1. Από Δομές σε Πλήθος Λύσεων

- Στις προηγούμενες διαλέξεις:

– ορίσαμε δομές (διαδρομές, δέντρα)

– αναπτύξαμε μεθόδους εύρεσης λύσεων

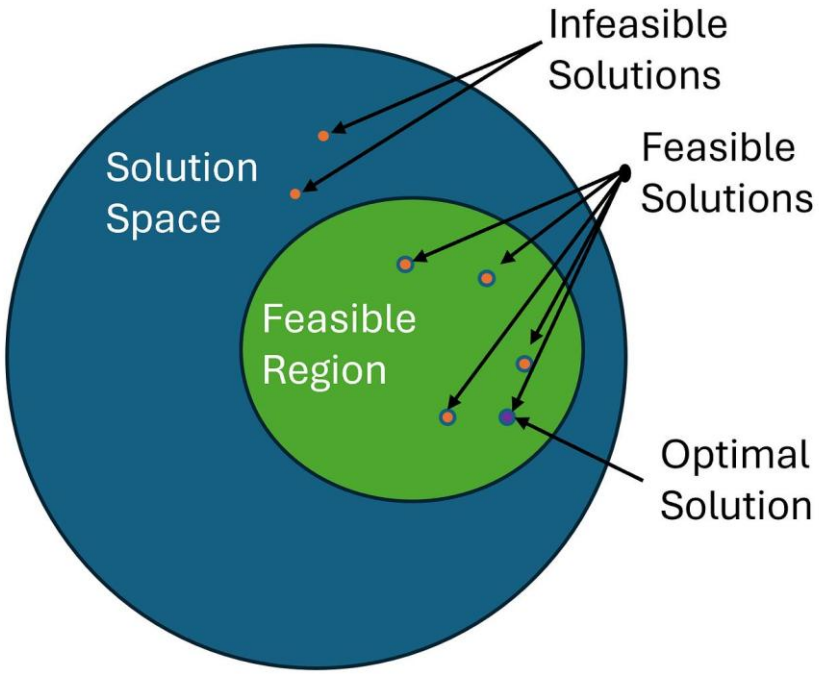
- Σε αυτή τη διάλεξη:

– εξετάζουμε το πλήθος των δυνατών λύσεων

- Το πρόβλημα:

