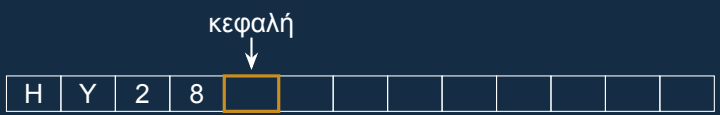
ΗΥ280 / ΘΕΩΡΙΑ ΥΠΟΛΟΓΙΣΜΟΥ

Μηχανές Turing

Μέρος Ι

Χαράλαμπος Ε. Τσουρακάκης

Πανεπιστήμιο Κρήτης — Διάλεξη 18



Ένας επιστήμονας, πέντε επαναστάσεις

1. Υπολογισιμότητα και όρια του υπολογισμού.
2. Κρυπτογραφία και κρυπτανάλυση.
3. Ηλεκτρονικοί υπολογιστές και λογισμικό.
4. Τεχνητή νοημοσύνη.
5. Βιολογία ως πληροφορία και υπολογισμός.



Μια σύντομη ζωή και μια τεράστια κληρονομιά για την ανθρωπότητα

- 1912: γεννιέται στο Λονδίνο.
- 1936: μαθηματικό μοντέλο του υπολογισμού.
- 1939–1945: κρυπτανάλυση στο Bletchley Park.
- 1945–1948: σχεδιασμός του ACE και έρευνα για νοήμονες μηχανές.
- 1950: το παιχνίδι της μίμησης.
- 1952: μορφογένεση
- 1952: καταδίκη για ομοφυλοφιλική σχέση και ορμονική αγωγή.
- 1954: πεθαίνει σε ηλικία 41 ετών.
- 2013: απονέμεται μεταθανάτια βασιλική χάρη.

1952: Η ποινικοποίηση της προσωπικής ζωής

- Στις 31 Μαρτίου 1952 ο Turing καταδικάστηκε βάσει του άρθρου 11 του Criminal Law Amendment Act του 1885, για συναινετική σχέση με άλλον άνδρα.
- Αντί της φυλάκισης, αποδέχθηκε αναστολή υπό τον όρο υποβολής σε ορμονική αγωγή, με σκοπό την καταστολή της σεξουαλικής επιθυμίας.
- Η αγωγή, γνωστή ως χημικός ευνουχισμός, συνιστούσε σοβαρή επέμβαση στη σωματική του αυτονομία.
- Του αφαιρέθηκε η διαβάθμιση ασφαλείας και αποκλείστηκε από τη μεταπολεμική συνεργασία με το GCHQ.

Πηγή: House of Lords, 19 Ιουλίου 2013· UK Ministry of Justice, 2013.

1954: Ο θάνατος του Alan Turing

- Στις 7 Ιουνίου 1954 πεθαίνει στο σπίτι του στο Wilmslow. Βρέθηκε νεκρός την επομένη, σε ηλικία μόλις 41 ετών.
- Η επίσημη ιατροδικαστική έρευνα κατέληξε σε αυτοκτονία. Οι ακριβείς περιστάσεις παραμένουν αντικείμενο ιστορικής συζήτησης.

Μια αδιαμφισβήτητη αδικία

Η δίωξη και η εξαναγκαστική θεραπεία του αποτελούν τεκμηριωμένα γεγονότα, ανεξάρτητα από τις αβεβαιότητες γύρω από τον θάνατό του.

Πηγή: Science Museum, The Spirit of Alan Turing· University of Manchester.

Η αναγνώριση ήρθε πολύ αργά

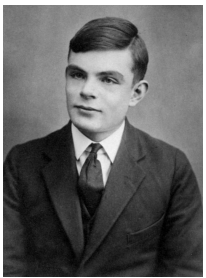
- 2009: ο πρωθυπουργός Gordon Brown ζητά επισήμως συγγνώμη για τη μεταχείριση του Turing.
- 2013: η βασίλισσα Ελισάβετ Β΄ του απονέμει μεταθανάτια χάρη.
- 2017: η νομοθεσία που έγινε γνωστή ως Alan Turing Law προβλέπει, μεταξύ άλλων, μεταθανάτιες χάρες για καταδίκες βάσει παλαιότερων διατάξεων.
- 2021: το πορτρέτο του εμφανίζεται στο βρετανικό χαρτονόμισμα των 50 λιρών.

Η δικαιώση μετά θάνατο δυστυχώς δεν μπορεί να αναιρέσει την αδικία που υπέστη εν ζωή.

Πηγή: University of Manchester, 2009· UK Ministry of Justice, 2013· Bank of England.

Η επιστημονική ιδιοφυΐα και η ανθρώπινη αξιοπρέπεια

- Ο άνθρωπος που συνέβαλε αποφασιστικά στην πολεμική προσπάθεια της Βρετανίας διώχθηκε από το ίδιο κράτος για την προσωπική του ζωή.
- Η επιστημονική και τεχνολογική πρόοδος δεν συνεπάγεται αυτομάτως ηθική ή κοινωνική πρόοδο.



Στη μνήμη του Alan Mathison Turing

1912–1954

1. Υπολογισιμότητα: το ερώτημα του Hilbert

Entscheidungsproblem (1928)

Υπάρχει αλγόριθμος που αποφασίζει αν οποιαδήποτε πρόταση της πρωτοβάθμιας λογικής είναι λογικά έγκυρη;

Η θεμελιώδης συνεισφορά του Turing (1936):

1. **Τυποποίηση της έννοιας του αλγορίθμου.** Εισάγει τη μηχανή Turing: ένα αφηρημένο μαθηματικό μοντέλο υπολογισμού με πεπερασμένους κανόνες και απεριόριστη ταινία.
2. **Καθολική μηχανή Turing.** Κατασκευάζει μία μηχανή που μπορεί να προσομοιώσει οποιαδήποτε άλλη μηχανή Turing. Θεμελιώνει την έννοια του υπολογιστή γενικού σκοπού.
3. **Μη αποφασισιμότητα.** Αποδεικνύει ότι υπάρχουν προβλήματα για τα οποία δεν μπορεί να υπάρξει αλγόριθμος απόφασης.
4. **Αρνητική απάντηση στο ερώτημα του Hilbert.** Ανάγει ένα μη αποφασίσιμο υπολογιστικό πρόβλημα στο πρόβλημα της εγκυρότητας προτάσεων της πρωτοβάθμιας λογικής.

Η μεγάλη τομή

Στην προσπάθειά του να αποδείξει τα όρια του υπολογισμού, ο Turing έθεσε τα θεμέλια της σύγχρονης Πληροφορικής.

Πηγή: Turing (1936/37), Church (1936, ανεξάρτητο αρνητικό αποτέλεσμα).

Άτυπη περιγραφή Μηχανής Turing

Πεπερασμένο πλήθος εσωτερικών καταστάσεων και «μνήμη» με τη μορφή μιας ταινίας (tape):

- Η ταινία: «ημιάπειρη ακολουθία» κελιών, καθένα με ένα σύμβολο πεπερασμένου αλφαβήτου.
- Πρόσβαση μόνο μέσω της κεφαλής (head), που βρίσκεται πάνω από ένα κελί.
- Αρχικά: η είσοδος και μετά άπειρα κενά (blanks)· η κεφαλή στο πρώτο κελί.

Κάθε βήμα: *διάβασε* το σύμβολο· ίσως *γράψε* νέο, *άλλαξε κατάσταση*, *κίνησε* την κεφαλή σε γειτονικό κελί, *τερμάτισε* (με αποδοχή ή απόρριψη). Όλα εξαρτώνται μόνο από κατάσταση και σύμβολο.
How Turing Machines Work [Youtube]

2. Κρυπτογραφία: Bletchley Park

- Η Enigma μετασχημάτιζε γράμματα μέσω περιστρεφόμενων δίσκων και ηλεκτρικών συνδέσεων.
- Οι Πολωνοί μαθηματικοί Rejewski, Różycki και Zygalski είχαν επιτύχει σημαντική πρόοδο πριν τον πόλεμο.
- Ο Turing συνέβαλε στη βρετανική Bombe
- Η επιτυχία ήταν συλλογική: μαθηματικοί, μηχανικοί, χειριστές, αναλυτές, . . .

Πηγή: Bletchley Park, ιστορικό της Bombe, Hodges (1983).

Ταινία ”Imitation Game” και η μηχανή Enigma



The Insane Real Engineering of the Nazi Enigma Machine

Βίντεο για το πώς σχεδιάσαν οι Γερμανοί (Arthur Scherbius)το Enigma.

Από το Enigma στη σύγχρονη ασφάλεια



- Συμμετρική κρυπτογραφία: κοινό μυστικό κλειδί.
- Δημόσιο κλειδί: διαφορετικά κλειδιά με μαθηματική σχέση.
- Η σύγχρονη ασφάλεια βασίζεται συχνά σε υποθέσεις υπολογιστικής δυσκολίας

Πηγή: Diffie και Hellman (1976), Rivest–Shamir–Adleman (1977), RFC 8446 (2018).

