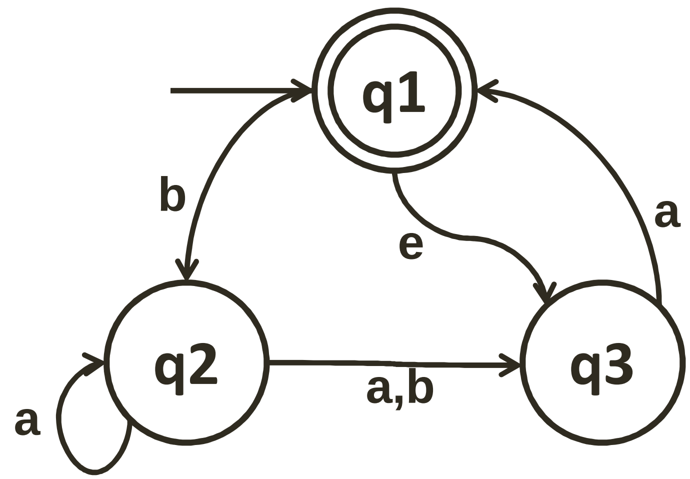
- Διαφορά 1:
  - DFA: Από κάθε κατάσταση του ντετερμινιστικού αυτόματου φεύγει ακριβώς ένα βέλος για κάθε σύμβολο του αλφαβήτου
  - NFA: Από κάθε κατάσταση μπορούν να εκκινούν **μηδέν, ένα, ή περισσότερα βέλη για κάθε σύμβολο** του αλφαβήτου
    - **Δέντρο Υπολογισμού**
- Διαφορά 2:
  - **Προσθέτει το σύμβολο e** στο αλφάβητο.

Ισοδυναμία NFA με DFA

Θεώρημα:
Για κάθε μη ντετερμινιστικό πεπερασμένο αυτόματο, υπάρχει ισοδύναμο ντετερμινιστικό.

- Κατασκευή ενός ντετερμινιστικού αυτόματου (DFA) που να προσομοιώνει το μη ντετερμινιστικό (NFA)

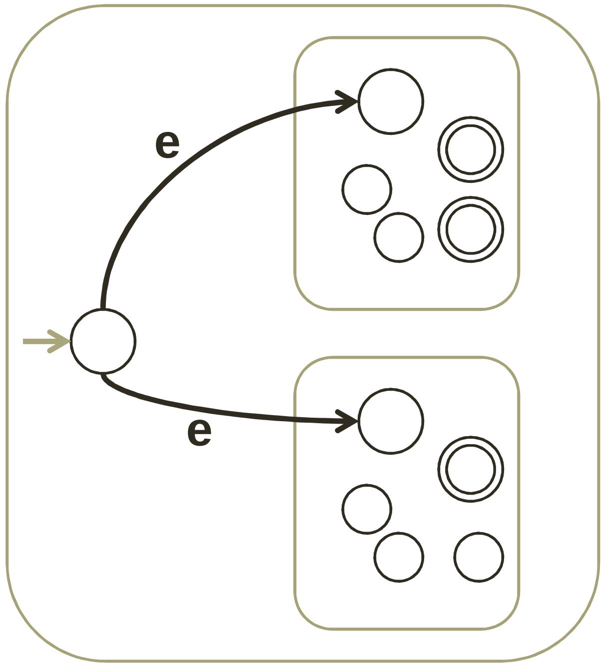
Παράδειγμα: NFA to DFA



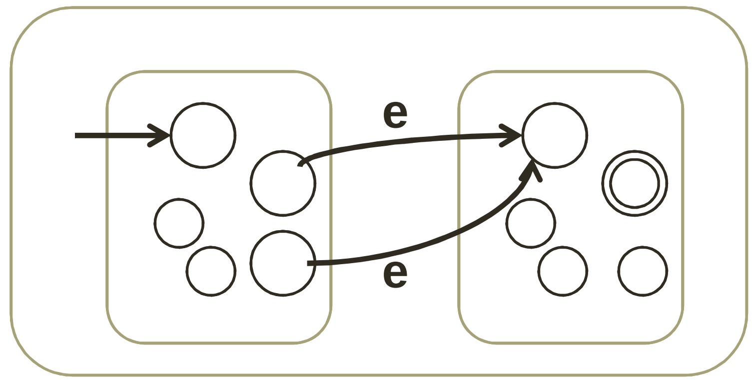
- $M(Q, \Sigma, \delta, q_0, F)$
  - $Q = \{ \emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\} \}$
  - $\Sigma = \{a, b\}$
  - $q_0 = E(\{1\}) = \{1,3\}$
  - $F = \{\{1\}, \{1,2\}, \{1,3\}, \{1,2,3\}\}$
  - δ?