– δεν είναι μόνο η εύρεση λύσης

– αλλά η κατανόηση του μεγέθους του χώρου λύσεων



2. Τι Κάνουμε σε Αυτή τη Διάλεξη

- Μέχρι τώρα:
  – διατυπώσαμε προβλήματα
  – αναπτύξαμε μεθόδους επίλυσης

- Σε αυτή τη διάλεξη:
  – δεν λύνουμε νέο πρόβλημα

- Εξετάζουμε:
  – πόσες λύσεις υπάρχουν
  – πόσο δύσκολο είναι να τις εξετάσουμε

- Δηλαδή:
  – αναλύουμε τον χώρο λύσεων
  – και όχι τη λύση

- Στόχος:
  – κατανόηση των ορίων της επίλυσης

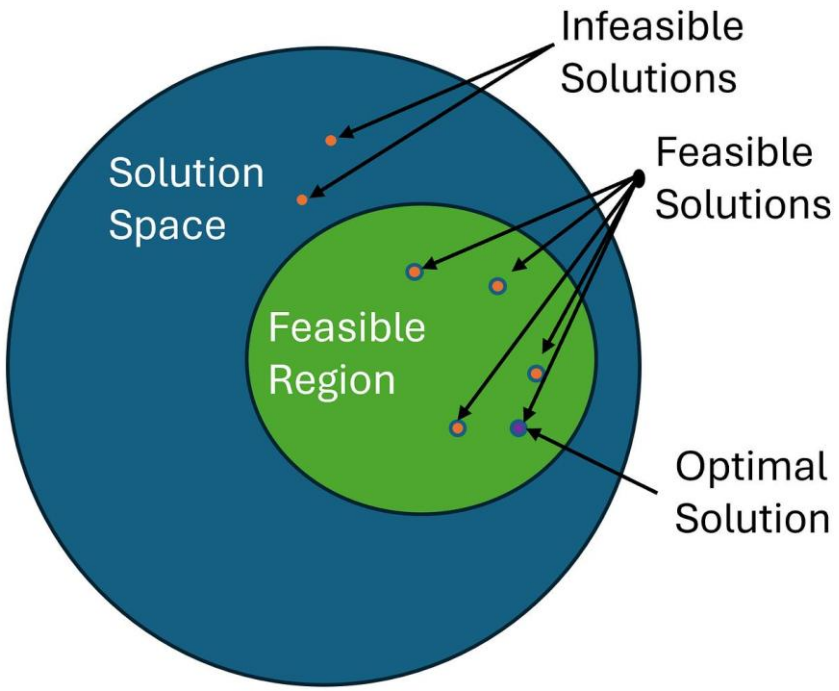
3. Τι Μετράμε

- Σε κάθε πρόβλημα:
  – υπάρχει σύνολο εφικτών λύσεων

- Το πλήθος αυτό:
  – εξαρτάται από τη δομή του προβλήματος

- Μετράμε:
  – πόσες διαφορετικές λύσεις υπάρχουν

- Η μέτρηση:
  – αποκαλύπτει τη δυσκολία του προβλήματος



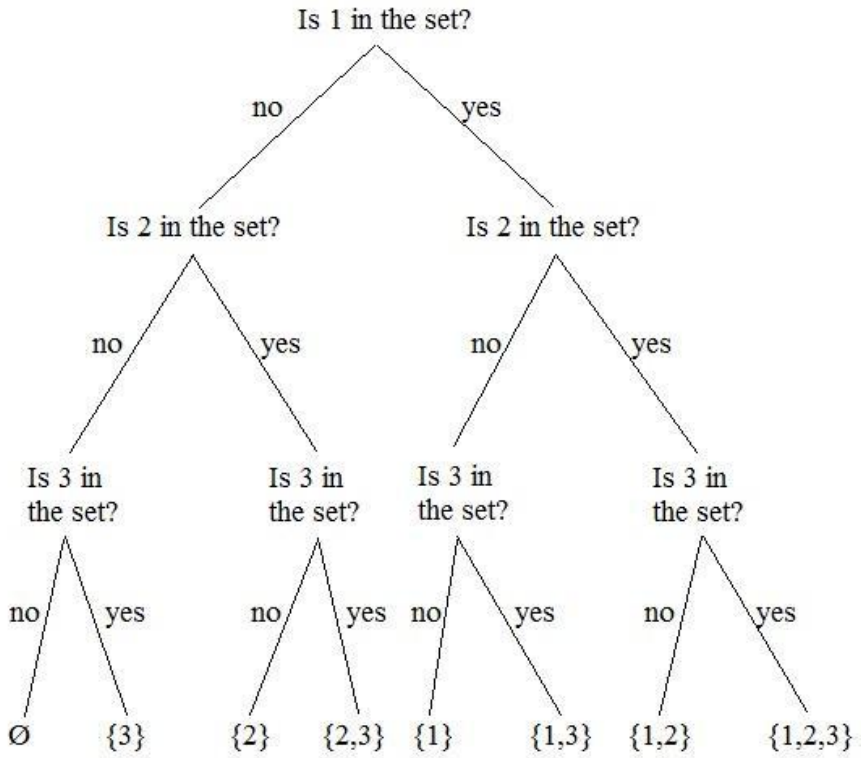
4. Συνδυαστική Φύση

- Πολλά προβλήματα:
  – απαιτούν επιλογή υποσυνόλου

- Για n δυαδικές μεταβλητές:
  – κάθε μεταβλητή έχει 2 επιλογές

- Συνολικές λύσεις:
  – 2^n

- Η αύξηση:
  – είναι εκθετική



5. Παράδειγμα

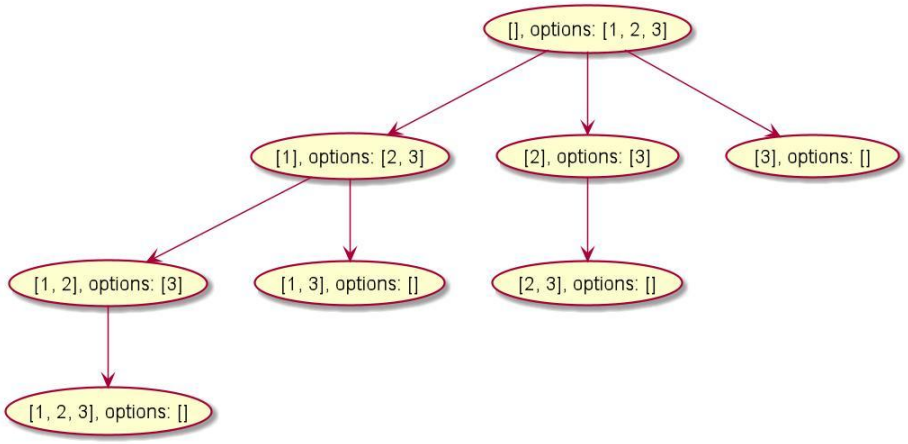