3. Από την αφηρημένη μηχανή στο υλικό

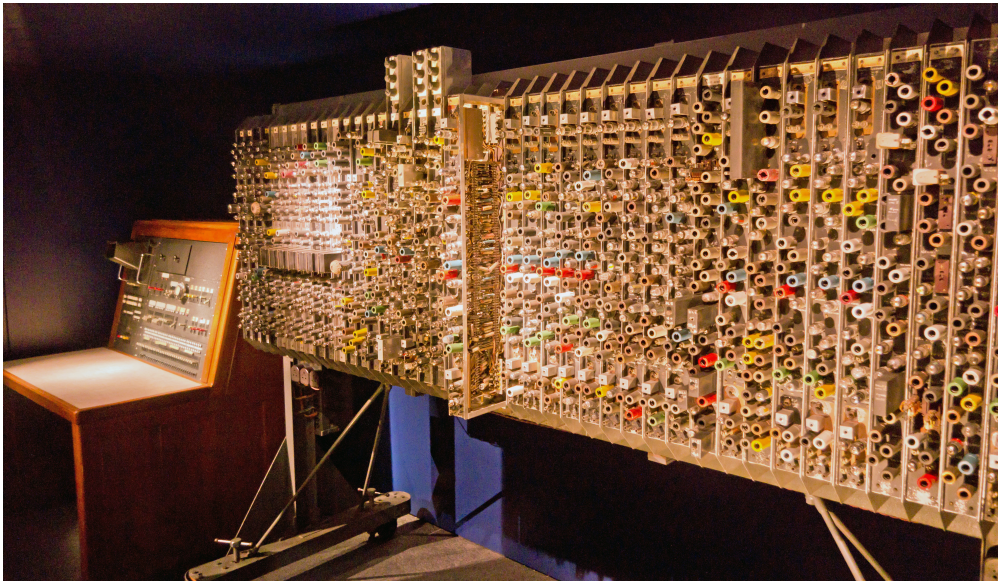
- Η Bombe ήταν ειδικού σκοπού, όχι καθολικός υπολογιστής.
- Το ACE (Automatic Computing Engine) ήταν σχέδιο ηλεκτρονικού υπολογιστή αποθηκευμένου προγράμματος.
- Το Pilot ACE εκτέλεσε το πρώτο του πρόγραμμα το 1950.
- Στο Manchester ο Turing εργάστηκε σε λογισμικό και σε ιδέες για μηχανική μάθηση και σκάκι.

Η μεγάλη αλλαγή

Δεν κατασκευάζουμε νέα μηχανή για κάθε εργασία: αλλάζουμε το πρόγραμμα.

Πηγή: Turing, Proposed Electronic Calculator (1946)· Hodges (1983).

Pilot Ace



4. Τεχνητή νοημοσύνη πριν από την εποχή της AI

- Το 1948 ο Turing εξετάζει δίκτυα απλών μονάδων και την εκπαίδευσή τους (Intelligent Machinery).
  - εξετάζει «μη οργανωμένες μηχανές»: τυχαία δίκτυα απλών λογικών μονάδων (τύπου NAND), των οποίων οι συνδέσεις μπορούν να τροποποιηθούν ώστε το δίκτυο να «εκπαιδευτεί».
  - Το αφεντικό του το θεώρησε ασήμαντο, και δε δημοσιεύτηκε μέχρι το 1968
  - Μάλιστα πρότεινε και ” P-type machines”, που θα εκπαιδεύονταν με σήματα ευχαρίστησης και πόνου, πρόδρομος του reinforcement learning (βάση των σύγχρονων agents),
- Το 1950 προτείνει να αντικαταστήσουμε το ερώτημα «μπορούν οι μηχανές να σκέφτονται;» με ένα λειτουργικό κριτήριο, το παιχνίδι μίμησης, που εξετάζει μόνο την παρατηρήσιμη συμπεριφορά αντί για έναν ορισμό της σκέψης.
- Προτείνει μια «παιδική μηχανή» που εκπαιδεύεται αντί να προγραμματίζεται εξ ολοκλήρου.
- Η ιδέα είναι πρόδρομος της μηχανικής μάθησης.

Πηγή: Turing (1948, 1950)

5. Βιολογία: πώς δημιουργούνται τα μοτίβα;

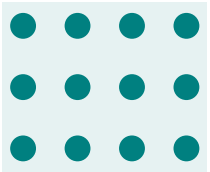
Το 1952 ο Turing δείχνει πώς η αλληλεπίδραση και διάχυση χημικών ουσιών μπορεί να αποσταθεροποιήσει μια αρχικά ομογενή κατάσταση.

$$\partial_t u = D_u \nabla^2 u + f(u, v)$$

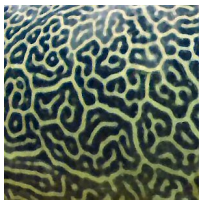
$$\partial_t v = D_v \nabla^2 v + g(u, v)$$

Βασική ιδέα

Από τοπικούς κανόνες αναδύεται μακροσκοπική δομή.

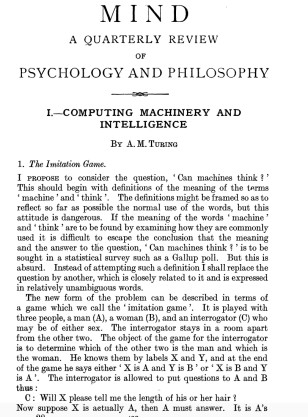


Σχηματική απεικόνιση κηλίδων, όχι αριθμητική προσομοίωση.



Πηγή: Turing, The Chemical Basis of Morphogenesis (1952).

1950: Μπορούν οι μηχανές να σκέφτονται;



MIND

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. *The Imitation Game.*

I propose to consider the question, 'Can machines think?' This should begin with definitions of the meaning of the terms 'machine' and 'think'. The definitions might be framed so as to reflect so far as possible the normal use of the words, but this attitude is dangerous. If the meaning of the words 'machine' and 'think' are to be found by examining how they are commonly used it is difficult to escape the conclusion that the meaning and the answer to the question, 'Can machines think?' is to be sought in a statistical survey such as a Gallup poll. But this is absurd. Instead of attempting such a definition I shall replace the question by another, which is closely related to it and is expressed in relatively unambiguous words.

The new form of the problem can be described in terms of a game which we call the 'imitation game'. It is played with three people, a man (A), a woman (B), and an interrogator (C) who may be of either sex. The interrogator stays in a room apart from the other two. The object of the game for the interrogator is to determine which of the other two is the man and which is the woman. He knows them by labels X and Y, and at the end of the game he says either 'X is A and Y is B' or 'X is B and Y is A'. The interrogator is allowed to put questions to A and B thus:

C: Will X please tell me the length of his or her hair?

Now suppose X is actually A, then A must answer. It is A's

Πηγή: Turing, Mind 59(236), 433–460. Εικόνα από το υλικό της εκφώνησης.

Το παιχνίδι της μίμησης



Ο κριτής δεν βλέπει ούτε ακούει τους συνομιλητές. Ποιος είναι ο άνθρωπος;

Η αρχική εκδοχή του άρθρου ξεκινά με παιχνίδι διάκρισης φύλου· εδώ παρουσιάζεται η καθιερωμένη τριμερής παραλλαγή.

Πηγή: Turing (1950), §1, §5.

Η πρόβλεψη για το έτος 2000

Ο Turing εκτίμησε ότι σε περίπου 50 χρόνια ένας μέσος κριτής, μετά από πέντε λεπτά ερωτήσεων, δεν θα είχε πιθανότητα μεγαλύτερη από 70% να κάνει σωστή αναγνώριση.

Δύο διαφορετικοί ισχυρισμοί

Μπορεί μια μηχανή να περάσει το τεστ; Και αν το περάσει, είναι το τεστ επαρκές ως κριτήριο νοημοσύνης;

Η πρώτη ερώτηση είναι εμπειρική. Η δεύτερη παραμένει εν μέρει εννοιολογική.

Πηγή: Turing (1950)

1. Η θεολογική αντίρρηση

Η αντίρρηση

Η σκέψη αποδίδεται στην αθάνατη ψυχή, την οποία υποτίθεται ότι μόνο οι άνθρωποι διαθέτουν.

Η απάντηση του Turing

- Απορρίπτει την προϋπόθεση, αλλά απαντά και μέσα στο θεολογικό πλαίσιο.
- Ο αποκλεισμός περιορίζει αυθαίρετα τη θεϊκή παντοδυναμία: ο Θεός θα μπορούσε να δώσει ψυχή και σε έναν ελέφαντα.
- Κατασκευάζοντας τέτοιες μηχανές δεν σφετεριζόμαστε τη δύναμή Του, όπως δεν τη σφετεριζόμαστε όταν κάνουμε παιδιά.

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(1).

2. Στρουθοκαμηλισμός («Heads in the Sand»)

Η αντίρρηση

Οι συνέπειες θα ήταν τόσο τρομακτικές που προτιμούμε να ελπίζουμε και να πιστεύουμε ότι οι μηχανές δεν μπορούν να σκέφτονται.

Η απάντηση του Turing

- Σπάνια διατυπώνεται ανοιχτά, αλλά είναι διάχυτη: θέλουμε να πιστεύουμε ότι ο άνθρωπος είναι *αναγκαστικά* ανώτερος.
- Είναι ισχυρότερη στους διανοούμενους, που εκτιμούν περισσότερο τη δύναμη της σκέψης.
- Δεν χρειάζεται ανασκευή αλλά παρηγοριά.

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(2).

3. Η μαθηματική αντίρρηση

Η αντίρρηση (Gödel 1931, Turing 1936)

Για κάθε διακριτή μηχανή υπάρχουν ερωτήματα στα οποία είτε απαντά λάθος είτε δεν απαντά ποτέ. Ο ανθρώπινος νους δεν έχει τέτοια όρια, άρα υπερέχει.

Η απάντηση του Turing

- Τα όρια των μηχανών αποδεικνύονται· η απουσία ορίων στον άνθρωπο απλώς δηλώνεται.
- Και οι άνθρωποι απαντούν συχνά λάθος.
- Όποιος «νικά» μια μηχανή, νικά μόνο *αυτή*: μπορεί να υπάρχει άλλη μηχανή εξυπνότερη από αυτόν.

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(3).

4. Το επιχείρημα της συνείδησης

Η αντίρρηση (Jefferson, 1949)

Δεν αρκεί η μηχανή να γράψει ένα σονέτο: πρέπει να το γράψει εξαιτίας σκέψεων και συναισθημάτων που βιώνει, και να *γνωρίζει* ότι το έγραψε.

Η απάντηση του Turing

- Στην ακραία μορφή του, το επιχείρημα οδηγεί στον σολιψισμό: μόνο για τη δική μας σκέψη θα ήμασταν βέβαιοι.
- Για τους άλλους ανθρώπους δεχόμαστε μια «ευγενική σύμβαση» ότι σκέφτονται. Γιατί όχι και για τη μηχανή;
- Δεν αρνείται το μυστήριο της συνείδησης, αλλά δεν χρειάζεται να λυθεί πριν απαντήσουμε στο ερώτημα.

