Θεωρία Υπολογισμού

Μη Επιλυσιμότητα (Undecidability – Unsolvability)

2003-04

1

Μη Επιλυσιμότητα

- Υπάρχουν γλώσσες για τις οποίες **δεν υπάρχει** μηχανή Turing που να τις αποφασίζει (ή να τις ημιαποφασίζει).
- Υπάρχουν συναρτήσεις για τις οποίες **δεν υπάρχει** μηχανή Turing που να τις υπολογίζει.
- Υπάρχουν προβλήματα που **δεν μπορούν να λυθούν** από κάποιο αλγόριθμο.

2

Καθολικές (Universal) Μηχανές Turing

- Θα κατασκευάσουμε μια **καθολική μηχανή Turing** U με τις εξής ιδιότητες. Η U παίρνει σαν είσοδο μια αναπαράσταση "M" μιας μηχανής Turing M και μια αναπαράσταση "w" μιας συμβολοσειράς εισόδου w και προσομοιώνει τη λειτουργία της M με είσοδο w.
- Ακριβέστερα, θα ισχύει η ακόλουθη πρόταση: Η U με είσοδο "M""w" τερματίζει αν και μόνο αν η M με είσοδο "w" τερματίζει. Επίσης

$$U(\text{“}M\text{”}\text{“}w\text{”}) = \text{“}M(w)\text{”}.$$

3

Αναπαράσταση Μηχανών Turing από Συμβολοσειρές

- Ας είναι $M = (K, \Sigma, \delta, s, H)$ μια μηχανή Turing, και i, j οι μικρότεροι ακέραιοι τέτοιοι ώστε $2^i \geq |K|$ και $2^j \geq |\Sigma| + 2$.
- Τα στοιχεία της μηχανής Turing M μπορούν να αναπαρασταθούν με τον εξής τρόπο:
  - Κάθε κατάσταση θα παριστάνεται με ένα q ακολουθούμενο από μια συμβολοσειρά μήκους i από το αλφάβητο $\{0, 1\}$.
  - Η αρχική κατάσταση θα παριστάνεται πάντα από τη λεξικογραφικά μικρότερη συμβολοσειρά $q0^i$.

4

Αναπαράσταση Μηχανών Turing από Συμβολοσειρές

- Κάθε σύμβολο του αλφάβητου θα παριστάνεται με ένα a ακολουθούμενο από μια συμβολοσειρά μήκους j από το αλφάβητο $\{0, 1\}$.
- Το σύμβολο $\sqcup$ θα παριστάνεται πάντα από τη λεξικογραφικά μικρότερη συμβολοσειρά $a0^j$, το σύμβολο $\triangleright$ από την $a0^{j-1}1$, το σύμβολο $\leftarrow$ από την $a0^{j-2}11$ και το σύμβολο $\rightarrow$ από την $a0^{j-3}111$.

5

Αναπαράσταση Μηχανών Turing από Συμβολοσειρές

- Η **αναπαράσταση** "M" της μηχανής Turing M θα είναι μια ακολουθία από τετράδες (Q, A, P, B) όπου Q, P είναι αναπαραστάσεις καταστάσεων και A, B είναι αναπαραστάσεις συμβόλων του αλφάβητου (έτσι έχουμε την αναπαράσταση της συνάρτησης μετάβασης).
- Υιοθετούμε τη σύμβαση ότι αυτές οι τετράδες δίνονται με λεξικογραφικά αύξουσα σειρά, αρχίζοντας με την τετράδα (Q, A, P, B) όπου Q είναι η αναπαράσταση της αρχικής κατάστασταση s και A είναι η αναπαράσταση του συμβόλου $\sqcup$.
- Με τον ίδιο τρόπο αναπαριστάνουμε συμβολοσειρές του αλφάβητου Σ. Έτσι κάθε $w \in \Sigma^*$ θα έχει μια μοναδική αναπαράσταση "w".

6

Παράδειγμα

$$M = (K, \Sigma, \delta, s, \{h\})$$

- όπου $K = \{s, q, h\}$, $\Sigma = \{a, \sqcup, \triangleright\}$ και η δ δίνεται από τον πίνακα

κατάσταση,	σύμβολο	δ
s	a	$(q, \sqcup)$
s	$\sqcup$	$(h, \sqcup)$
s	$\triangleright$	$(s, \rightarrow)$
q	a	(s, a)
q	$\sqcup$	$(s, \rightarrow)$
q	$\triangleright$	$(q, \rightarrow)$

7

Αναπαράσταση της M

- Μπορούμε να πάρουμε $i = 2$ και $j = 3$. Οι καταστάσεις και τα σύμβολα της M αναπαριστάνονται ως εξής:

κατάσταση/σύμβολο	αναπαράσταση
s	$q00$
q	$q01$
h	$q11$
$\sqcup$	$a000$
$\triangleright$	$a001$
$\leftarrow$	$a010$
$\rightarrow$	$a011$
a	$a100$

8

Αναπαράσταση της M

- Η αναπαράσταση της συμβολοσειράς $\triangleright aa \sqcup a$ είναι

$$\text{“}\triangleright aa \sqcup a\text{”} = a001a100a100a000a100$$

- Η αναπαράσταση της μηχανής M είναι

$$\text{“}M\text{”} = (q00, a100, q01, a000), (q00, a000, q11, a000), (q00, a001, q00, a011),$$
$$(q01, a100, q00, a011), (q01, a000, q00, a011), (q01, a001, q01, a011)$$

9

Μια Καθολική Μηχανή Turing

- Θα περιγράψουμε μια καθολική μηχανή U με 3 ταινίες. Η U χρησιμοποιεί τις ταινίες της ως εξής:
  - Η πρώτη ταινία περιέχει την αναπαράσταση της ταινίας της M.
  - Η δεύτερη ταινία περιέχει την αναπαράσταση της M.
  - Η τρίτη ταινία περιέχει την αναπαράσταση της κατάστασης της M.