- Έστω n έργα

- Κάθε έργο:
  – επιλέγεται ή όχι

- Πλήθος δυνατών συνδυασμών:
  – 2^n

- Για n = 10:
  – 1024 λύσεις

- Για n = 20:
  – πάνω από 1 εκατομμύριο



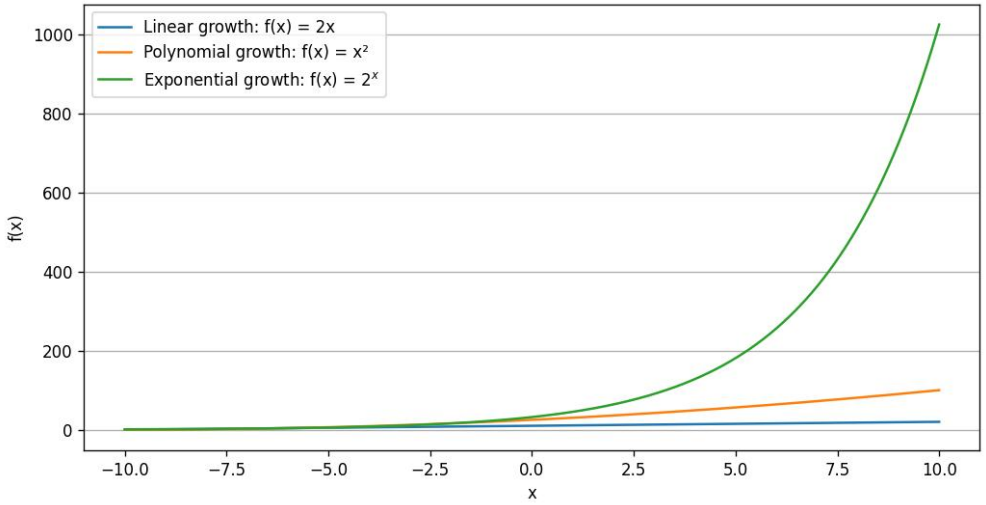
6. Εκθετική Αύξηση

- Η αύξηση:
  – δεν είναι γραμμική

- Κάθε νέα μεταβλητή:
  – διπλασιάζει τις λύσεις

- Η διαφορά:
  – γίνεται γρήγορα πολύ μεγάλη

- Το αποτέλεσμα:
  – πρακτική δυσκολία επίλυσης



7. Γιατί Μας Ενδιαφέρει το Πλήθος Λύσεων

- Σε πολλά προβλήματα:

– δεν γνωρίζουμε ποια λύση είναι βέλτιστη

- Μία προσέγγιση:

– εξέταση όλων των δυνατών λύσεων

- Αν το πλήθος είναι μικρό:

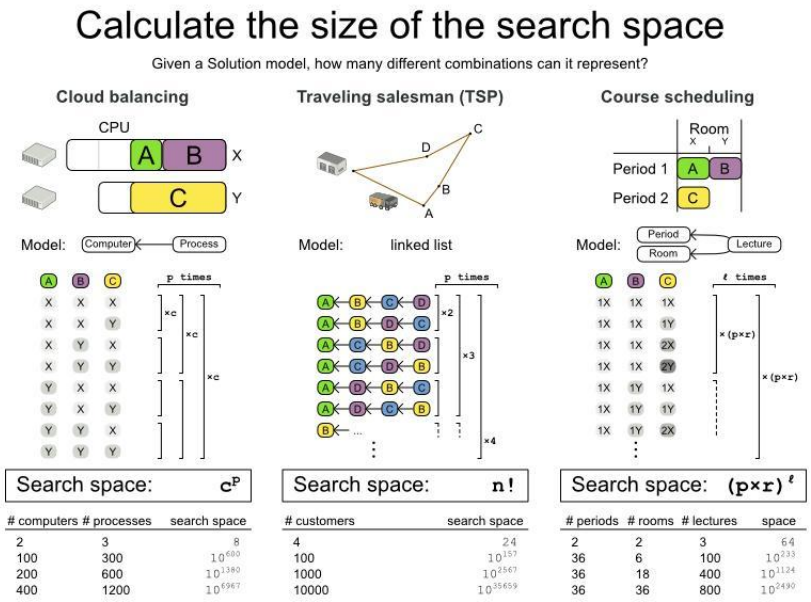
– η προσέγγιση είναι εφικτή

- Αν το πλήθος είναι μεγάλο:

– η προσέγγιση αποτυγχάνει

- Συνεπώς:

– το πλήθος λύσεων καθορίζει αν το πρόβλημα μπορεί να λυθεί



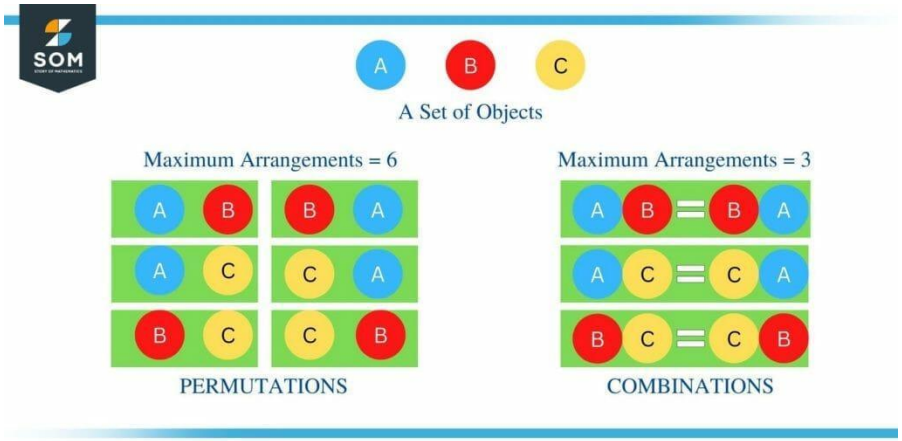
8. Συνδυασμοί και Διατάξεις

- Δεν εξετάζουμε μόνο υποσύνολα

- Υπάρχουν:
  - διατάξεις
  - επιλογές σειράς

- Για n στοιχεία:
  - οι διατάξεις είναι n!