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(4).

ChatGPT για μαντινάδες αντί σονέτου

Prompt στο ChatGPT για χιουμοριστικές μαντινάδες προσωπικού επαναπατρισμού μου βάζοντας τον ΟΠΕΚΕΠΕ.

Ο Τσουρακάκης κάθεται στου Ψηλορείτη τα ύψη,
κι αντί να βόσκει πρόβατα, θεωρήματα θα δείξει.

Στον ΟΠΕΚΕΠΕ ήθελε να στείλει τα χαρτιά του,
μα σε μορφή LaTeX ήτανε τα δικαιολογητικά του.

Στην Κρήτη κάνει μάθημα και γράφει με ουσία,
μα ο βοσκός με ΟΠΕΚΕΠΕ βγάζει περιουσία.

Σαν του ζητήσαν μέτρημα πόσα οζά κατέχει,
είπε «το πρόβλημα είν' NP, και λύση εύκολη δεν έχει».

5. Επιχειρήματα περί επιμέρους ανικανοτήτων

Η αντίρρηση

«Μια μηχανή δεν θα μπορέσει ποτέ να…» είναι ευγενική, έχει χιούμορ, ερωτευτεί, απολαύσει φράουλες με κρέμα, κάνει λάθη, μάθει από την εμπειρία, κάνει κάτι πραγματικά νέο.

Η απάντηση του Turing

- Αβάσιμες γενικεύσεις από τις περιορισμένες μηχανές που έχουμε δει.
- Πολλές είναι μεταμφιεσμένες μορφές του επιχειρήματος της συνείδησης.
- Λάθη: διακρίνει σφάλματα λειτουργίας από σφάλματα συμπεράσματος. Στο παιχνίδι μίμησης η μηχανή μπορεί και σκόπιμα να σφάλλει.
- Μάθηση από την εμπειρία: βλ. «μηχανές που μαθαίνουν» (§7).

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(5).

6. Η αντίρρηση της Lady Lovelace

Η αντίρρηση (Lovelace, 1843)

«The Analytical Engine has no pretensions whatever to originate anything.» Η μηχανή εκτελεί μόνο όσα ξέρουμε πώς να της υποδείξουμε.

Η απάντηση του Turing

- Οι μηχανές τον εκπλήσσουν συχνά, γιατί δεν υπολογίζει αρκετά για να τις προβλέψει.
- Πλάνη: ότι όλες οι συνέπειες γνωστών δεδομένων είναι αμέσως γνωστές.
- Είναι η δική μας «πρωτότυπη» δουλειά κάτι παραπάνω από καρπός της εκπαίδευσής μας;
- Η ουσιαστική απάντηση: μηχανές που μαθαίνουν (§7).

Πηγή: Lovelace (1843), σημ. G· Turing (1950), Computing Machinery and Intelligence, §6(6).

7. Η συνέχεια του νευρικού συστήματος

Η αντίρρηση

Το νευρικό σύστημα δεν είναι μηχανή διακριτών καταστάσεων: ένα μικρό σφάλμα στην ώση που φτάνει σε έναν νευρώνα μπορεί να αλλάξει σημαντικά την ώση που εξέρχεται.

Η απάντηση του Turing

- Η διαφορά υπάρχει, αλλά ο ανακριτής βλέπει μόνο απαντήσεις, όχι τον μηχανισμό.
- Παράδειγμα: ένας διαφορικός αναλυτής (αναλογικός) δίνει για το π τιμές γύρω από το 3,14 με σφάλμα.
- Ένας ψηφιακός υπολογιστής μιμείται και την ανακρίβεια, απαντώντας τυχαία 3,12–3,16 με κατάλληλες πιθανότητες (π.χ. 3,14 με πιθανότητα 0,55).

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(7).

8. Η μη τυποποιησιμότητα της συμπεριφοράς

Η αντίρρηση

Αν ο άνθρωπος ακολουθούσε πλήρες σύνολο κανόνων, θα ήταν μηχανή. Κανένα σύνολο κανόνων δεν καλύπτει κάθε περίσταση (κόκκινο *και* πράσινο μαζί;). Άρα ο άνθρωπος δεν είναι μηχανή.

Η απάντηση του Turing

- *Κανόνες* συμπεριφοράς (συνειδητές εντολές) $\neq$ *νόμοι* συμπεριφοράς (φυσικοί νόμοι: «αν τον τσιμπήσεις, θα τσιρίξει»).
- Η απουσία πλήρων κανόνων δεν αποδεικνύει απουσία νόμων.
- Πείραμα: πρόγραμμα στο Manchester αντιστοιχίζει αριθμούς 16 ψηφίων σε άλλους. Κανείς δεν μπορεί να το προβλέψει από τις απαντήσεις του.

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(8).

9. Η εξωαισθητηριακή αντίληψη

Η αντίρρηση

Αν υπάρχει τηλεπάθεια, ο άνθρωπος μπορεί να κάνει κάτι που η μηχανή δεν μπορεί. Παράδειγμα: «Τι χρώμα έχει η κάρτα στο χέρι μου;» Ο τηλεπαθής βρίσκει 130/400, η μηχανή $\approx$104/400, και ο ανακριτής τους ξεχωρίζει.

Η απάντηση του Turing

- Θεωρεί την αντίρρηση ισχυρή: τα στατιστικά στοιχεία της εποχής (Rhine, δεκαετία 1930) του φαίνονταν συντριπτικά.
- Λύση: οι παίκτες τοποθετούνται σε «δωμάτιο αδιαπέραστο από τηλεπάθεια».
- Τα πειράματα αυτά δεν αναπαράχθηκαν· η θέση έχει σήμερα ιστορικό ενδιαφέρον.

Πηγή: Turing (1950), Computing Machinery and Intelligence, §6(9).

Μηχανές που μαθαίνουν: η «μηχανή-παιδί»

Η πρόταση του Turing (§7)

Αντί να προγραμματίσουμε έναν ενήλικο νου, να κατασκευάσουμε έναν νου-παιδί και να τον *εκπαιδεύσουμε*.

- Τρία συστατικά: αρχική κατάσταση, εκπαίδευση, εμπειρία.
- Αναλογία με την εξέλιξη: δομή $\leftrightarrow$ κληρονομικότητα, αλλαγές $\leftrightarrow$ μεταλλάξεις, κρίση του πειραματιστή $\leftrightarrow$ φυσική επιλογή.
- Εκπαίδευση με ανταμοιβή και τιμωρία (συνέχεια των P-type μηχανών του 1948).
- Ο δάσκαλος συχνά δεν ξέρει τι συμβαίνει μέσα στη μηχανή: η ουσιαστική απάντηση στη Lovelace.

Σήμερα: προεκπαίδευση σε δεδομένα και μάθηση μέσω ανταμοιβής (Reinforcement learning from human feedback (RLHF)) στα LLM.

Πηγή: Turing (1948), Intelligent Machinery· Turing (1950), Computing Machinery and Intelligence, §7.

Συστατικά μιας μηχανής Turing

Ορισμός 1

Μια μηχανή Turing αποτελείται από:

- πεπερασμένο σύνολο Γ ($|\Gamma| \geq 2$): το αλφάβητο ταινίας (tape alphabet)·
- σύμβολο $\square \in \Gamma$: το κενό (blank)·
- μη κενό $\Sigma \subseteq \Gamma \setminus \{\square\}$: το αλφάβητο εισόδου (input alphabet)·
- πεπερασμένο σύνολο Q καταστάσεων (states), με τρεις διακεκριμένες ειδικές καταστάσεις $\text{start}, \text{accept}, \text{reject} \in Q$·
- συνάρτηση μετάβασης (transition function)

$$\delta : (Q \setminus \{\text{accept}, \text{reject}\}) \times \Gamma \to Q \times \Gamma \times \{-1, +1\} \text{ ή } \{L, R\}.$$

Παρατήρηση

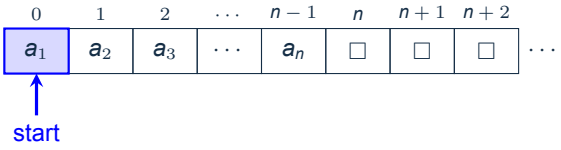
$|\Gamma| \geq 2$ αφού $\Sigma \neq \emptyset$ και $\square \in \Gamma$.

Διαμορφώσεις

Ορισμός 2

Διαμόρφωση (configuration): τριάδα $(q, x, i) \in Q \times \Gamma^* \times \mathbb{N}$ — κατάσταση q, ταινία x ακολουθούμενη από άπειρα κενά, κεφαλή στη θέση i.

- Τα τελικά κενά αγνοούνται: (q, x, i) και $(q, x\square, i)$ είναι η *ίδια* διαμόρφωση.
- Αρχική διαμόρφωση για είσοδο $w \in \Sigma^*$: $(\text{start}, w, 0)$.

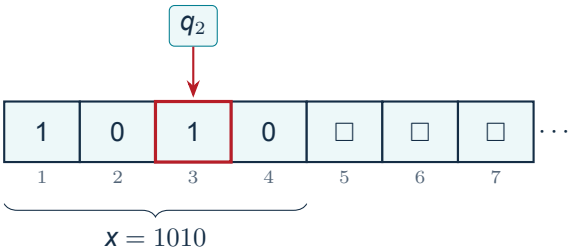


- $\delta(q, a) = (p, b, -1)$: «στην q διαβάζοντας a, πήγαινε στην p, γράψε b, κινήσου *αριστερά*».
- Αν η θέση της κεφαλής γίνει αρνητική, δεν υπάρχει μετάβαση: η μηχανή καταρρέει (crashes). Χρησιμοποιούμε εδώ τη σύμβαση του Erickson· στην παραλλαγή του Sipser η κεφαλή παραμένει στη θέση 0. Τα μοντέλα είναι ισοδύναμα.