10

Μια Καθολική Μηχανή Turing

- Η μηχανή U ξεκινάει με είσοδο την συμβολοσειρά “M”“w” (στην πρώτη ταινία) και λειτουργεί ως εξής:
  - Μετακίνησε την συμβολοσειρά “M” στην δεύτερη ταινία, και μετατόπισε την “w” αριστερά ώστε η αρχή της πρώτης ταινίας να περιέχει την συμβολοσειρά $\triangleright \sqcup$ “w”.
  - Γράψε στην τρίτη ταινία την αναπαράσταση της αρχικής κατάστασης της M.
  - Προσομοίωσε τη λειτουργία της M.

11

Μια Καθολική Μηχανή Turing

- Η προσομοίωση της λειτουργίας της M γίνεται ως εξής:
  - Σάρωσε την δεύτερη ταινία μέχρι να βρείς μια τετράδα (Q, A, P, B) όπου Q είναι η αναπαράσταση της κατάστασης που είναι γραμμένη στην τρίτη ταινία, και A είναι η αναπαράσταση του συμβόλου που σαρώνεται στην πρώτη ταινία.
  - Αν βρείς τέτοια τετράδα (Q, A, P, B), άλλαξε την κατάσταση σε P (ενημέρωσε την τρίτη ταινία), και εκτέλεσε στην πρώτη ταινία την πράξη που παριστάνει το B.
  - Αν δεν βρείς τέτοια τετράδα (Q, A, P, B), τερμάτισε (η Q είναι τελική κατάσταση της M).

12

Το Πρόβλημα Τερματισμού

- Έστω ένα πρόγραμμα halts(P, X) γραμμένο σε κάποια γλώσσα προγραμματισμού που έχει την εξής λειτουργία:
  - **Είσοδος:** Ένα πρόγραμμα P γραμμένο στην ίδια γλώσσα προγραμματισμού με το halts, και μια είσοδος Q του P.
  - **Έξοδος:** “ναι” αν το πρόγραμμα P τερματίζει με είσοδο X, αλλιώς “όχι”.
- Το πρόγραμμα halts είναι πολύ χρήσιμο (αν υπάρχει!)

13

Το Πρόβλημα Τερματισμού

- Έστω τώρα το εξής πρόγραμμα:

$$\mathsf{diagonal}(X)$$
$$a:\ \mathsf{if}\ \mathsf{halts}(X, X)\ \mathsf{then\ goto}\ a\ \mathsf{else}\ halt$$

- Το πρόγραμμα diagonal(Q) λειτουργεί ως εξής: Αν το πρόγραμμα halts αποφασίζει ότι το πρόγραμμα Q τερματίζει με είσοδο τον εαυτό του, τότε το πρόγραμμα diagonal πέφτει σε ένα άπειρο loop, διαφορετικά τερματίζει.

14

Το Πρόβλημα Τερματισμού

- **Ερώτηση:** Τερματίζει το πρόγραμμα diagonal(diagonal);
- **Απάντηση:** Το πρόγραμμα diagonal(diagonal) τερματίζει αν και μόνο αν δεν τερματίζει. **Αδύνατο!**
- **Αρα το αρχικό πρόγραμμα halts δεν μπορεί να υπάρχει!**

15

Το Πρόβλημα Τερματισμού για Μηχανές Turing

- Έστω η γλώσσα

$$H = \{\text{“}M\text{”“}w\text{”} :\ \text{η μηχανή Turing } M \text{ τερματίζει με είσοδο } w\ \}.$$

- Η γλώσσα H ημιαποφασίζεται από την καθολική μηχανή Turing U. Αρα η H **είναι αναδρομικά αριθμήσιμη**.
- **Πρόταση.** Η κλάση των αναδρομικά αριθμήσιμων γλωσσών είναι η ίδια με την κλάση των αναδρομικών γλωσσών αν και μόνο αν η γλώσσα H είναι αναδρομική.

16

Απόδειξη

(Αν)

- Υποθέτουμε ότι η γλώσσα H είναι αναδρομική. Τότε υπάρχει μια μηχανή Turing M_0 που την αποφασίζει.
- Έστω L μια αναδρομικά αριθμήσιμη γλώσσα και M μια μηχανή Turing που την ημιαποφασίζει. Μπορούμε να κατασκευάσουμε μια μηχανή Turing M' που **αποφασίζει** την L ως εξής.
- Η M' αρχικά μετατρέπει την ταινία της από $\triangleright\sqcup w\sqcup$ σε $\triangleright\sqcup$"M""w"$\sqcup$ και μετά προσομοιώνει την M_0 με αυτή την είσοδο. Αν η M_0 τερματίζει σε κατάσταση αποδοχής, τότε η M' επίσης τερματίζει σε κατάσταση αποδοχής. Διαφορετικά τερματίζει σε κατάσταση απόρριψης.
- Αρα η κλάση των αναδρομικά αριθμήσιμων γλωσσών είναι ίση με την κλάση των αναδρομικών γλωσσών.

(Μόνο αν) Προφανές.

Το Πρόβλημα Τερματισμού για Μηχανές Turing

Είναι όμως η H αναδρομική;

- Αν η H είναι αναδρομική, τότε η γλώσσα

$$H_1 = \{\text{“}M\text{”}: \ \text{η μηχανή Turing } M \text{ τερματίζει με είσοδο “}M\text{” }\}.$$

  είναι επίσης αναδρομική.
- Αν υπάρχει μια μηχανή Turing M_0 η οποία αποφασίζει την H, τότε μια μηχανή Turing M_1 που αποφασίζει την H_1 μπορεί να λειτουργεί ως εξής:
  - Μετέτρεψε την ταινία από $\triangleright\sqcup$"M"$\sqcup$ σε $\triangleright\sqcup$"M""M"$\sqcup$.
  - Προσομοίωσε την M_0.