Κατασκευές Κλειστότητας

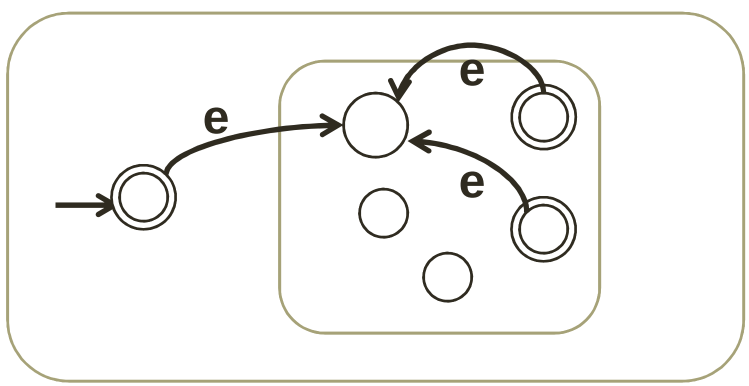
L(N) = A ∪B



L(N) = AB



L(N) = A*



Κανονικές Εκφράσεις

Ορισμός

R ***είναι μια*** ***κανονική έκφραση*** ***αν είναι της μορφής***

1. ***α, όπου α είναι ένα σύμβολο του αλφαβήτου Σ***
2. ***e,***
3. $\emptyset$
4. $(R_1 \cup R_2)$ ***,όπου R_1 και R_2 δύο κανονικές εκφράσεις***
5. $(R_1 \circ R_2)$ ***,όπου R_1 και R_2 δύο κανονικές εκφράσεις***
6. R_1^* ***,όπου R_1 μια κανονική έκφραση***

Ισοδυναμία με πεπερασμένα αυτόματα

- Κατασκευή NFA από Κανονική έκφραση (RegEx => NFA)
  - Χρήση κατασκευών κλειστότητας ένωσης, σύναρμογής, σώρευσης

- Κατασκευή Κανονικής έκφρασης από DFA (DFA => RegEx)
  - DFA -> GNFA -> RegEx
  - Μετατροπή από DFA σε **γενικευμένο μη ντετερμινιστικό πεπερασμένο αυτόματο** (GNFA), και από GNFA σε κανονική έκφραση.

  - **Θεώρημα:** Μια γλώσσα είναι κανονική αν υπάρχει πεπερασμένο αυτόματο που την αποδέχεται ή κανονική έκφραση που την παράγει.

Μη Κανονικές Γλώσσες

Λήμμα της Άντλησης:
Για κάθε κανονική γλώσσα A, υπάρχει αριθμός p (το μήκος της άντλησης αυτής) τέτοιος ώστε κάθε λέξη w της A με μήκος μεγαλύτερο ή ίσο του p να μπορεί να χωριστεί σε τρία τμήματα, w = xyz, που να ικανοποιούν τις εξής συνθήκες:

1. ***για κάθε $i \geq 0$, $xy^iz \in A$***
2. ***$|y| > 0$, και***
3. ***$|xy| \leq p$.***

Παράδειγμα

- $L = \{ww^R \mid w \text{ in } \{0,1\}^* \}$

Ασυμφραστικές Γλώσσες

- Ασυμφραστικές Γραμματικές

- Αυτόματα με Στοίβα

- Μη ασυμφραστικές γλώσσες (Λήμμα Άντλησης)

Ασυμφραστικές Γραμματικές

- Συλλογή **Κανόνων Αντικατάστασης**
  - **Μη-τερματικά σύμβολα παράγουν λέξεις από τερματικά και μη-τερματικά σύμβολα**
  - **Παραγωγή Συντακτικού Δέντρου**
  - **Αντικαθιστούμε τα μη τερματικά σύμβολα με βάση τους κανόνες μέχρις ότου η λέξη να περιέχει μόνο τερματικά σύμβολα.**