Τι είναι μια διαμόρφωση (configuration);

Ορισμός

Μια διαμόρφωση (q, x, i) περιγράφει πλήρως τη μηχανή Turing σε μια δεδομένη χρονική στιγμή.



- $q = q_2$: η τρέχουσα κατάσταση.
- $x = 1010$: το περιεχόμενο της ταινίας.
- $i = 3$: η θέση της κεφαλής.

$$\boxed{(q_2, 1010, 3)}$$

Πεπερασμένη περιγραφή μιας άπειρης ταινίας

Η ταινία είναι άπειρη προς τα δεξιά, αλλά σε κάθε πεπερασμένη χρονική στιγμή μόνο πεπερασμένα κελιά περιέχουν μη κενά σύμβολα.



Οι συμβολοσειρές

$$101, \qquad 101\square, \qquad 101\square\square$$

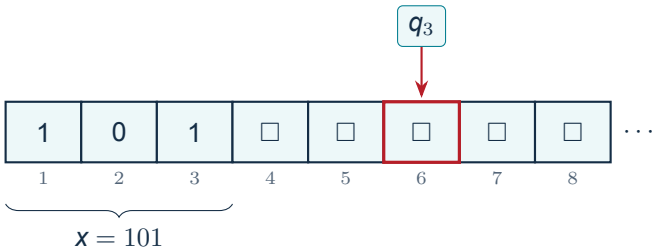
περιγράφουν ακριβώς το ίδιο περιεχόμενο ταινίας.

Σύμβαση

Επιλέγουμε τη συντομότερη δυνατή συμβολοσειρά x. Όλα τα κελιά μετά τη θέση $|x|$ θεωρούνται κενά.

Αν η ταινία είναι εντελώς κενή, θέτουμε $x = \varepsilon$.

Η θέση της κεφαλής είναι ανεξάρτητη από την x



Η παραπάνω διαμόρφωση είναι $\boxed{(q_3, 101, 6)}$

- Η συμβολοσειρά x έχει μήκος 3.
- Η κεφαλή βρίσκεται στη θέση 6.
- Το σύμβολο που διαβάζει η κεφαλή είναι □.

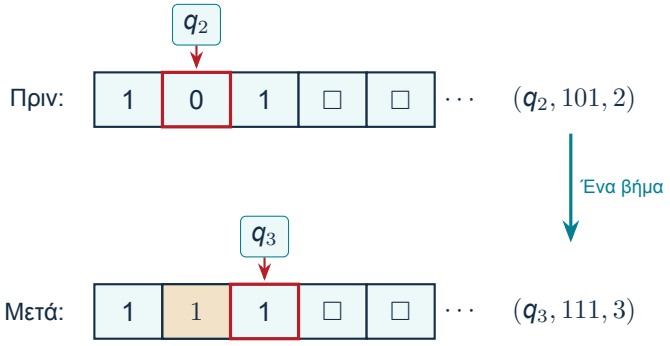
Προσοχή

Δεν απαιτείται $i \leq |x|$. Η κεφαλή μπορεί να βρίσκεται σε οποιοδήποτε κελί της άπειρης ταινίας.

Μετάβαση μεταξύ διαμορφώσεων

Έστω ότι η συνάρτηση μετάβασης περιέχει τον κανόνα

$$\delta(q_2, 0) = (q_3, 1, R).$$

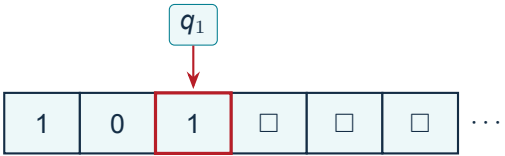


$$(q_2, 101, 2) \rightsquigarrow (q_3, 111, 3)$$

Η μηχανή γράφει 1 αντί για 0, μετακινεί την κεφαλή δεξιά και εισέρχεται στην κατάσταση q_3.

Άσκηση κατανόησης

Έστω η διαμόρφωση $(q_1, 101, 3)$
και ο κανόνας μετάβασης $\delta(q_1, 1) = (q_2, \square, R)$.

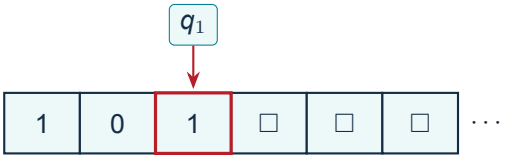


Ποια είναι η νέα διαμόρφωση;

1. $(q_2, 101, 4)$
2. $(q_2, 10, 3)$
3. $(q_2, 10, 4)$
4. $(q_1, 10, 4)$

Άσκηση κατανόησης

Έστω η διαμόρφωση $(q_1, 101, 3)$
και ο κανόνας μετάβασης $\delta(q_1, 1) = (q_2, \square, R)$.



Ποια είναι η νέα διαμόρφωση;

1. $(q_2, 101, 4)$
2. $(q_2, 10, 3)$
3. $(q_2, 10, 4)$
4. $(q_1, 10, 4)$

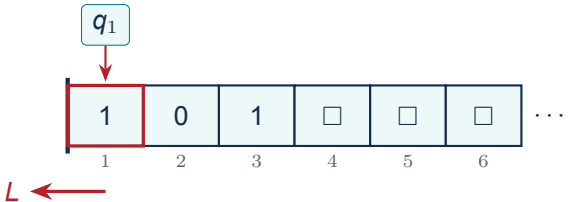
Σωστή απάντηση: 3

$$(q_1, 101, 3) \rightsquigarrow (q_2, 10, 4).$$

Το τελευταίο 1 διαγράφεται και η κεφαλή μετακινείται στη θέση 4.

Άσκηση κατανόησης: το αριστερό άκρο της ταινίας

Έστω η διαμόρφωση $(q_1, 101, 1)$ και ο κανόνας μετάβασης $\delta(q_1, 1) = (q_2, 0, L)$.

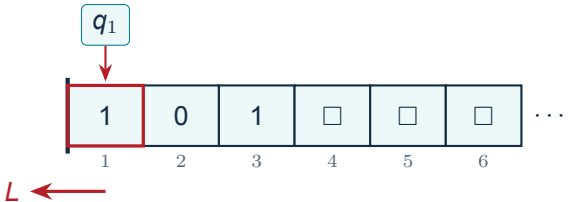


Ποια είναι η νέα διαμόρφωση;

1. $(q_2, 001, 0)$
2. $(q_2, 001, 1)$
3. $(q_1, 101, 1)$

Άσκηση κατανόησης: το αριστερό άκρο της ταινίας

Έστω η διαμόρφωση $(q_1, 101, 1)$ και ο κανόνας μετάβασης $\delta(q_1, 1) = (q_2, 0, L)$.



Ποια είναι η νέα διαμόρφωση;

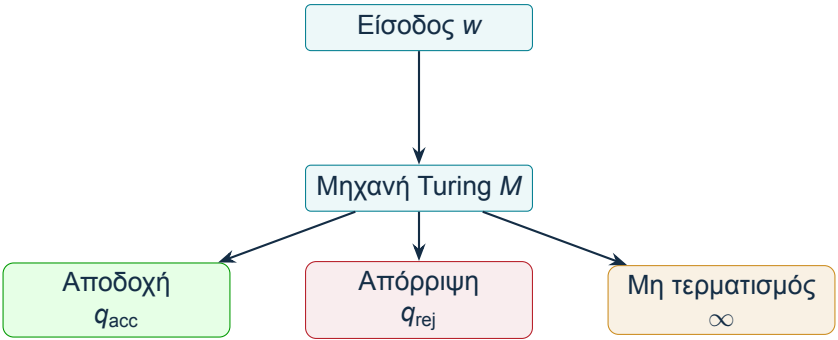
1. $(q_2, 001, 0)$
2. $(q_2, 001, 1)$
3. $(q_1, 101, 1)$

Σωστή απάντηση: 2

$(q_1, 101, 1) \rightsquigarrow (q_2, 001, 1)$. Σύμφωνα με τη σύμβαση του Sipser, όταν η κεφαλή βρίσκεται στο πρώτο κελί και λαμβάνει εντολή L, παραμένει στο πρώτο κελί. Η εγγραφή του συμβόλου και η αλλαγή κατάστασης πραγματοποιούνται κανονικά.

Οι τρεις δυνατές εκβάσεις ενός υπολογισμού

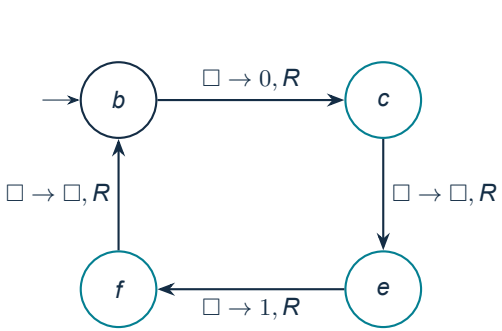
Έστω μια μηχανή Turing M και μια είσοδος $w \in \Sigma^*$.



Προσοχή

Το ότι μια μηχανή Turing δεν αποδέχεται μια είσοδο δεν σημαίνει απαραίτητα ότι την απορρίπτει!

Παράδειγμα μηχανής Turing που δεν τερματίζει





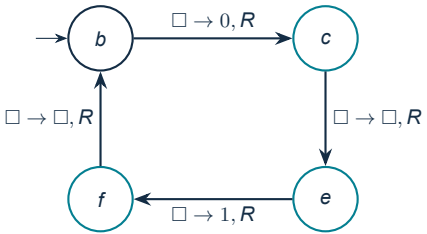
Περιεχόμενο ταινίας κατά την εκτέλεση

$$0\,\square\,1\,\square\,0\,\square\,1\,\square\,\cdots$$

Για αρχικά κενή ταινία, η μηχανή δεν επισκέπτεται ποτέ τις καταστάσεις q_{acc} και q_{rej}.

Πηγή: Προσαρμογή του ιστορικού παραδείγματος του Turing (1936/37) στον ορισμό του Sipser.