- Η αύξηση:
  - είναι ταχύτερη από εκθετική



9. Παράδειγμα Διατάξεων

- Για n = 5:
  – 5! = 120

- Για n = 10:
  – 10! ≈ 3.6 εκατομμύρια

- Για n = 20:
  – 20! είναι εξαιρετικά μεγάλος αριθμός

- Η απαρίθμηση:
  – δεν είναι πρακτικά εφικτή

n! = n
0 = 1
1 = 1
2 = 2
3 = 6
4 = 24
5 = 120
6 = 720
7 = 5.040
8 = 40.320
9 = 362.880
10 = 3.628.800
11 = 39.916.800
12 = 479.001.600

10. Από Συνδυαστική σε Πολυπλοκότητα

- Το πλήθος λύσεων:

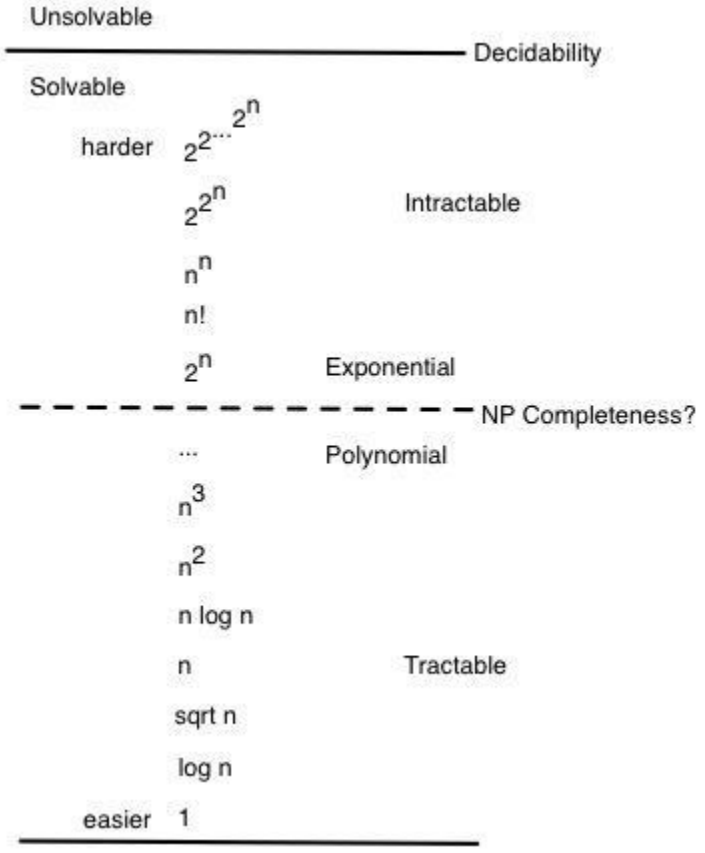
– επηρεάζει άμεσα τον χρόνο επίλυσης

- Περισσότερες λύσεις:

– απαιτούν περισσότερους υπολογισμούς

- Η πολυπλοκότητα:

– συνδέει το μέγεθος του προβλήματος με τον χρόνο επίλυσης



11. Τι Είναι Πολυπλοκότητα

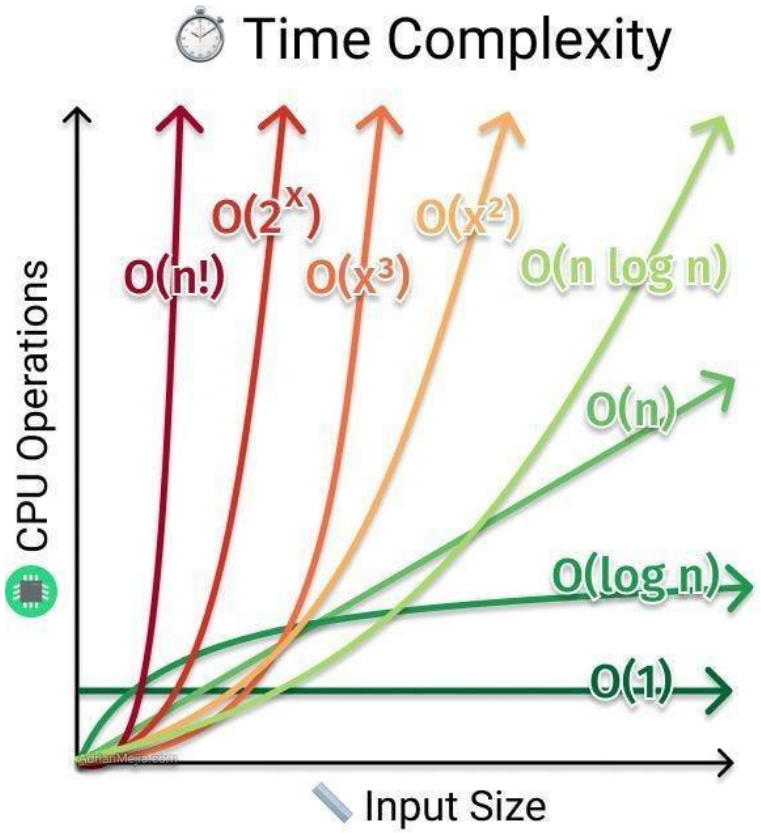
- Η πολυπλοκότητα:
  – μετρά πόσο δύσκολο είναι ένα πρόβλημα