Το Πρόβλημα Τερματισμού για Μηχανές Turing

- Αν η H_1 είναι αναδρομική, τότε και το συμπλήρωμα της

$$\begin{aligned}\overline{H_1} = \{\ w: \ & \text{η } w \text{ δεν είναι η αναπαράσταση μιας μηχανής Turing,}\\ & \text{ή είναι η αναπαράσταση “}M\text{” μιας μηχανής Turing } M\\ & \text{που δεν τερματίζει με είσοδο “}M\text{” }\}\end{aligned}$$

  είναι επίσης μια αναδρομική γλώσσα.
- Στην πραγματικότητα ομως η $\overline{H_1}$ δεν είναι **ούτε αναδρομικά αριθμήσιμη!**

Το Πρόβλημα Τερματισμού για Μηχανές Turing

- Έστω μια μηχανή Turing M^* που ημιαποφασίζει την $\overline{H_1}$. Τότε

$$\text{“}M^*\text{”} \in \overline{H_1} \text{ αν και μόνο αν η } M^* \text{ δέχεται την συμβολοσειρά “}M^*\text{”}$$

- Από τον ορισμό της $\overline{H_1}$ έχουμε

$$\text{“}M^*\text{”} \in \overline{H_1} \text{ αν και μόνο αν η } M^* \text{ δεν δέχεται την συμβολοσειρά “}M^*\text{”}$$

- Εχουμε λοιπόν
  - η M^* δέχεται την συμβολοσειρά "M^*" αν και μόνο αν η M^* δεν δέχεται την συμβολοσειρά "M^*"
- **Αδύνατο!**

Το Πρόβλημα Τερματισμού για Μηχανές Turing

- Άρα έχουμε αποδείξει το ακόλουθο αποτέλεσμα:
- **Θεώρημα.** Η γλώσσα H δεν είναι αναδρομική. Αρα η κλάση των αναδρομικά αριθμήσιμων γλωσσών είναι ένα γνήσιο υποσύνολο της κλάσης των αναδρομικών γλωσσών.

Το Πρόβλημα Τερματισμού για Μηχανές Turing

- **Θεώρημα.** Αν μια γλώσσα δεν είναι αναδρομική το ίδιο ισχύει και για το συμπλήρωμα της.
- **Θεώρημα.** Η κλάση των αναδρομικά αριθμήσιμων γλωσσών δεν είναι κλειστή ως προς την πράξη της συμπλήρωσης.
- **Απόδειξη.** Η γλώσσα H_1 είναι αναδρομικά αριθμήσιμη αλλά το συμπλήρωμα της ($\overline{H_1}$) δεν είναι αναδρομικά αριθμήσιμη γλώσσα.

Η Αρχή της Διαγωνιοποίησης

- Έστω R μια δυαδική σχέση και $R \subseteq A \times A$. Το **διαγώνιο σύνολο** της R είναι

$$D = \{a: \ a \in A \text{ και } (a,a) \notin R\}.$$

- Για κάθε $a \in A$, έστω

$$R_a = \{b: \ b \in A \text{ και } (a,b) \in R\}.$$

  Τότε το σύνολο D είναι διαφορετικό από κάθε σύνολο R_a.

Εναλλακτική Απόδειξη

- Έστω η παρακάτω δυαδική σχέση R που ορίζεται στο σύνολο των συμβολοσειρών του αλφάβητου που χρησιμοποιήσαμε για την αναπαράσταση μηχανών Turing:
- $(u,w) \in R$ αν και μόνο αν $u =$ "M" για κάποια μηχανή Turing M που τερματίζει με είσοδο την συμβολοσειρά με αναπαράσταση w.
- Για κάθε συμβολοσειρά u ορίζουμε το σύνολο

$$R_u = \{w: \ (u,w) \in R\}.$$

  Τα σύνολα R_u αντιστοιχούν στις αναδρομικά αριθμήσιμες γλώσσες.

Εναλλακτική Απόδειξη

- Το **διαγώνιο σύνολο** της R είναι
$$D = \{w : (w, w) \notin R\}.$$
Το σύνολο D είναι μια γλώσσα. Σύμφωνα με την αρχή της διαγωνιοποίησης, η γλώσσα D είναι διαφορετική από κάθε γλώσσα R_u δηλ. δεν είναι αναδρομικά αριθμήσιμη.

Μη Επιλύσιμα Προβλήματα

- Η προηγούμενη συζήτηση μπορεί να συνοψιστεί ως εξής:
- **Δεν υπάρχει αλγόριθμος που για μια τυχαία μηχανή Turing M και συμβολοσειρά εισόδου w να αποφασίζει αν η M τερματίζει με είσοδο w.**
- Προβλήματα για τα οποία δεν υπάρχει αλγόριθμος (ισοδύναμα: οι γλώσσες που τα κωδικοποιούν δεν είναι αναδρομικές) λέγονται **μη επιλύσιμα** (undecidable or unsolvable).

Μη Επιλύσιμα Προβλήματα

- Πως αποδεικνύουμε ότι ένα πρόβλημα Π_2 δεν είναι επιλύσιμο;
  - Με κάποια τεχνική που είναι συγκεκριμένη για το Π_2 (π.χ., διαγωνιοποίηση στην περίπτωση του προβλήματος τερματισμού για μηχανές Turing).
  - Με **αναγωγή** (reduction) από ένα άλλο πρόβλημα Π_1 που ήδη ξέρουμε ότι δεν είναι επιλύσιμο.
    Σ' αυτή την περίπτωση δείχνουμε ότι αν το πρόβλημα Π_2 ήταν επιλύσιμο τότε και το Π_1 θα ήταν επιλύσιμο (αδύνατο!).