- **Σχεδίαση CFG**
  - **Απλοποίηση σύνθετων γλωσσών**
  - **Κανονική Γλώσσα**
    - **Μετατροπή από Πεπερασμένο Αυτόματο σε Γραμματική**
    - **Μετατροπή Γραμματικής σε Πεπερασμένο Αυτόματο**
  - **Αλληλένδετες υπολέξεις**
  - **Αναδρομικές Δομές**

Ασυμφραστικές Γραμματικές

- Πολυτροπία
  - Παραγωγή ίδιας λέξης με διαφορετικές αντικαταστάσεις
  - Εξ αριστερών παραγωγές

- Κανονική Μορφή Chomsky
  - Κάθε κανόνας βρίσκεται σε μια από τις μορφές

$$
\begin{aligned}
A &\rightarrow BC \\
A &\rightarrow a
\end{aligned}
$$

- **Θεώρημα:** Κάθε ασυμφραστική γραμματική μπορεί να μετατραπεί σε κανονική μορφή Chomsky.
  - Βήματα Μετατροπής

Αυτόματα με Στοίβα

- **Μη Ντετερμινιστικό Πεπερασμένο Αυτόματο** με επιπλέον «εξάρτημα» - **ΣΤΟΙΒΑ**

Ορισμός

***Αυτόματο στοίβας* είναι μια εξάδα** $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$

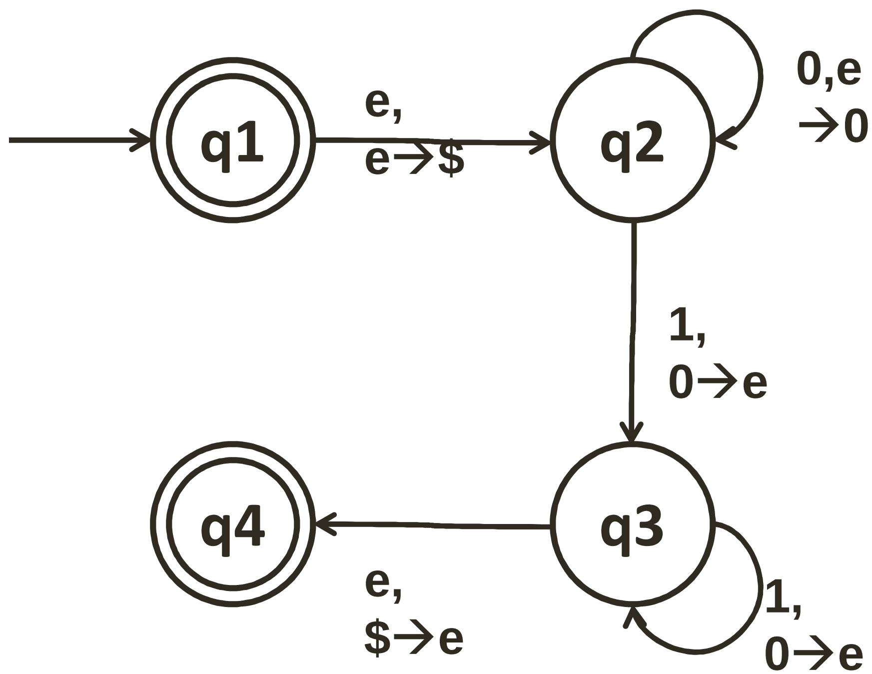
1. ***Q* = *πεπερασμένο* σύνολο καταστάσεων**
2. Σ_e ***= αλφάβητο εισόδου***
3. Γ_e ***= αλφάβητο στοίβας***
4. ***Συνάρτηση μετάβασης*** δ: $Q \times \Sigma_e \times \Gamma_e \rightarrow P(Q \times \Gamma_e)$
   Όταν το αυτόματο είναι στην κατάσταση $q \in Q$ ***και διαβάσει το σύμβολο*** $a \in \Sigma$ ***και υπάρχει σύμβολο*** $\gamma \in \Gamma$ ***μεταβιβάζεται σε ένα υποσύνολο καταστάσεων με αντίστοιχα σύμβολα στην κορυφή της στοίβας***
   ***π.χ.* δ (q,α,γ)={(q’,γ’), (q’’,γ’’)}**
5. $q_o \in Q$ ***= εναρκτήρια κατάσταση***
6. $F \subseteq Q$ ***= σύνολο τελικών καταστάσεων***

Δρ. Νικόλας Νικολάου

Παράδειγμα 1

- Γλώσσα L={0^n1^n | n≥0}

$$Q = \{q_1, q_2, q_3, q_4\}, \Sigma = \{0,1\}, \Gamma = \{0,\$\}, F = \{q_1, q_4\}$$