- Εκφράζεται:
  – ως συνάρτηση του μεγέθους n

- Δεν μετράμε:
  – πραγματικό χρόνο σε δευτερόλεπτα

- Μετράμε:
  – πόσα υπολογιστικά βήματα απαιτούνται για την επίλυση

- Εξετάζουμε:
  – πώς αυξάνεται ο αριθμός των βημάτων καθώς αυξάνεται το n



12. Πώς Συνδέεται ο Χρόνος με την Επίλυση

- Κάθε μέθοδος:

– αποτελείται από ακολουθία υπολογιστικών βημάτων

- Κάθε βήμα:

– απαιτεί χρόνο εκτέλεσης

- Περισσότερα βήματα:

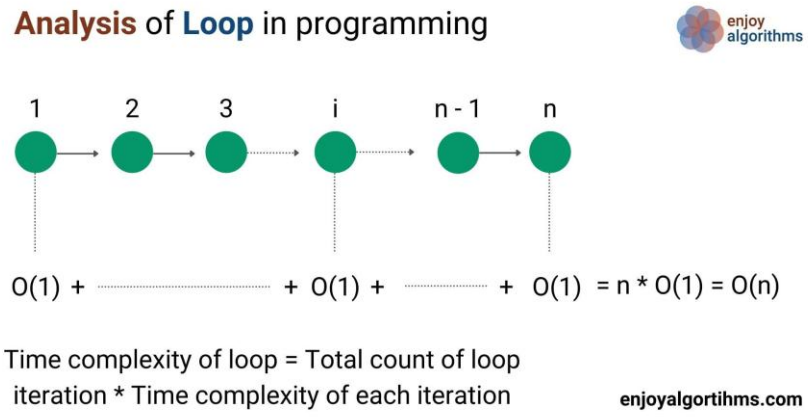
– συνεπάγονται περισσότερο χρόνο

- Αν μία μέθοδος εξετάζει όλες τις λύσεις:

– ο αριθμός βημάτων είναι ίσος με τον αριθμό λύσεων

- Συνεπώς:

– ο χρόνος εξαρτάται από το πλήθος των λύσεων που εξετάζονται



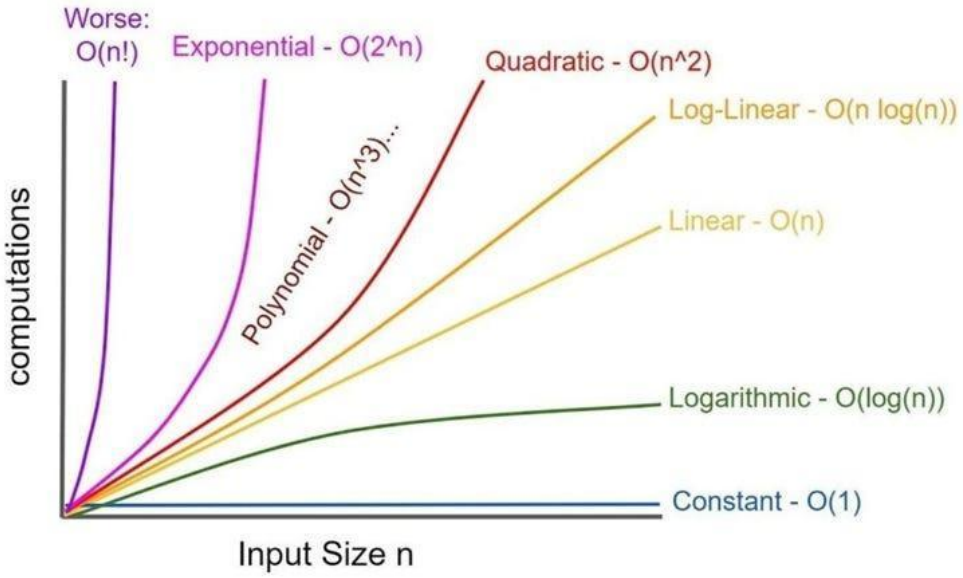
13. Κατηγορίες Πολυπλοκότητας

- Πολυωνυμική:
  – χρόνος ~ n, n^2, n^3

- Εκθετική:
  – χρόνος ~ 2^n

- Παραγοντική:
  – χρόνος ~ n!