Μη Επιλύσιμα Προβλήματα

- **Ορισμός.** Έστω γλώσσες $L_1, L_2 \subseteq \Sigma^*$. Μια **αναγωγή** από την L_1 στην L_2 είναι μια αναδρομική συνάρτηση $\tau : \Sigma^* \to \Sigma^*$ τέτοια ώστε $x \in L_1$ αν και μόνο αν $\tau(x) \in L_2$.
- **Θεώρημα.** Αν η γλώσσα L_1 δεν είναι αναδρομική και υπάρχει αναγωγή από την L_1 στην L_2, τότε και η L_2 δεν είναι αναδρομική.
- **Απόδειξη.** Έστω ότι η L_2 είναι αναδρομική και αποφασίζεται από μια μηχανή Turing M_2. Έστω T η μηχανή Turing που αποφασίζει την συνάρτηση τ. Τότε η μηχανή Turing TM_2 αποφασίζει την L_1. Αυτό όμως είναι αδύνατο επειδή η L_1 δεν είναι αναδρομική.

Μη Επιλύσιμα Προβλήματα για Μηχανές Turing

- **Θεώρημα.** Τα παρακάτω προβλήματα σχετικά με μηχανές Turing είναι μη επιλύσιμα.
  (a) Αν δοθεί μια μηχανή Turing M και μια συμβολοσειρά εισόδου w, τερματίζει η M με είσοδο w;
  (b) Αν δοθεί μια μηχανή Turing M, τερματίζει η M με είσοδο e;
  (c) Αν δοθεί μια μηχανή Turing M, υπάρχει κάποια συμβολοσειρά εισόδου για την οποία η M τερματίζει;
  (d) Αν δοθεί μια μηχανή Turing M, τερματίζει η M για κάθε συμβολοσειρά εισόδου;
  (e) Αν δοθούν δυό μηχανές Turing M_1 και M_2, τερματίζουν οι M_1 και M_2 για τις ίδιες συμβολοσειρές εισόδου;
  (f) Αν δοθεί μια μηχανή Turing M, είναι η γλώσσα την οποία ημιαποφασίζει η M κανονική; Είναι γλώσσα χωρίς συμφραζόμενα; Είναι αναδρομική;
  (g) Υπάρχει μια συγκεκριμένη μηχανή Turing M_0 για την οποία το ακόλουθο πρόβλημα είναι μη επιλύσιμο: Αν w είναι μια δοσμένη συμβολοσειρά εισόδου, τερματίζει η M_0 με είσοδο w;

Απόδειξη

- **(a)** Έχει ήδη αποδειχτεί.
- **(b)** Περιγράφουμε μια αναγωγή τ από την μη αναδρομική γλώσσα
$$H = \{\text{“}M\text{”}\text{“}w\text{”} : \text{ η μηχανή Turing } M \text{ τερματίζει με είσοδο } w\ \}$$
στη γλώσσα που κωδικοποιεί το δοσμένο πρόβλημα:
$$L = \{\text{“}M\text{”} : \text{ η } M \text{ τερματίζει με είσοδο } e\ \}$$
Η τ αντιστοιχεί κάθε συμβολοσειρά “M”“w” στη συμβολοσειρά “M_w” όπου M_w είναι η μηχανή που όταν ξεκινάει με ταινία $\triangleright\sqcup$, γράφει w στην ταινία της και μετά προσομοιώνει την M.

Απόδειξη

- **(b) συνέχεια** Αν $w = a_1 a_2 \cdots a_n$ τότε η M_w είναι η μηχανή
$$Ra_1 Ra_2 \cdots Ra_n L_\sqcup M.$$
Παρατηρούμε τώρα ότι η τ είναι αναδρομική (μπορεί εύκολα να υπολογιστεί από μια μηχανή Turing) και
$$x = \text{“}M\text{”}\text{“}w\text{”} \in H \text{ αν και μόνο αν } \tau(x) = \text{“}M_w\text{”} \in L.$$
Αρα η L είναι μια μη αναδρομική γλώσσα.

Απόδειξη

- **(c)** Περιγράφουμε μια αναγωγή τ από την γλώσσα της περίπτωσης (b)
$$L = \{\text{“}M\text{”} : \text{ η } M \text{ τερματίζει με είσοδο } e\ \}$$
στην γλώσσα του δοσμένου προβλήματος:
$$L' = \{\text{“}M'\text{”} : \text{ η } M' \text{ τερματίζει με κάποια είσοδο }\}$$
Αν δοθεί η αναπαράσταση “M” μιας μηχανής Turing M, η τ(“M”) είναι η αναπαράσταση μιας μηχανής Turing M' που σβήνει κάθε είσοδο που της δίνεται, και μετά προσομοιώνει την M με είσοδο e.

Απόδειξη

- **(c) συνέχεια** Η M' τερματίζει με είσοδο κάποια συμβολοσειρά αν και μόνο αν η M τερματίζει με είσοδο e.
  Αρα $x \in L$ αν και μόνο αν $\tau(x) \in L'$. Επίσης η τ είναι αναδρομική.
  Τελικά η L' δεν είναι αναδρομική γλώσσα.

Απόδειξη

- **(d)** Η αναγωγή είναι ίδια με την περίπτωση (c).
  Παρατηρήστε ότι η M' τερματίζει με είσοδο οποιαδήποτε συμβολοσειρά αν και μόνο αν η M τερματίζει με είσοδο e.

Απόδειξη

- **(e)** Περιγράφουμε μια αναγωγή τ από την γλώσσα της περίπτωσης (d)

$$L = \{\text{“}M\text{”} : \text{ η } M \text{ τερματίζει με κάθε είσοδο } \}$$

στη γλώσσα του δοσμένου προβλήματος:

$$L' = \{\text{“}M_1\text{”“}M_2\text{”} : \text{ οι } M_1 \text{ και } M_2 \text{ τερματίζουν για τις ίδιες συμβολοσειρές εισόδου } \}$$

Αν δοθεί η αναπαράσταση μιας μηχανής Turing M, τότε

$$\tau(\text{“}M\text{”}) = \text{“}M\text{”“}y\text{”}$$

όπου “y” είναι η αναπαράσταση της μηχανής y που μόλις ξεκινάει μεταφέρεται σε κατάσταση αποδοχής (δηλ. δέχεται κάθε είσοδο).