Το σύμβολο $ μας βοηθά να εντοπίσουμε τον «πάτο» της στοίβας

Ισοδυναμία PDA με CFG

- **Θεώρημα:** Κάθε ασυμφραστική γραμματική μπορεί να μετατραπεί σε ένα ισοδύναμο αυτόματο στοίβας
  - Κατασκευαστική Απόδειξη

- **Θεώρημα:** Μια γλώσσα είναι ασυμφραστική εάν και μόνο εάν υπάρχει αυτόματο στοίβας που την αναγνωρίζει και ασυμφραστική γραμματική που την παράγει.

Μη Ασυμφραστικές Γλώσσες

Λήμμα της Άντλησης για ασυμφραστικές γλώσσες:
Για κάθε ασυμφραστική γλώσσα A, υπάρχει αριθμός p (το μήκος της άντλησης αυτής) τέτοιος ώστε κάθε λέξη w με μήκος μεγαλύτερο ή ίσο του p να μπορεί να χωριστεί σε πέντε τμήματα, w = uvxyz, που να ικανοποιούν τις εξής συνθήκες:

1. *για κάθε $i \geq 0$, $uv^ixy^iz \in A$*
2. *$|vy| > 0$, και*
3. *$|vxy| \leq p$.*

Πιο ισχυρά μοντέλα υπολογισμού?

- **Μηχανές Turing**
  - μπορούν να διαβάζουν και να γράφουν στην ταινία εισόδου
  - και να μετακινούνται σε αυτή και αριστερά
- Ντετερμινιστικές Μηχανές Turing
  - όπως τα ντετερμινιστικά πεπερασμένα αυτόματα
    - πρέπει να ορίζονται όλες οι δυνατές μεταβάσεις
- Μη Ντετερμινιστικές Μηχανές Turing
  - όπως τα μη-ντετερμινιστικά πεπερασμένα αυτόματα
    - ορίζονται μόνο οι επιτρεπτές μεταβάσεις

Διαγνώσιμες (Αναδρομικές) Γλώσσες

- Μια TM M **διαγιγνώσκει** μια γλώσσα $L \subseteq \Sigma_0^*$ αν για κάθε λέξη $w \in \Sigma^*_0$ είναι αληθές ότι:
  - Αν $w \in L$, η M αποδέχεται την w (φτάνει στην κατάσταση αποδοχής) και
  - αν $w \notin L$ η M απορρίπτει την w (φτάνει στην κατάσταση απόρριψης).

- Μια γλώσσα είναι **αδιαγνώσιμη (αναδρομική)** αν υπάρχει μια μηχανή που την αποφασίζει.

Αναγνωρίσιμες (Αναδρομικά Απαριθμήσιμες) Γλώσσες

- Η Μ **αναγνωρίζει** την L αν για κάθε λέξη $w \in \Sigma^*_0$ αληθεύει το εξής:
  - $w \in L$ **αν και μόνο αν** η Μ **τερματίζει και αποδέχεται** με είσοδο w.

- Παρατηρείστε.
  - Αν η Μ έχει είσοδο $w \in L$ τερματίζει
  - Αν όμως $w \notin L$ τότε η Μ μπορεί να απορρίψει ή να εγκλωβιστεί και να μην τερματίσει ποτέ.

- Μια γλώσσα L είναι **αναγνωρίσιμη (αναδρομικά απαριθμήσιμη)** αν υπάρχει μια μηχανή Μ που αναγνωρίζει την L.

- Μια γλώσσα L είναι **συμπληρωματικά αναγνωρίσιμη (συναναδρομικά απαριθμήσιμη)** αν υπάρχει μια μηχανή Μ που αναγνωρίζει το συμπλήρωμα της L.