Οι μεταβάσεις προς την κατάσταση απόρριψης





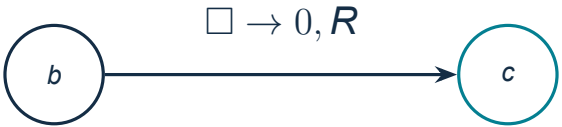
Παραδείγματα μεταβάσεων που δεν σχεδιάζονται

$$\delta(b, 0) = (q_{rej}, 0, R), \quad \delta(b, 1) = (q_{rej}, 1, R),$$

$$\delta(c, 0) = (q_{rej}, 0, R), \quad \delta(e, 1) = (q_{rej}, 1, R).$$

Γενικότερα, για κάθε $q \in \{b, c, e, f\}$ και κάθε $a \in \{0, 1\}$, ορίζουμε $\boxed{\delta(q, a) = (q_{rej}, a, R).}$

Πώς διαβάζουμε ένα διάγραμμα μηχανής Turing;



Το παραπάνω βέλος αντιστοιχεί στον κανόνα

$$\boxed{\delta(b, \square) = (c, 0, R)}.$$

Αν η μηχανή βρίσκεται στην κατάσταση b και η κεφαλή διαβάζει το κενό σύμβολο □:

1. Γράφει το σύμβολο 0 στο τρέχον κελί.
2. Μετακινεί την κεφαλή κατά μία θέση δεξιά.
3. Μεταβαίνει στην κατάσταση c.

Γενική μορφή μιας μετάβασης

Η ετικέτα $a \to a', D$ σημαίνει: διάβασε a, γράψε a' και κινήσου προς την κατεύθυνση $D \in \{L, R\}$.

Η τυπική περιγραφή της μηχανής

$$M = (Q, \Sigma, \Gamma, \delta, b, q_{\text{acc}}, q_{\text{rej}})$$

όπου

$$Q = \{b, c, e, f, q_{\text{acc}}, q_{\text{rej}}\},$$
$$\Sigma = \{0, 1\},$$
$$\Gamma = \{0, 1, \square\}.$$

Η συνάρτηση μετάβασης δ δίνεται από τον πίνακα:

δ	$\square$	0	1
b	$(c, 0, R)$	$(q_{\text{rej}}, 0, R)$	$(q_{\text{rej}}, 1, R)$
c	$(e, \square, R)$	$(q_{\text{rej}}, 0, R)$	$(q_{\text{rej}}, 1, R)$
e	$(f, 1, R)$	$(q_{\text{rej}}, 0, R)$	$(q_{\text{rej}}, 1, R)$
f	$(b, \square, R)$	$(q_{\text{rei}}, 0, R)$	$(q_{\text{rei}}, 1, R)$

Ο ρόλος των τεσσάρων καταστάσεων

Κατάσταση *b*

Γράφει το ψηφίο 0 και μετακινεί την κεφαλή δεξιά.

$$\delta(b, \square) = (c, 0, R)$$

Κατάσταση *e*

Γράφει το ψηφίο 1 και μετακινεί την κεφαλή δεξιά.

$$\delta(e, \square) = (f, 1, R)$$

Κατάσταση *c*

Αφήνει ένα κενό κελί και μετακινεί την κεφαλή δεξιά.

$$\delta(c, \square) = (e, \square, R)$$

Κατάσταση *f*

Αφήνει ένα κενό κελί και επιστρέφει στην κατάσταση *b*.

$$\delta(f, \square) = (b, \square, R)$$

$$b \longrightarrow c \longrightarrow e \longrightarrow f \longrightarrow b$$

Κάθε πλήρης κύκλος γράφει το μοτίβο $0\square1\square$ σε τέσσερα διαδοχικά κελιά.

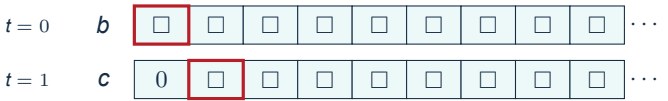
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση b, με την κεφαλή στο πρώτο κελί.



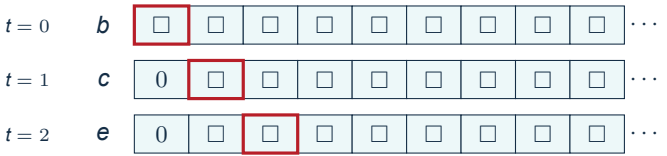
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση *b*, με την κεφαλή στο πρώτο κελί.



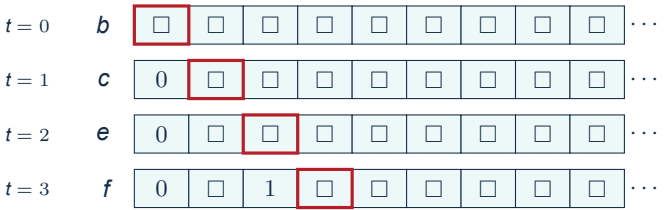
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση b, με την κεφαλή στο πρώτο κελί.



Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση *b*, με την κεφαλή στο πρώτο κελί.



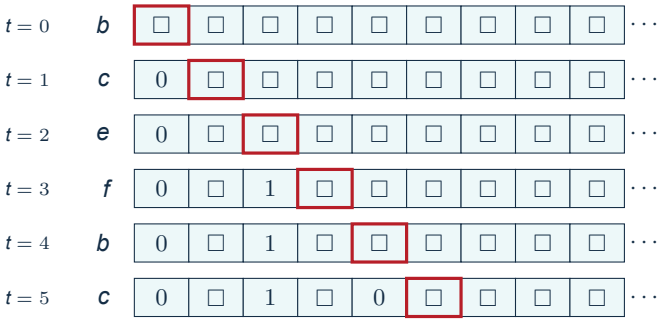
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση *b*, με την κεφαλή στο πρώτο κελί.



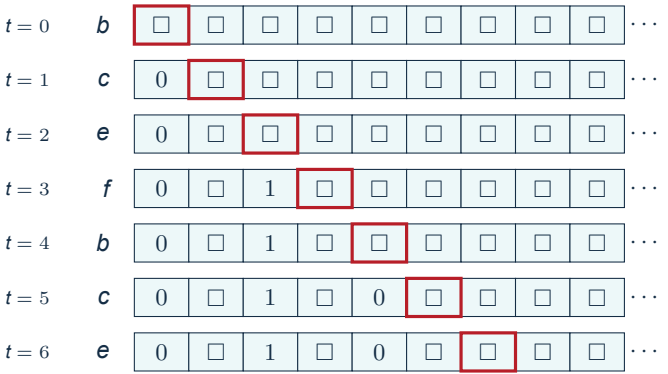
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση *b*, με την κεφαλή στο πρώτο κελί.



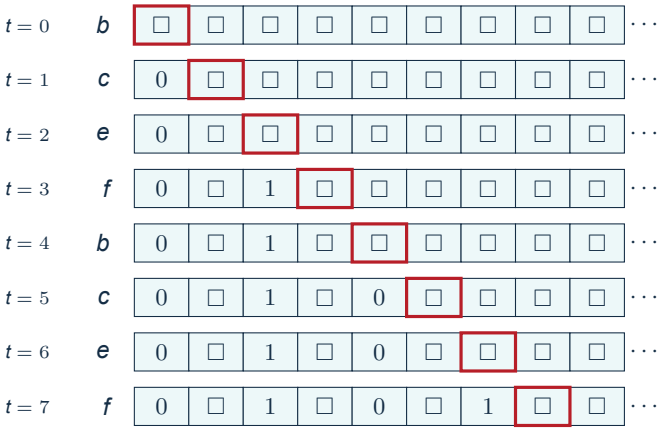
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση *b*, με την κεφαλή στο πρώτο κελί.



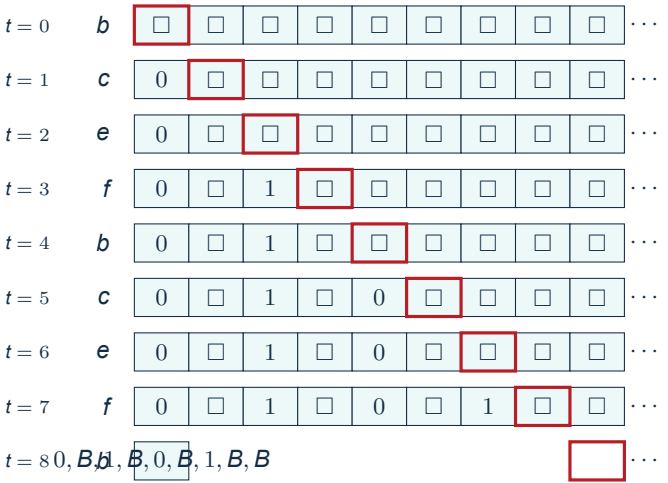
Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση b, με την κεφαλή στο πρώτο κελί.



Εκτέλεση της μηχανής: βήμα προς βήμα

Η μηχανή ξεκινά με κενή ταινία, στην κατάσταση *b*, με την κεφαλή στο πρώτο κελί.



Ο κύκλος επαναλαμβάνεται επ' άπειρον.