Απόδειξη

- **(e) συνέχεια** Οι δύο μηχανές M και y δέχονται τις ίδιες συμβολοσειρές εισόδου αν και μόνο αν η M δέχεται όλες τις συμβολοσειρές εισόδου. Αρα $x \in L$ αν και μόνο αν $\tau(x) \in L'$. Επίσης η τ είναι αναδρομική.
  Τελικά η L' δεν είναι αναδρομική γλώσσα.

Απόδειξη

- **(f)** (κανονική γλώσσα) Περιγράφουμε μια αναγωγή τ από το **συμπλήρωμα** της γλώσσας της περίπτωσης (b)

$$L = \{\text{“}M\text{”} : \text{ η } M \text{ δεν τερματίζει με είσοδο } e \}$$

στη γλώσσα του δοσμένου προβλήματος:

$$L_1 = \{\text{“}M_1\text{”} : \text{ η γλώσσα που ημιαποφασίζει η } M_1 \text{ είναι κανονική } \}$$

Απόδειξη

- **(f) συνέχεια** Αν δοθεί η αναπαράσταση μιας μηχανής Turing M, τότε $\tau(\text{“}M\text{”}) = \text{“}M_1\text{”}$ όπου “M_1” είναι η αναπαράσταση μιας μηχανής M_1 που λειτουργεί ως εξής:
  - Σβήσε την συμβολοσειρά εισόδου αφού την αντιγράψεις σε μια βοηθητική ταινία.
  - Προσομοίωσε την λειτουργία της M με είσοδο e.
  - Αν η M τερματίσει, τότε επανέφερε την αρχική συμβολοσειρά εισόδου και προσομοίωσε την λειτουργία της καθολικής μηχανής U.

Απόδειξη

- **(f) συνέχεια** Αν η M δεν τερματίζει με είσοδο e, τότε η M_1 δεν τερματίζει για καμιά είσοδο δηλ. η γλώσσα που ημιαποφασίζει ($\emptyset$) είναι κανονική.
  Αν η M τερματίζει με είσοδο e, τότε η γλώσσα που ημιαποφασίζει η M_1 είναι η H που δεν είναι κανονική.
  Δηλαδή $x \in L$ αν και μόνο αν $\tau(x) \in L_1$. Επίσης η τ είναι αναδρομική.
  Τελικά η L_1 δεν είναι αναδρομική γλώσσα.
  Η αναγωγή είναι η ίδια για τις άλλες δύο περιπτώσεις.

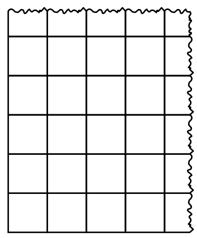
Απόδειξη

- **(g)** Η συγκεκριμένη μηχανή M_0 για την οποία μιλάμε είναι η καθολική μηχανή U.
  Δηλαδή το πρόβλημα της περίπτωσης (g) είναι το ίδιο με το πρόβλημα της περίπτωσης (a).

Μη Επιλύσιμα Προβλήματα για Γραμματικές

- **Θεώρημα.** Τα παρακάτω προβλήματα σχετικά με γραμματικές χωρίς συμφραζόμενα δεν είναι επιλύσιμα.
  (a) Αν δοθεί μια γραμματική χωρίς συμφραζόμενα G, είναι $L(G) = \Sigma^*$;
  (b) Αν δοθούν γραμματικές χωρίς συμφραζόμενα G_1 και G_2, είναι $L(G_1) = L(G_2)$;
  (c) Αν δοθούν αυτόματα στοίβας M_1 και M_2, είναι $L(M_1) = L(M_2)$;
  (d) Αν δοθεί ένα αυτόματο στοίβας M, να βρεθεί ένα ισοδύναμο αυτόματο με τον ελάχιστο αριθμό καταστάσεων.

Ένα Μη Επιλύσιμο Πρόβλημα Πλακόστρωσης



Το Πρόβλημα Πλακόστρωσης

- Ένα **σύστημα πλακόστρωσης** είναι μια τετράδα $\mathcal{D} = (D, d_0, H, V)$, όπου D είναι ένα πεπερασμένο σύνολο (που δείχνει τι πλακάκια έχουμε στη διάθεση μας), $d_0 \in D$, και $H, V \subseteq D \times D$.
- Μια **πλακόστρωση** με το σύστημα $\mathcal{D}$ είναι μια συνάρτηση $f : N \times N \to D$, τέτοια ώστε να ισχύουν τα παρακάτω:

$$f(0,0) = d_0$$
$$(f(n,m), f(n+1,m)) \in H \text{ για όλα τα } n, m \in N$$
$$(f(n,m), f(n,m+1)) \in V \text{ για όλα τα } n, m \in N$$

Το Πρόβλημα Πλακόστρωσης

- **Θεώρημα.** Έστω ότι δίνεται ένα σύστημα πλακόστρωσης $\mathcal{D}$. Το πρόβλημα "Υπάρχει πλακόστρωση με το σύστημα $\mathcal{D}$;" είναι μη επιλύσιμο.
- **Απόδειξη.** Θα δώσουμε μια αναγωγή από το παρακάτω μη επιλύσιμο πρόβλημα (συμπλήρωμα του προβλήματος τερματισμού για μηχανές Turing):
  Αν δοθεί μια μηχανή Turing M, είναι δυνατό η M να μην τερματίζει με είσοδο e;