Παραλλαγές TM

- Πολυταινιακές TM
- **Θεώρημα:** Για κάθε μηχανή Turing πολλαπλών κεφαλών υπάρχει ισοδύναμη μονοταινιακή μηχανή Turing

- Μη Ντετερμινιστικές TM
- **Θεώρημα:** Για κάθε μη ντετερμινιστική μηχανή Turing υπάρχει ισοδύναμη μονοταινιακή μηχανή Turing

- Καθολικές TM
- Προσομοιώνουν άλλες TM
- Δέχονται ως είσοδο την κωδικοποίηση άλλων TM

Δόγμα Church - Turing

- **Δόγμα Church-Turing (1936):**
    - ότι μπορεί να υπολογιστεί (υπάρχει αλγόριθμος) μπορεί να υπολογιστεί από μια μηχανή Turing.
    - Δηλ. Αλγόριθμος είναι μια μηχανή Turing που τερματίζει για όλες τις εισόδους του προβλήματος
    - Τα προβλήματα που δεν μπορούν να επιλυθούν από μια μηχανή Turing δεν μπορούν να επιλυθούν. (είναι **μη επιλύσιμα**).

- Περιγραφή αλγορίθμων με TM
    - Κωδικοποίηση εισόδου <…>

Το πρόβλημα του τερματισμού

- ΑΠΟΔΟΧΗ/TM={<M,w> | M είναι μια TM που αποδέχεται την λέξη w}

- **Θεώρημα:** Η ΑΠΟΔΟΧΗ/TM είναι αναγνωρίσιμη αλλά όχι διαγνώσιμη.

- **Θεώρημα:** Η $\overline{\mathrm{ΑΠΟΔΟΧΗ/TM}}$ είναι μη αναγνωρίσιμη

Συνέπεια: Πολλά Μη-επιλύσημα προβλήματα

- Ορισμός. (**Απεικονιστική Αναγωγιμότητα**) Έστω $L_1, L_2 \subseteq \Sigma^*$ δύο γλώσσες. Μια **αναγωγή από την** L_1 **στην** L_2, $L_1 \leq L_2$, είναι μια υπολογίσιμη (αναδρομική) συνάρτηση $\tau: \Sigma^* \to \Sigma^*$ τέτοια ώστε $x \in L_1$ αν και μόνο αν $\tau(x) \in L_2$.
- **Χρήση**: **Για να δείξουμε ότι η L_2 δεν είναι διαγνώσιμη:**
  - Προσδιορίζω μια γλώσσα L_1 που είναι γνωστό ότι είναι μη διαγνώσιμη (μη αναδρομική)
  - Ανάγω την L_1 στην L_2.

Γνωστή Μη Αναδρομική Γλώσσα L_1

αναδρομική συνάρτηση τ:

$x \in L_1$ αν και μόνο αν $\tau(x) \in L_2$

(Αναδρομική?)

Γλώσσα L_2

Χρήση Αναγωγών

- Θεώρημα 1. Εάν A $\leq_m$ B και η B είναι διαγνώσιμη (αναδρομική) τότε και η A είναι διαγνώσιμη (αναδρομική).

- Θεώρημα 2. Εάν A $\leq_m$ B και η A είναι μη διαγνώσιμη τότε και η B είναι μη διαγνώσιμη.

- Θεώρημα 3. Εάν A $\leq_m$ B και η B είναι αναγνωρίσιμη(αναδρομικά απαριθμήσιμη) τότε και η A είναι αναγνωρίσιμη (αναδρομικά απαριθμήσιμη).

- Θεώρημα 4. Εάν A $\leq_m$ B και η A δεν είναι αναγνωρίσιμη(μη αναδρομικά απαριθμήσιμη) τότε και η B δεν είναι αναγνωρίσιμη (μη αναδρομικά απαριθμήσιμη).

Πολυπλοκότητα Προβλημάτων

- Επιλύσιμα προβλήματα
  - Πόσους υπολογιστικούς πόρους απαιτούν;
  - **Χρόνο:** Πόσα βήματα απαιτούνται από μια TM για την επίλυση του προβλήματος.
  - **Χώρο:** Πόσα κελιά πάνω στην ταινία μιας TM απαιτούνται για την επίλυση του προβλήματος.
- Κλάσεις Γλωσσών
  - Χρονικής Πολυπλοκότητας: P, NP,EXP
  - Χωρικής Πολυπλοκότητας: PSPACE, NPSPACE
- Αναγωγές πολυωνιμικού χρόνου
  - NP-πληρότητα

Κλάσεις Πολυπλοκότητας

$$
\begin{aligned}
P &= \bigcup_{c>0} DTIME(n^c) \\
EXP &= \bigcup_{c>0} DTIME(2^{cn}) \\
EXPPOLY &= \bigcup_{c>0} DTIME(2^{n^c}) \\
PSPACE &= \bigcup_{c>0} DSPACE(n^c)
\end{aligned}
$$

Ερωτήσεις;