Η ακολουθία των διαμορφώσεων

$$(\boldsymbol{b}, \varepsilon, 1)$$

Η ακολουθία των διαμορφώσεων

$$(\boldsymbol{b}, \varepsilon, 1)$$
$$\rightsquigarrow (\boldsymbol{c}, 0, 2)$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3)
\end{aligned}
$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1, 4)
\end{aligned}
$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1, 4) \\
\rightsquigarrow\ & (\boldsymbol{b}, 0\square 1, 5)
\end{aligned}
$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1, 4) \\
\rightsquigarrow\ & (\boldsymbol{b}, 0\square 1, 5) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0\square 1\square 0, 6)
\end{aligned}
$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1, 4) \\
\rightsquigarrow\ & (\boldsymbol{b}, 0\square 1, 5) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0\square 1\square 0, 6) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0\square 1\square 0, 7)
\end{aligned}
$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1, 4) \\
\rightsquigarrow\ & (\boldsymbol{b}, 0\square 1, 5) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0\square 1\square 0, 6) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0\square 1\square 0, 7) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1\square 0\square 1, 8)
\end{aligned}
$$

Η ακολουθία των διαμορφώσεων

$$
\begin{aligned}
& (\boldsymbol{b}, \varepsilon, 1) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0, 2) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0, 3) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1, 4) \\
\rightsquigarrow\ & (\boldsymbol{b}, 0\square 1, 5) \\
\rightsquigarrow\ & (\boldsymbol{c}, 0\square 1\square 0, 6) \\
\rightsquigarrow\ & (\boldsymbol{e}, 0\square 1\square 0, 7) \\
\rightsquigarrow\ & (\boldsymbol{f}, 0\square 1\square 0\square 1, 8) \\
\rightsquigarrow\ & \cdots
\end{aligned}
$$

Παρατήρηση

Η μηχανή επιστρέφει στην ίδια κατάσταση *b*, αλλά όχι στην ίδια διαμόρφωση!

Γιατί η μηχανή δεν τερματίζει;

Θεώρημα 3

Αν η μηχανή ξεκινήσει με κενή ταινία, εκτελεί άπειρα βήματα.

Απόδειξη.
Με επαγωγή στον αριθμό των βημάτων.

Βάση: Αρχικά, η κεφαλή βρίσκεται στο πρώτο κελί, το οποίο είναι κενό.

Επαγωγικό βήμα: Υποθέτουμε ότι μετά από t βήματα η κεφαλή βρίσκεται στο κελί $t+1$, το οποίο είναι κενό.

- Κάθε μετάβαση πάνω στο □ μετακινεί την κεφαλή δεξιά.
- Επομένως, η κεφαλή μετακινείται στο κελί $t+2$.
- Το κελί αυτό δεν έχει επισκεφθεί ποτέ η μηχανή, άρα είναι κενό.
- Η νέα κατάσταση ανήκει στο $\{b, c, e, f\}$.

Συνεπώς, υπάρχει πάντα επόμενη μετάβαση και η μηχανή δεν εισέρχεται ποτέ σε κατάσταση τερματισμού. □

Μια μηχανή ως γεννήτρια άπειρης ακολουθίας

Μετά από $4k$ βήματα, για $k \geq 1$, η διαμόρφωση της μηχανής είναι $\boxed{(b, (0\square 1\square)^{k-1}0\square 1, 4k+1)}$.



Αγνοώντας τα κενά σύμβολα, τα ψηφία που παράγονται διαδοχικά σχηματίζουν την άπειρη ακολουθία $\boxed{010101010101\ldots}$

Προσοχή

Σε κάθε πεπερασμένη χρονική στιγμή έχουν γραφτεί μόνο πεπερασμένα ψηφία. Η άπειρη ακολουθία περιγράφει το αποτέλεσμα της συνεχιζόμενης εκτέλεσης.

Άσκηση κατανόησης: ποια γλώσσα αναγνωρίζει η μηχανή;

Θεωρούμε τη μηχανή M των προηγούμενων διαφανειών με αλφάβητο εισόδου $\Sigma = \{0, 1\}$.

Ποια είναι η γλώσσα $L(M)$;

1. $L(M) = \{(01)^n : n \geq 0\}$.
2. $L(M) = \{\varepsilon\}$.
3. $L(M) = \emptyset$.
4. Η M δεν αναγνωρίζει καμία γλώσσα, διότι δεν τερματίζει για κενή είσοδο.

Άσκηση κατανόησης: ποια γλώσσα αναγνωρίζει η μηχανή;

Θεωρούμε τη μηχανή M των προηγούμενων διαφανειών με αλφάβητο εισόδου $\Sigma = \{0, 1\}$.

Ποια είναι η γλώσσα $L(M)$;

1. $L(M) = \{(01)^n : n \geq 0\}$.
2. $L(M) = \{\varepsilon\}$.
3. $L(M) = \emptyset$.
4. Η M δεν αναγνωρίζει καμία γλώσσα, διότι δεν τερματίζει για κενή είσοδο.

Σωστή απάντηση: 3

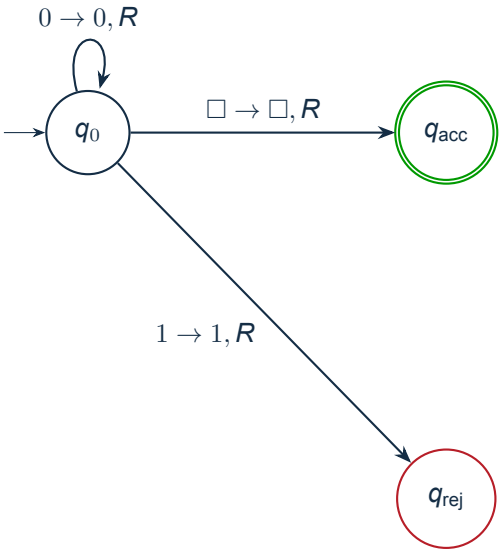
Η μηχανή δεν επισκέπτεται ποτέ την κατάσταση αποδοχής q_{acc}.

- Για $w = \varepsilon$, δεν τερματίζει.
- Για $w \neq \varepsilon$, απορρίπτει αμέσως.

Επομένως,

$$L(M) = \emptyset.$$

Turing Machine για $L = 0^*$.



Η γλώσσα μιας μηχανής Turing

Ορισμός 4

Η γλώσσα που αναγνωρίζει μια μηχανή Turing M είναι το σύνολο όλων των συμβολοσειρών που αποδέχεται η M:

$$\boxed{L(M) = \{w \in \Sigma^* : M \text{ αποδέχεται το } w\}.}$$

- Αν η M αποδέχεται το w, τότε $w \in L(M)$.
- Αν η M απορρίπτει το w, τότε $w \notin L(M)$.
- Αν η M δεν τερματίζει για το w, τότε επίσης $w \notin L(M)$.

Βασική παρατήρηση

Η γλώσσα $L(M)$ ορίζεται αποκλειστικά από τις συμβολοσειρές που αποδέχεται η M, ανεξάρτητα από τη συμπεριφορά της στις υπόλοιπες.

Πηγή: Sipser, Introduction to the Theory of Computation.

Αναγνωρίσιμες γλώσσες κατά Turing

Ορισμός 5

Μια γλώσσα $L \subseteq \Sigma^*$ ονομάζεται **αναγνωρίσιμη κατά Turing** (Turing-recognizable), αν υπάρχει μηχανή Turing M τέτοια ώστε

$$L(M) = L.$$

Ισοδύναμα, υπάρχει μηχανή Turing M τέτοια ώστε:

- Για κάθε $w \in L$, η M τερματίζει και αποδέχεται το w.
- Για κάθε $w \notin L$, η M είτε απορρίπτει το w είτε δεν τερματίζει ποτέ.

	$w \in L$	$w \notin L$
Αποδοχή	Ναι	Όχι
Απόρριψη	Όχι	Επιτρέπεται
Μη τερματισμός	Όχι	Επιτρέπεται

Πηγή: Sipser, Introduction to the Theory of Computation.

Αποφασίσιμες/υπολογίσιμες γλώσσες κατά Turing

Ορισμός 6

Μια μηχανή Turing που τερματίζει για κάθε δυνατή είσοδο ονομάζεται **αποφασιστής** (decider).

Ορισμός 7

Μια γλώσσα L ονομάζεται **αποφασίσιμη κατά Turing** (Turing-decidable), αν υπάρχει αποφασιστής M τέτοιος ώστε

$$L(M) = L.$$

Επομένως, για κάθε $w \in \Sigma^*$,

$$M(w) = \begin{cases} \text{αποδοχή}, & \text{αν } w \in L, \\ \text{απόρριψη}, & \text{αν } w \notin L. \end{cases}$$

Η κρίσιμη διαφορά

Ο αποφασιστής δίνει πάντοτε απάντηση, είτε θετική είτε αρνητική, σε πεπερασμένο χρόνο.

Πηγή: Sipser, Introduction to the Theory of Computation.

Η χαρακτηριστική συνάρτηση μιας γλώσσας

Ορισμός 8

Για κάθε γλώσσα $L \subseteq \Sigma^*$ ορίζουμε τη **χαρακτηριστική συνάρτηση** (characteristic function)

$$\chi_L : \Sigma^* \longrightarrow \{0, 1\}$$

ως εξής:

$$\chi_L(w) = \begin{cases} 1, & \text{αν } w \in L, \\ 0, & \text{αν } w \notin L. \end{cases}$$

Παράδειγμα

Έστω $L = \{0^{2n} : n \geq 0\}$.

$$\chi_L(00) = 1, \qquad \chi_L(000) = 0, \qquad \chi_L(\varepsilon) = 1.$$

Κάθε γλώσσα έχει χαρακτηριστική συνάρτηση. Μπορούμε όμως πάντοτε να την υπολογίσουμε;

Αποφασισιμότητα και υπολογισιμότητα

Θεώρημα 9

Μια γλώσσα L είναι αποφασίσιμη κατά Turing αν και μόνο αν η χαρακτηριστική της συνάρτηση χ_L είναι υπολογίσιμη κατά Turing.



Ισοδύναμη ορολογία

- Αποφασίσιμες γλώσσες (decidable languages).
- Υπολογίσιμες γλώσσες (computable languages).
- Αναδρομικές γλώσσες (recursive languages).

Αποφασισιμότητα και υπολογισιμότητα: απόδειξη

(⇒) Έστω ότι η L είναι αποφασίσιμη.
Υπάρχει επομένως αποφασιστής M για τη L.

- Εκτελούμε τη M με είσοδο w.
- Αν η M αποδεχθεί, επιστρέφουμε 1.
- Αν η M απορρίψει, επιστρέφουμε 0.