Παράδειγμα

- Έστω η μηχανή Turing

$$M = (K, \Sigma, \delta, s, \{h\})$$

όπου

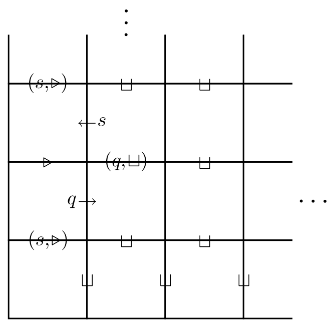
$$K = \{s, q, h\}, \quad \Sigma = \{\sqcup, \triangleright\}$$

και η δ δίνεται από:

$$\delta(s, \triangleright) = (q, \rightarrow)$$
$$\delta(q, \sqcup) = (s, \leftarrow)$$

Παράδειγμα

- Η πλακόστρωση που αντιστοιχεί σε ένα υπολογισμό της M με είσοδο e που δεν τερματίζει είναι:

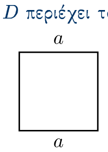


Το Πρόβλημα Πλακόστρωσης

- Τυπικά έστω $M = (K, \Sigma, \delta, s)$. Η αναγωγή κατασκευάζει ένα σύστημα πλακόστρωσης $\mathcal{D} = (D, d_0, H, V)$ τέτοιο ώστε
  Τα σύνολα H και V επιτρέπουν σε δύο πλακάκια να εφάπτονται μόνο αν οι εφαπτόμενες πλευρές τους έχουν την ίδια πληροφορία.
  Το σύνολο D περιέχει τα παρακάτω πλακάκια.

Το Πρόβλημα Πλακόστρωσης

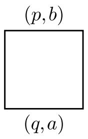
- (a) Για κάθε $a \in \Sigma$, το σύνολο D περιέχει το πλακάκι



Τα πλακάκια αυτά μεταβιβάζουν οποιοδήποτε αμετάβλητο σύμβολο προς τα επάνω από configuration σε configuration.

Το Πρόβλημα Πλακόστρωσης

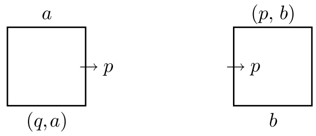
- (b) Για κάθε $a, b \in \Sigma$ και $q, p \in K$ τέτοια ώστε $\delta(q, a) = (p, b)$, το σύνολο D περιέχει το πλακάκι



Αυτό το πλακάκι μεταβιβάζει την θέση της κεφαλής προς τα πάνω και επίσης αλλάζει κατάλληλα την κατάσταση και το σύμβολο που σαρώθηκε.

Το Πρόβλημα Πλακόστρωσης

- (c) Για κάθε $a \in \Sigma$ και $q, p \in K$ τέτοια ώστε $\delta(q, a) = (p, \rightarrow)$, και για κάθε $b \in \Sigma \setminus \{\triangleright\}$, το σύνολο D περιέχει τα πλακάκια



Αυτό τα πλακάκια μεταβιβάζουν την κίνηση της κεφαλής κατά ένα τετράγωνο από τα αριστερά προς τα δεξιά, ενώ μεταβάλλουν κατάλληλα την κατάσταση.

Το Πρόβλημα Πλακόστρωσης

- (d) Για κάθε $a \in \Sigma$ και $q, p \in K$ τέτοια ώστε $\delta(q, a) = (p, \leftarrow)$, και για κάθε $b \in \Sigma$, το σύνολο D περιέχει τα πλακάκια

Αυτό τα πλακάκια μεταβιβάζουν την κίνηση της κεφαλής κατά ένα τετράγωνο από τα δεξιά προς τα αριστερά, ενώ μεταβάλλουν κατάλληλα την κατάσταση.

Το Πρόβλημα Πλακόστρωσης

- (e) Το $\mathcal{D}$ περιέχει ακόμα το πλακάκι της αρχής d_0 και ένα ακόμα πλακάκι που εξασφαλίζει ότι η κάτω σειρά του τεταρτημόριου πλακοστρώνεται σωστά:

Το Πρόβλημα Πλακόστρωσης

- Τυπικά η αναγωγή τ απεικονίζει την αναπαράσταση "M" μιας μηχανής Turing M σε ένα σύστημα πλακόστρωσης $\mathcal{D}$ που ορίζεται όπως παραπάνω και είναι τέτοιο ώστε: Υπάρχει μια πλακόστρωση με το σύστημα $\mathcal{D}$ αν και μόνο αν η M δεν τερματίζει με είσοδο e.
- Τα configurations της M αναπαρίστανται οριζόντια σε μια πλακόστρωση. Διαδοχικά configurations εμφανίζονται το ένα πάνω από το άλλο. Δηλ. η οριζόντια διάσταση παριστάνει την ταινία της M ενώ η κάθετη παριστάνει το χρόνο.
- Αν η M δεν τερματίζει με είσοδο e, τότε διαδοχικές σειρές μπορούν να πλακοστρώνονται επ' άπειρο. Όμως αν η M σταματήσει μετά από k βήματα δεν είναι δυνατό να πλακοστρωθούν πάνω από k σειρές.

Ιδιότητες των Αναδρομικών Γλωσσών

Έχουμε ήδη αποδείξει τις ακόλουθες ιδιότητες:

- Αν μια γλώσσα είναι αναδρομική τότε είναι αναδρομικά αριθμήσιμη. Η κλάση των αναδρομικά αριθμήσιμων γλωσσών είναι ένα γνήσιο υποσύνολο της κλάσης των αναδρομικών γλωσσών.
- Αν L είναι μια αναδρομική γλώσσα τότε το συμπλήρωμα της $\overline{L}$ είναι επίσης μια αναδρομική γλώσσα.
- Αν μια γλώσσα δεν είναι αναδρομική το ίδιο ισχύει και για το συμπλήρωμα της.
- Η κλάση των αναδρομικά αριθμήσιμων γλωσσών δεν είναι κλειστή ως προς την πράξη της συμπλήρωσης.

Ιδιότητες των Αναδρομικών Γλωσσών

- **Θεώρημα.** Μια γλώσσα L είναι αναδρομική αν και μόνο αν η L και η $\overline{L}$ είναι αναδρομικά αριθμήσιμες.