- Η διαφορά:
  – είναι καθοριστική για την πρακτική επίλυση

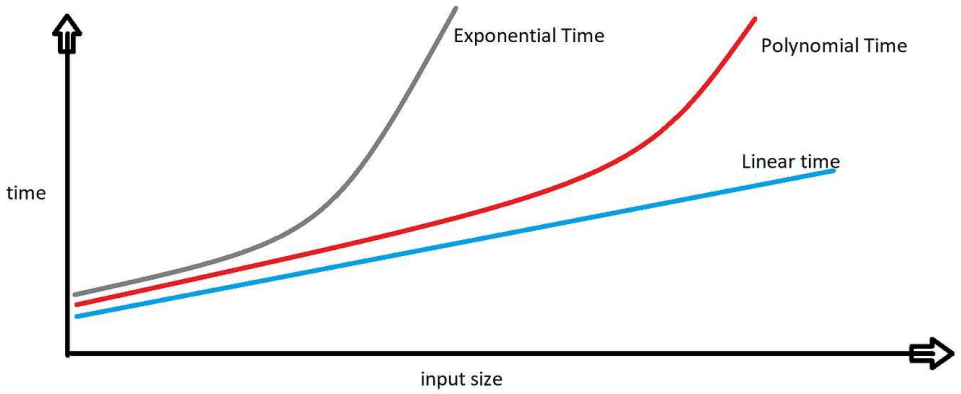


14. Πολυωνυμικά Προβλήματα

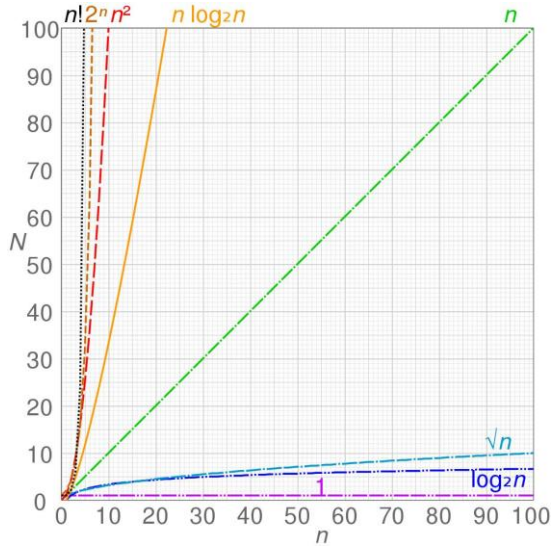
- Ο αριθμός βημάτων:
  – αυξάνεται σχετικά αργά

- Παραδείγματα:
  – συντομότερη διαδρομή
  – ελάχιστο δέντρο

- Η επίλυση:
  – είναι εφικτή για μεγάλα προβλήματα



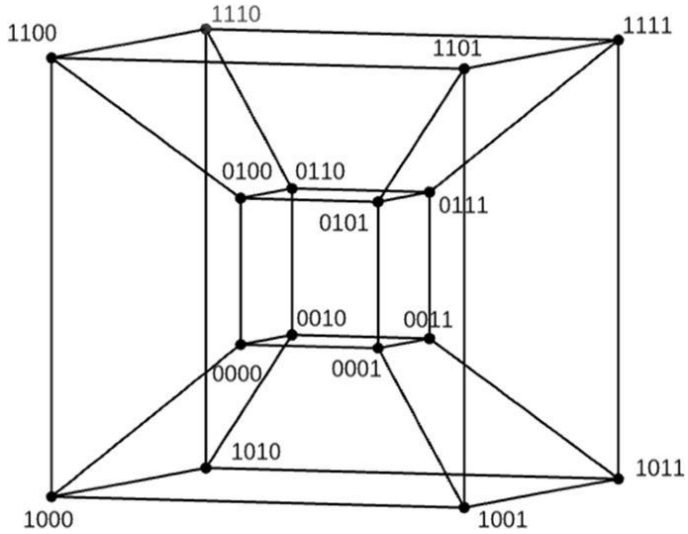
15. Εκθετικά Προβλήματα



- Ο αριθμός βημάτων:
  – αυξάνεται πολύ γρήγορα

- Για μεγάλα n:
  – η επίλυση γίνεται πρακτικά αδύνατη

- Παραδείγματα:
  – επιλογή υποσυνόλων
  – ακέραιος προγραμματισμός



16. Παραγοντική Πολυπλοκότητα

- Ο αριθμός βημάτων:

– αυξάνεται ακόμη πιο γρήγορα

- Αφορά:

– προβλήματα διάταξης

- Η πλήρης απαρίθμηση:

– δεν είναι εφικτή

s =	x	y	z
1st permutation	1	2	3
2nd permutation	1	3	2
3rd permutation	2	1	3
4th permutation	2	3	1
5th permutation	3	1	2
6th permutation	3	2	1

17. Γιατί Δεν Μπορούμε να Εξετάσουμε Όλες τις Λύσεις

- Ο αριθμός λύσεων:

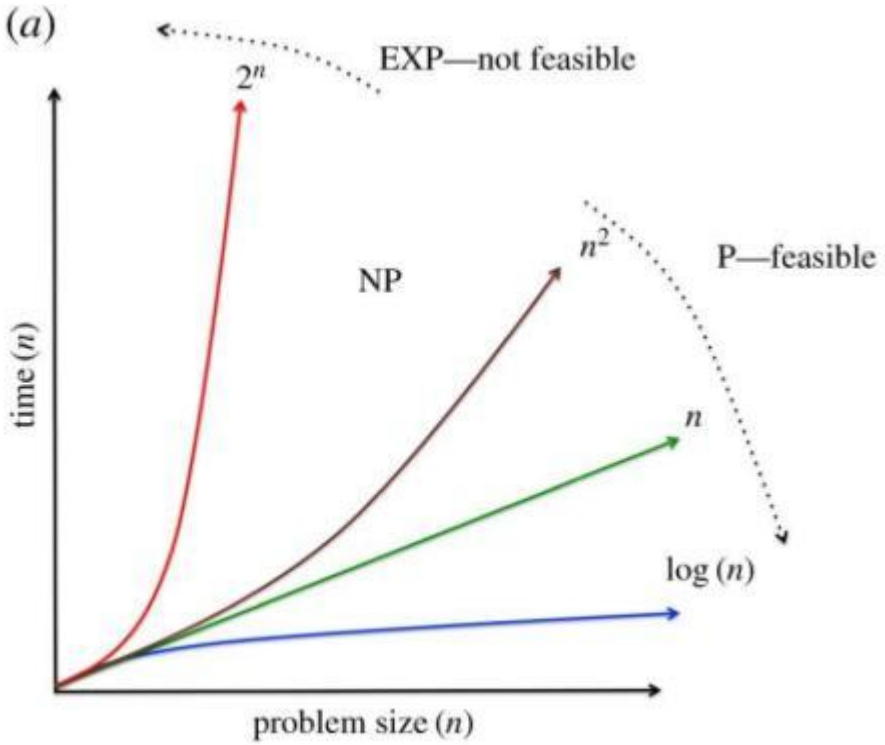
– είναι τεράστιος

- Ακόμη και με ισχυρούς υπολογιστές:

– δεν μπορούμε να τις εξετάσουμε όλες

- Η πλήρης αναζήτηση:

– δεν είναι ρεαλιστική στρατηγική



18. Τι Σημαίνει Υπολογιστική Εφικτότητα

- Ένα πρόβλημα:

– είναι εφικτό αν επιλύεται σε εύλογο χρόνο

- Ο χρόνος:

– εξαρτάται από τον αριθμό των βημάτων

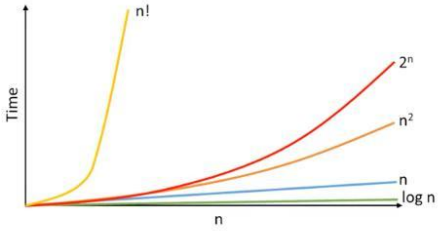
- Πολυωνυμικά προβλήματα:

– θεωρούνται επιλύσιμα

- Εκθετικά προβλήματα:

– συνήθως όχι

Tractable vs intractable problems



Mathematical notation	Name	Tractable/Intractable
n!	Exponential time	Intractable
2^n	Exponential time	Intractable
n^2	Polynomial time	Tractable
n	Linear time	Tractable
log n	Logarithmic time	Tractable

- An **intractable** problem is any problem that *cannot* be solved in **polynomial** time or better.
- That doesn't mean it can't be solved by an **algorithm**, but as soon as the input size increases to anything other than a very small data set, a computer can't solve it within a reasonable timeframe.
- Using **heuristic** methods, many **intractable** problems can be solved by sacrificing the optimal answer for one that is "good enough".
- We are going a little beyond the specification here – you don't need to know about the terms **tractable** and **intractable** for the exam, but we will be discussing algorithm efficiency in later videos.

19. Ρόλος της Δομής

- Η δομή του προβλήματος:

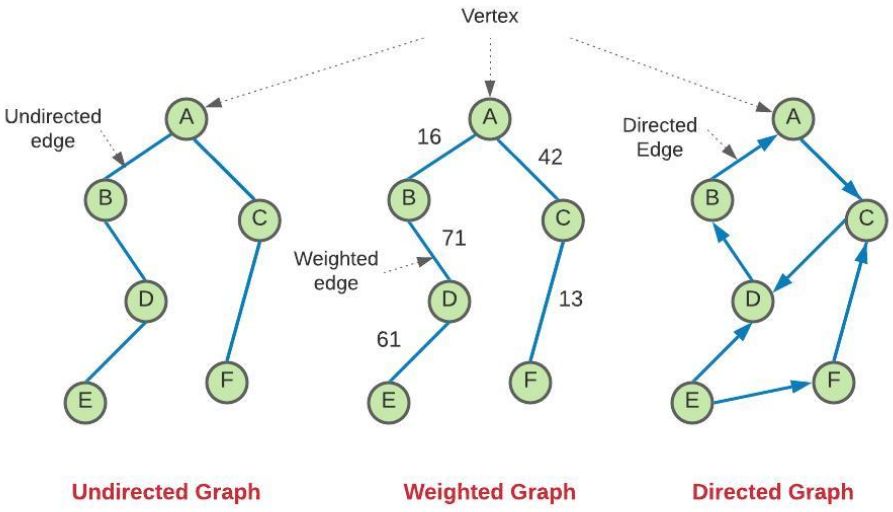
– καθορίζει την πολυπλοκότητα

- Αν υπάρχει εκμεταλλεύσιμη δομή:

– μπορούν να αναπτυχθούν αποδοτικές μέθοδοι

- Αν όχι:

– η επίλυση είναι δύσκολη



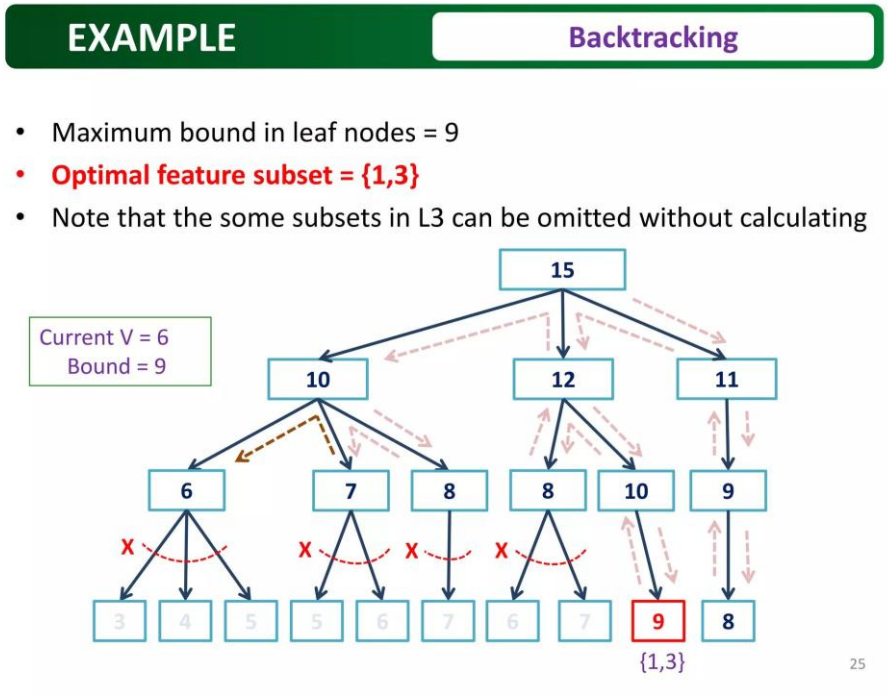
20. Τι Μπορούμε να Κάνουμε σε Δύσκολα Προβλήματα

- Αν το πρόβλημα είναι πολυωνυμικό:
  – εφαρμόζουμε ακριβείς μεθόδους

- Αν είναι εκθετικό:
  – η πλήρης αναζήτηση δεν είναι εφικτή

- Τότε:
  – περιορίζουμε τον χώρο λύσεων
  – εκμεταλλευόμαστε τη δομή
  – χρησιμοποιούμε προσεγγιστικές μεθόδους

- Οι προσεγγίσεις αυτές:
  – θα αναπτυχθούν στις επόμενες διαλέξεις



21. Εννοιολογικό Λάθος 1 – Μικρός Χώρος Λύσεων

- Υποθέτουμε ότι:
  – το πλήθος λύσεων είναι μικρό

- Στην πραγματικότητα:
  – αυξάνεται εκθετικά

- Μικρή αύξηση στο n:
  – οδηγεί σε τεράστια αύξηση

- Το αποτέλεσμα:
  – υποεκτίμηση της δυσκολίας

22. Εννοιολογικό Λάθος 2 – Αρκετή Υπολογιστική Ισχύς

- Πιστεύουμε ότι:
  – ένας ισχυρός υπολογιστής αρκεί

- Όμως:
  – η αύξηση των λύσεων είναι ταχύτερη

- Το πρόβλημα:
  – δεν είναι τεχνολογικό

- Είναι:
  – εγγενές στη δομή

23. Εννοιολογικό Λάθος 3 – Όλα τα Προβλήματα Είναι Ίδια

- Θεωρούμε ότι:
  – όλα τα προβλήματα έχουν παρόμοια δυσκολία

- Στην πραγματικότητα:
  – διαφέρουν σημαντικά

- Η πολυπλοκότητα:
  – εξαρτάται από τη δομή

24. Το Κεντρικό Μήνυμα

- Η συνδυαστική φύση:

– οδηγεί σε εκθετική αύξηση λύσεων

- Η πολυπλοκότητα:

– καθορίζει τι είναι υπολογιστικά εφικτό

- Οι μέθοδοι:

– πρέπει να προσαρμόζονται στη δομή του προβλήματος