Εφόσον η M τερματίζει πάντοτε, υπολογίζουμε τη $\chi_L(w)$ για κάθε w.

Αποφασισιμότητα και υπολογισιμότητα: απόδειξη

(⇒) Έστω ότι η L είναι αποφασίσιμη.
Υπάρχει επομένως αποφασιστής M για τη L.

- Εκτελούμε τη M με είσοδο w.
- Αν η M αποδεχθεί, επιστρέφουμε 1.
- Αν η M απορρίψει, επιστρέφουμε 0.

Εφόσον η M τερματίζει πάντοτε, υπολογίζουμε τη $\chi_L(w)$ για κάθε w.

(⇐) Έστω ότι η χ_L είναι υπολογίσιμη κατά Turing.
Κατασκευάζουμε μια μηχανή Turing M η οποία:

- Υπολογίζει τη $\chi_L(w)$.
- Αν $\chi_L(w) = 1$, αποδέχεται.
- Αν $\chi_L(w) = 0$, απορρίπτει.

Η χ_L είναι ολική υπολογίσιμη συνάρτηση, άρα η M τερματίζει πάντοτε και αποφασίζει τη L.

□

Αναγνώριση και μερική υπολογισιμότητα

Για μια γλώσσα $L \subseteq \Sigma^*$ ορίζουμε τη **μερική χαρακτηριστική συνάρτηση**

$$\psi_L(w) = \begin{cases} 1, & \text{αν } w \in L, \\ \text{μη ορισμένη}, & \text{αν } w \notin L. \end{cases}$$

Θεώρημα 10

Μια γλώσσα L είναι αναγνωρίσιμη κατά Turing αν και μόνο αν η ψ_L είναι μερικώς υπολογίσιμη κατά Turing.

	$\chi_L(w)$	$\psi_L(w)$
$w \in L$	1	1
$w \notin L$	0	Μη ορισμένη

Αναγνώριση και μερική υπολογισιμότητα: απόδειξη

(⇒) Έστω M ένας αναγνωριστής της L.
Κατασκευάζουμε μια μηχανή N που προσομοιώνει τη M με είσοδο w.

- Αν η M αποδεχθεί, η N επιστρέφει 1.
- Αν η M απορρίψει, η N εισέρχεται σε έναν ατέρμονα υπολογισμό.
- Αν η M δεν τερματίσει, ούτε η N τερματίζει.

Επομένως, η N υπολογίζει τη μερική συνάρτηση ψ_L.

Αναγνώριση και μερική υπολογισιμότητα: απόδειξη

(⇒) Έστω M ένας αναγνωριστής της L.
Κατασκευάζουμε μια μηχανή N που προσομοιώνει τη M με είσοδο w.

- Αν η M αποδεχθεί, η N επιστρέφει 1.
- Αν η M απορρίψει, η N εισέρχεται σε έναν ατέρμονα υπολογισμό.
- Αν η M δεν τερματίσει, ούτε η N τερματίζει.

Επομένως, η N υπολογίζει τη μερική συνάρτηση ψ_L.

(⇐) Έστω ότι μια μηχανή N υπολογίζει τη ψ_L.
Κατασκευάζουμε έναν αναγνωριστή M που εκτελεί τη $N(w)$ και αποδέχεται αν αυτή επιστρέψει 1.

- Αν $w \in L$, η M τελικά αποδέχεται.
- Αν $w \notin L$, η M δεν τερματίζει.

Άρα $L(M) = L$.

□

Αποφασισιμότητα και ύπαρξη αλγορίθμου

Θεωρούμε ένα υπολογιστικό πρόβλημα με απάντηση **ΝΑΙ** ή **ΟΧΙ**.

Κωδικοποιούμε κάθε στιγμιότυπο του προβλήματος ως μια συμβολοσειρά w και ορίζουμε τη γλώσσα

$$L = \{w : \text{η απάντηση για το } w \text{ είναι ΝΑΙ}\}.$$



Θέση Church–Turing (ξανά)

Η ύπαρξη αλγορίθμου που επιλύει ένα πρόβλημα απόφασης ταυτίζεται, σύμφωνα με τη θέση Church–Turing, με την ύπαρξη μιας μηχανής Turing που αποφασίζει την αντίστοιχη γλώσσα.

Παραδείγματα αποφασίσιμων γλωσσών

Γλώσσα	Αλγοριθμική μέθοδος
Κάθε κανονική γλώσσα L	Προσομοίωση ενός DFA.
Κάθε γλώσσα χωρίς συμφραζόμενα L	Αλγόριθμος CYK.
$\{0^n 1^n 2^n : n \geq 0\}$	Αντιστοίχιση συμβόλων με διαδοχικές σημάνσεις.
$\{\langle n \rangle : n \text{ πρώτος}\}$	Έλεγχος διαιρετότητας.
$\{\langle G \rangle : G \text{ συνεκτικό}\}$	Αναζήτηση BFS ή DFS.
$\{\langle G, k \rangle : G \text{ έχει κλίκα μεγέθους } k\}$	Εξαντλητική αναζήτηση υποσυνόλων κορυφών.

Προσοχή

Αποφασίσιμο δεν σημαίνει απαραίτητα υπολογιστικά αποδοτικό!

Υπολογισιμότητα έναντι πολυπλοκότητας

Θεωρία υπολογισιμότητας

Υπάρχει αλγόριθμος;

- Αποφασίσιμες γλώσσες.
- Αναγνωρίσιμες γλώσσες.
- Μη αποφασίσιμες γλώσσες.
- Μη αναγνωρίσιμες γλώσσες.

Θεωρία πολυπλοκότητας

Πόσο κοστίζει ο υπολογισμός;

- Χρόνος εκτέλεσης.
- Απαιτούμενη μνήμη.
- Πολυωνυμικός χρόνος.
- Κλάσεις *P*, *NP*, *PSPACE*.

Παράδειγμα

Έστω ένας αποφασιστής με χρόνο εκτέλεσης

$$T(n) = 2^{2^n}.$$

Παρά τον τεράστιο χρόνο εκτέλεσης, η γλώσσα του παραμένει αποφασίσιμη.

Η ύπαρξη λύσης και το κόστος της είναι διαφορετικά μαθηματικά ερωτήματα.

Η ιεραρχία των γλωσσών



$$\text{REG} \subsetneq \text{CFL} \subsetneq \text{DEC} \subsetneq \text{RE} \subsetneq \mathcal{P}(\Sigma^*)$$

Για αλφάβητο Σ με τουλάχιστον δύο σύμβολα, όλοι οι παραπάνω εγκλεισμοί είναι γνήσιοι.

Άσκηση κατανόησης: υπολογισιμότητα

Ποιες από τις παρακάτω προτάσεις είναι σωστές;

1. Κάθε αποφασίσιμη γλώσσα έχει υπολογίσιμη χαρακτηριστική συνάρτηση.
2. Κάθε γλώσσα έχει υπολογίσιμη χαρακτηριστική συνάρτηση.
3. Κάθε αναγνωρίσιμη γλώσσα είναι αποφασίσιμη.
4. Υπάρχουν αποφασίσιμες γλώσσες που δεν είναι κανονικές.
5. Αν ένας αλγόριθμος απαιτεί εκθετικό χρόνο, τότε η αντίστοιχη γλώσσα δεν είναι αποφασίσιμη.

Άσκηση κατανόησης: υπολογισιμότητα

Ποιες από τις παρακάτω προτάσεις είναι σωστές;

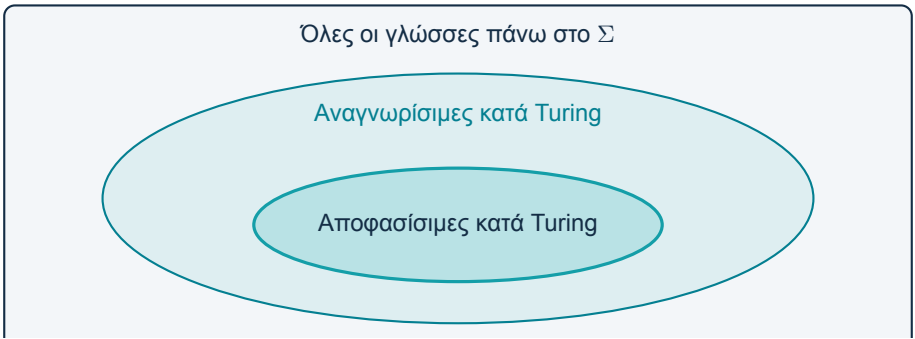
1. Κάθε αποφασίσιμη γλώσσα έχει υπολογίσιμη χαρακτηριστική συνάρτηση.
2. Κάθε γλώσσα έχει υπολογίσιμη χαρακτηριστική συνάρτηση.
3. Κάθε αναγνωρίσιμη γλώσσα είναι αποφασίσιμη.
4. Υπάρχουν αποφασίσιμες γλώσσες που δεν είναι κανονικές.
5. Αν ένας αλγόριθμος απαιτεί εκθετικό χρόνο, τότε η αντίστοιχη γλώσσα δεν είναι αποφασίσιμη.

Σωστές απαντήσεις: 1 και 4

- Η χαρακτηριστική συνάρτηση του HALT δεν είναι υπολογίσιμη.
- Το HALT είναι αναγνωρίσιμο αλλά όχι αποφασίσιμο.
- Η $\{0^n 1^n : n \geq 0\}$ είναι αποφασίσιμη αλλά όχι κανονική.
- Ο εκθετικός χρόνος παραμένει πεπερασμένος για κάθε πεπερασμένη είσοδο.

Αναγνώριση έναντι απόφασης

Είσοδος	Αναγνωριστής	Αποφασιστής
$w \in L$	Αποδοχή	Αποδοχή
$w \notin L$	Απόρριψη ή ∞	Απόρριψη



Ίδια γλώσσα, διαφορετικές μηχανές

Θεωρούμε την αποφασίσιμη γλώσσα

$$L = \{w \in \{0,1\}^* : w \text{ περιέχει τουλάχιστον ένα } 1\}.$$

Μηχανή M_1

1. Σαρώνει την είσοδο από αριστερά προς τα δεξιά.
2. Αν συναντήσει 1, αποδέχεται.
3. Αν συναντήσει $\square$, απορρίπτει.