Απόδειξη.

- **(Μόνο αν)** Προφανές.
- **(Αν)** Υποθέτουμε ότι υπάρχει μια μηχανή Turing M_1 που ημιαποφασίζει την L και μια μηχανή Turing M_2 που ημιαποφασίζει την $\overline{L}$. Τότε μπορούμε να κατασκευάσουμε μια μηχανή Turing M με δύο ταινίες που **αποφασίζει** την L.

Απόδειξη

- Η μηχανή M λειτουργεί ως εξής:
  - Γράψε την συμβολοσειρά εισόδου w και στις δύο ταινίες.
  - Προσομοίωσε την λειτουργία των M_1 και M_2 **παράλληλα.** Δηλ. κάνε συνεχώς ένα βήμα της M_1 στην πρώτη ταινία και ένα βήμα της M_2 στην δεύτερη ταινία.
  - Αν τερματίσει η M_1 τότε μεταπήδησε σε κατάσταση αποδοχής.
  - Αν τερματίσει η M_2 τότε μεταπήδησε σε κατάσταση απόρριψης.

Turing-αριθμήσιμες Γλώσσες

- **Ορισμός.** Λέμε ότι μια μηχανή Turing **απαριθμεί** (enumerates) την γλώσσα L αν και μόνο αν για κάποια διακεκριμένη κατάσταση q της M,

$$L = \{w : (s, \triangleright\sqcup) \vdash_M^* (q, \triangleright\sqcup w)\}$$

Μια γλώσσα είναι **Turing-αριθμήσιμη** (Turing-enumerable) αν και μόνο αν υπάρχει κάποια μηχανή Turing που την απαριθμεί.

Turing-αριθμήσιμες Γλώσσες

- **Θεώρημα.** Μια γλώσσα είναι αναδρομικά αριθμήσιμη αν και μόνο αν είναι Turing-αριθμήσιμη.

Απόδειξη. (Μόνο Αν)

- Έστω μια αναδρομικά αριθμήσιμη γλώσσα L που ημιαποφασίζεται από μια μηχανή Turing M.
- Μπορούμε να σχεδιάσουμε μια μηχανή M' η οποία απαριθμεί την L. Η M' ξεκινάει με κενή ταινία και παράγει με συστηματικό τρόπο (π.χ., σε λεξικογραφική διάταξη) όλες τις συμβολοσειρές του αλφάβητου της L, και εκτελεί σε κάθε μια από αυτές τον ίδιο υπολογισμό που θα εκτελούσε η M.

Απόδειξη

- Η M' αποφεύγει να πέσει σε loop ακόμα και στην περίπτωση που η M πέφτει σε loop για κάποια συμβολοσειρά εισόδου, χρησιμοποιώντας μια διαδικασία **dovetailing.** Δηλ. αντί να προσπαθεί να ολοκληρώσει τον υπολογισμό σε κάθε συμβολοσειρά καθώς αυτή παράγεται, η M' λειτουργεί ως εξής:
- **Φάση 1.** Εκτέλεσε ένα βήμα του υπολογισμού της M στην πρώτη συμβολοσειρά του αλφάβητου της M.
- **Φάση 2.** Εκτέλεσε δύο βήματα του υπολογισμού της M σε κάθε μια από τις δύο πρώτες συμβολοσειρές.
- **Φάση 3.** Εκτέλεσε τρία βήματα του υπολογισμού της M σε κάθε μια από τις τρεις πρώτες συμβολοσειρές.

κ.ο.κ.

Απόδειξη

- Κάθε φορά που η M' ανακαλύπτει ότι η M θα αποδεχόταν μια συμβολοσειρά w_i, τότε κάνει τα εξής:
  - Γράψε την w_i στην ταινία και πέρασε στην κατάσταση q ώστε να επισημάνεις αυτό το γεγονός (δηλ. **παρουσίασε** την w_i).
  - Σώσε την w_i και επανέλαβε την απαρίθμηση απο τη Φάση 1. Πρόσεξε όμως ώστε να μην παραλειφθεί η συμβολοσειρά που είναι λεξικογραφικά **μετά** από την w_i.

Απόδειξη

(Αν)

- Έστω Turing-αριθμήσιμη γλώσσα L και μηχανή Turing M που απαριθμεί την L. Τότε μπορούμε να τροποποιήσουμε την M για να ημιαποφασίζει την L ως εξής. Ξανασχεδιάζουμε την M, έτσι ώστε, πριν αρχίσει την διαδικασία απαρίθμησης, να αποθηκεύει την συμβολοσειρά εισόδου.
- Επιπλέον, κάθε φορά που η M μπαίνει στη διακεκριμένη κατάσταση q, η τροποποιημένη μηχανή συγκρίνει την συμβολοσειρά που παρουσιάζεται στην ταινία με την αποθηκευμένη συμβολοσειρά εισόδου. Αν οι συμβολοσειρές είναι ίδιες, τότε η συμβολοσειρά εισόδου γίνεται αποδεκτή, αλλιώς συνεχίζεται η απαρίθμηση.
- Άρα η νέα μηχανή ημιαποφασίζει ακριβώς την γλώσσα που απαριθμεί η M.

Λεξικογραφικά Turing-αριθμήσιμες Γλώσσες

- **Ορισμός.** Έστω M μια μηχανή Turing που απαριθμεί τη γλώσσα L. Λέμε ότι η M **απαριθμεί λεξικογραφικά** την L αν ισχύει η ακόλουθη πρόταση: Αν $(q, \triangleright\sqcup w) \vdash_M^+ (q, \triangleright\sqcup w')$ (όπου q είναι η διακεκριμένη κατάσταση) τότε η w' είναι **λεξικογραφικά μετά** την w.
- Μια γλώσσα είναι **λεξικογραφικά Turing-αριθμήσιμη** (lexicographically Turing-enumerable) αν και μόνο αν υπάρχει κάποια μηχανή Turing που την απαριθμεί λεξικογραφικά.

Λεξικογραφικά Turing-αριθμήσιμες Γλώσσες

- **Θεώρημα.** Μια γλώσσα είναι αναδρομική αν και μόνο αν είναι λεξικογραφικά Turing-αριθμήσιμη.

Απόδειξη. (Μόνο αν)

- Έστω L είναι μια αναδρομική γλώσσα και M μια μηχανή Turing που την αποφασίζει. Τότε μπορούμε εύκολα να κατασκευάσουμε μια μηχανή M' που απαριθμεί λεξικογραφικά την L. Η M' λειτουργεί ως εξής:
  - Παράγει όλες τις συμβολοσειρές του αλφάβητου της L τη μια μετά την άλλη σε λεξικογραφική σειρά.
  - Προσομοιώνει την M σε κάθε τέτοια συμβολοσειρά.
  - Αν η M δέχεται την συμβολοσειρά, τότε η M' "παρουσιάζει" την συμβολοσειρά και συνεχίζει στην επόμενη. Αν η M απορρίπτει την συμβολοσειρά, τότε η M' συνεχίζει στην επόμενη.

Λεξικογραφικά Turing-αριθμήσιμες Γλώσσες

(Αν)

- Υποθέτουμε ότι η L είναι μια λεξικογραφικά Turing-αριθμήσιμη γλώσσα και M η μηχανή Turing που την απαριθμεί. Αν η L είναι πεπερασμένη, τότε είναι επίσης αναδρομική.
- Αν η L είναι άπειρη, τότε η μηχανή M' που αποφασίζει την L λειτουργεί ως εξής:
  - Με είσοδο w, ξεκίνησε την μηχανή απαρίθμησης M.
  - Περίμενε μέχρι είτε να "παρουσιαστεί" η w, είτε μια συμβολοσειρά που είναι λεξικογραφικά μετά την w. Στην πρώτη περίπτωση, μεταπήδησε σε κατάσταση αποδοχής. Στη δεύτερη περίπτωση, μεταπήδησε σε κατάσταση απόρριψης.

Το Θεώρημα του Rice

- **Θεώρημα.** Υποθέτουμε ότι το $\mathcal{C}$ είναι ένα μη κενό, γνήσιο υποσύνολο της κλάσης όλων των αναδρομικά αριθμήσιμων γλωσσών. Τότε το ακόλουθο πρόβλημα είναι μη επιλύσιμο: Αν δοθεί μια μηχανή Turing M, είναι η γλώσσα που ημιαποφασίζεται από την M ένα στοιχείο της κλάσης $\mathcal{C}$;
- Το παραπάνω θεώρημα είναι πολύ ισχυρό και μας δίνει την μη επιλυσιμότητα πολλών προβλημάτων σχετικών με μηχανές Turing!

65

Απόδειξη

- Υποθέτουμε ότι $\emptyset \notin \mathcal{C}$ (αλλιώς θεωρούμε το σύνολο $\overline{\mathcal{C}}$).
- Έστω $L \in \mathcal{C}$ μια γλώσσα που ημιαποφασίζεται από την μηχανή Turing M_L.
- Θα περιγράψουμε μια αναγωγή τ από την γλώσσα

$$H = \{\text{“}M\text{”}\,\text{“}w\text{”} : \quad \text{η μηχανή Turing } M \text{ τερματίζει με είσοδο } w\ \}$$

στη γλώσσα του προβλήματος μας:

$$R = \{\text{“}M\text{”} : \quad \text{η γλώσσα που ημιαποφασίζεται από την } M \text{ είναι στοιχείο της κλάσης } \mathcal{C}\ \}$$

66

Απόδειξη

- Η συνάρτηση τ αντιστοιχεί την συμβολοσειρά “M”“w” (αναπαράσταση μιας μηχανής Turing M και μιας συμβολοσειρά w) στην συμβολοσειρά “$T_{M,w}$” (αναπαράσταση μιας νέας μηχανής $T_{M,w}$).
- Η μηχανή $T_{M,w}$ λειτουργεί ως εξής:

$$T_{M,w}(x): \quad \text{if } U(\text{“}M\text{”}\,\text{“}w\text{”}) \neq \nearrow \text{ then } M_L(x) \text{ else } \nearrow$$

Η γλώσσα που ημιαποφασίζεται από την $T_{M,w}$ είναι στην κλάση $\mathcal{C}$ αν και μόνο αν η M τερματίζει με είσοδο w.

67

Απόδειξη

- Υποθέτουμε ότι η M τερματίζει με είσοδο w. Τότε η $T_{M,w}$ με είσοδο x θα προσομοιώσει την M_L και θα δεχτεί την συμβολοσειρά x αν και μόνο αν $x \in L$. Αρα σ' αυτή την περίπτωση η γλώσσα που ημιαποφασίζεται απο την $T_{M,w}$, δηλαδή η L, ανήκει στην κλάση $\mathcal{C}$.
- Υποθέτουμε τώρα ότι η M δεν τερματίζει με είσοδο w. Σ' αυτή την περίπτωση η $T_{M,w}$ δεν τερματίζει, και άρα ημιαποφασίζει την γλώσσα $\emptyset$ που δεν ανήκει στην κλάση $\mathcal{C}$.
- Δηλαδή έχουμε $x \in H$ αν και μόνο αν $\tau(x) \in R$. Επίσης η τ είναι αναδρομική.
- Τελικά η R δεν είναι αναδρομική, άρα το δοσμένο πρόβλημα δεν είναι επιλύσιμο.

68

Μελέτη

- Κεφάλαια 5.1-5.4 και 5.6-5.7 από το βιβλίο:
Harry R. Lewis και Χρίστος Χ. Παπαδημητρίου, Elements of the Theory of Computation, 2nd edition, Prentice Hall, 1998.

69