Τερματίζει πάντοτε.

Μηχανή M_2

1. Σαρώνει την είσοδο από αριστερά προς τα δεξιά.
2. Αν συναντήσει 1, αποδέχεται.
3. Αν συναντήσει $\square$, κινείται δεξιά για πάντα.

Δεν τερματίζει πάντοτε.

$$L(M_1) = L(M_2) = L.$$

Ίδια γλώσσα, διαφορετικές μηχανές

Θεωρούμε την αποφασίσιμη γλώσσα

$$L = \{w \in \{0,1\}^* : w \text{ περιέχει τουλάχιστον ένα } 1\}.$$

Μηχανή M_1

1. Σαρώνει την είσοδο από αριστερά προς τα δεξιά.
2. Αν συναντήσει 1, αποδέχεται.
3. Αν συναντήσει $\square$, απορρίπτει.

Τερματίζει πάντοτε.

Μηχανή M_2

1. Σαρώνει την είσοδο από αριστερά προς τα δεξιά.
2. Αν συναντήσει 1, αποδέχεται.
3. Αν συναντήσει $\square$, κινείται δεξιά για πάντα.

Δεν τερματίζει πάντοτε.

$$L(M_1) = L(M_2) = L.$$

Συμπέρασμα

Η M_1 αποφασίζει τη L, ενώ η M_2 την αναγνωρίζει αλλά δεν την αποφασίζει.

Άσκηση κατανόησης

Έστω μια μηχανή Turing M πάνω στο αλφάβητο $\Sigma = \{0, 1\}$, με την εξής συμπεριφορά:

- Αν η είσοδος αρχίζει από 1, αποδέχεται.
- Αν η είσοδος αρχίζει από 0, δεν τερματίζει.
- Αν η είσοδος είναι ε, απορρίπτει.

Ποια από τις παρακάτω προτάσεις είναι σωστή;

1. Η M δεν αναγνωρίζει καμία γλώσσα.
2. Η M αναγνωρίζει τη γλώσσα $1\{0, 1\}^*$ και είναι αποφασιστής.
3. Η M αναγνωρίζει τη γλώσσα $1\{0, 1\}^*$, αλλά δεν είναι αποφασιστής.
4. Η γλώσσα $L(M)$ είναι μη αποφασίσιμη.

Άσκηση κατανόησης

Έστω μια μηχανή Turing M πάνω στο αλφάβητο $\Sigma = \{0, 1\}$, με την εξής συμπεριφορά:

- Αν η είσοδος αρχίζει από 1, αποδέχεται.
- Αν η είσοδος αρχίζει από 0, δεν τερματίζει.
- Αν η είσοδος είναι ε, απορρίπτει.

Ποια από τις παρακάτω προτάσεις είναι σωστή;

1. Η M δεν αναγνωρίζει καμία γλώσσα.
2. Η M αναγνωρίζει τη γλώσσα $1\{0, 1\}^*$ και είναι αποφασιστής.
3. Η M αναγνωρίζει τη γλώσσα $1\{0, 1\}^*$, αλλά δεν είναι αποφασιστής.
4. Η γλώσσα $L(M)$ είναι μη αποφασίσιμη.

Σωστή απάντηση: 3

Η M δεν τερματίζει για εισόδους που αρχίζουν από 0.
Ωστόσο, η γλώσσα $1\{0, 1\}^*$ είναι κανονική και επομένως αποφασίσιμη.

Θεώρημα 11

Μια γλώσσα L είναι αποφασίσιμη αν και μόνο αν τόσο η L όσο και η συμπληρωματική της γλώσσα

$$\overline{L} = \Sigma^* \setminus L$$

είναι αναγνωρίσιμες κατά Turing.

Απόδειξη ($\Rightarrow$).
Έστω M ένας αποφασιστής της L.

- Προφανώς, η M αναγνωρίζει τη L.
- Αν ανταλλάξουμε τις καταστάσεις αποδοχής και απόρριψης της M, προκύπτει ένας αποφασιστής της $\overline{L}$.

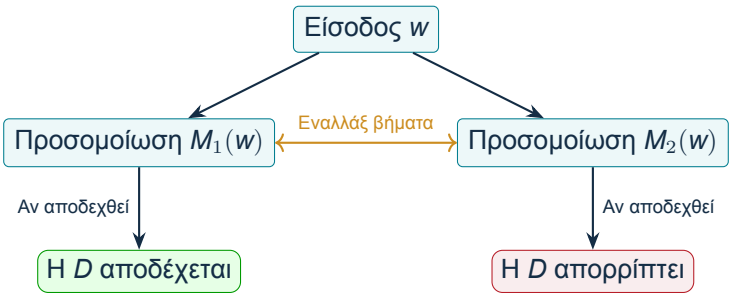
Επομένως, τόσο η L όσο και η $\overline{L}$ είναι αναγνωρίσιμες.

Πηγή: Sipser, Introduction to the Theory of Computation.

Αναγνώριση και συμπλήρωμα: απόδειξη

Απόδειξη ($\Leftarrow$).
Έστω M_1 ένας αναγνωριστής της L και M_2 ένας αναγνωριστής της $\overline{L}$.

Κατασκευάζουμε μια μηχανή Turing D που, για κάθε είσοδο w, εκτελεί τα εξής:



- Αν $w \in L$, η M_1 τελικά αποδέχεται.
- Αν $w \notin L$, η M_2 τελικά αποδέχεται.

Άρα η D τερματίζει πάντοτε και αποφασίζει τη L. □

Το πρόβλημα τερματισμού (Halting Problem)

Θεωρούμε τη γλώσσα

$$\boxed{\text{HALT}_{\text{TM}} = \{\langle M, w\rangle : M \text{ τερματίζει για είσοδο } w\}.}$$

- Η είσοδος είναι η κωδικοποίηση μιας μηχανής Turing M και μιας συμβολοσειράς w.
- Θέλουμε να γνωρίζουμε αν η M θα τερματίσει όταν εκτελεστεί με είσοδο w.
- Δεν μας ενδιαφέρει αν η M αποδέχεται ή απορρίπτει, αλλά μόνο αν τερματίζει.

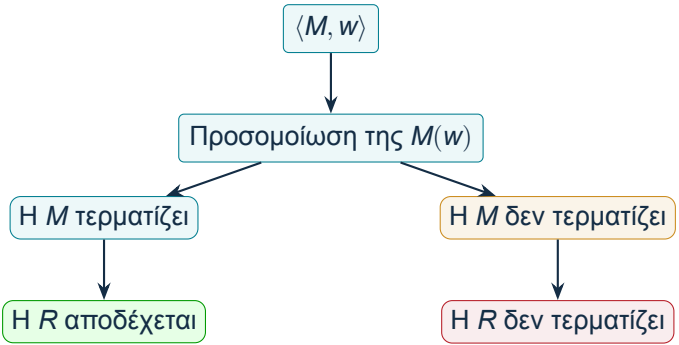
Θεώρημα 12

Η γλώσσα HALT_{TM} *είναι αναγνωρίσιμη κατά Turing, αλλά δεν είναι αποφασίσιμη κατά Turing.*

Πηγή: Sipser, Introduction to the Theory of Computation.

Γιατί το HALT είναι αναγνωρίσιμο;

Κατασκευάζουμε μια μηχανή Turing R που, για είσοδο $\langle M, w \rangle$, προσομοιώνει την εκτέλεση της M με είσοδο w.



$$L(R) = \text{HALT}_{\text{TM}}.$$

Γιατί το HALT δεν είναι αποφασίσιμο;

Έστω ότι προσομοιώνουμε μια μηχανή Turing M με είσοδο w.

Περίπτωση 1: Η M τερματίζει

Αρκεί να περιμένουμε μέχρι να τερματίσει. Τότε γνωρίζουμε με βεβαιότητα ότι

$$\langle M, w \rangle \in \mathrm{HALT}_{\mathrm{TM}}.$$

Γιατί το HALT δεν είναι αποφασίσιμο;

Έστω ότι προσομοιώνουμε μια μηχανή Turing M με είσοδο w.

Περίπτωση 1: Η M τερματίζει

Αρκεί να περιμένουμε μέχρι να τερματίσει. Τότε γνωρίζουμε με βεβαιότητα ότι

$$\langle M, w \rangle \in \mathrm{HALT}_{\mathrm{TM}}.$$

Περίπτωση 2: Η M δεν τερματίζει

Μετά από 10^6, 10^{100} ή οποιονδήποτε πεπερασμένο αριθμό βημάτων, πώς γνωρίζουμε ότι η M δεν θα τερματίσει στο επόμενο βήμα;

Γιατί το HALT δεν είναι αποφασίσιμο;

Έστω ότι προσομοιώνουμε μια μηχανή Turing M με είσοδο w.

Περίπτωση 1: Η M τερματίζει

Αρκεί να περιμένουμε μέχρι να τερματίσει. Τότε γνωρίζουμε με βεβαιότητα ότι

$$\langle M, w \rangle \in \mathrm{HALT}_{\mathrm{TM}}.$$

Περίπτωση 2: Η M δεν τερματίζει

Μετά από 10^6, 10^{100} ή οποιονδήποτε πεπερασμένο αριθμό βημάτων, πώς γνωρίζουμε ότι η M δεν θα τερματίσει στο επόμενο βήμα;

- Η απλή προσομοίωση δεν μπορεί να δώσει πάντοτε αρνητική απάντηση.
- Θα αποδείξουμε ότι δεν υπάρχει ούτε κάποια άλλη αλγοριθμική μέθοδος που να αποφασίζει το πρόβλημα τερματισμού.

$$\boxed{\mathrm{HALT}_{\mathrm{TM}} \in \mathrm{RE} \setminus \mathrm{DEC}}$$