

Κριτήρια που πρέπει να ικανοποιεί κάθε αλγόριθμος (~ 2.1)	1. Είσοδος 2. Έξοδος 3. Καθοριστικότητα (Διαίρεση με το Μηδέν) 4. Περαιτότητα (Άπειρος Βρόχος) 5. Αποτελεσματικότητα
Πληροφορική μελετά τους αλγορίθμους από τις ακόλουθες σκοπιές (~ 2.2)	1. Υλικό (Hardware) 2. Γλωσσών Προγραμματισμού 3. Θεωρητική 4. Αναλυτική
Τρόποι Αναπαράστασης Αλγορίθμων (2.3)	1. Ελεύθερο Κείμενο (Free Text) 2. Διάγραμμα Ροής (Flow Chart) 3. Φυσική Γλώσσα κατά βήματα 4. Κωδικοποίηση (Coding)
Ολίσθηση (Shift) προς τα Αριστερά = Πολλαπλασιασμός επί 2 --- Ολίσθηση προς τα Δεξιά = Ακέραια Διαίρεση δια 2	
Πληροφορική μελετά τα Δεδομένα από τις ακόλουθες σκοπιές (~ 3.1)	1. Υλικό (Hardware) 2. Γλωσσών Προγραμματισμού 3. Δομών Δεδομένων 4. Ανάλυση Δεδομένων (Databases)
Βασικές Λειτουργίες (Πράξεις) στις Δομές Δεδομένων (~ 3.2)	1. Προσπέλαση 2. Εισαγωγή νέου κόμβου 3. Διαγραφή κόμβου 4. Αναζήτηση 5. Ταξινόμηση 6. Αντιγραφή 7. Συγχώνευση 8. Διαχωρισμός
Στατικές Δομές Δεδομένων → Συνεχόμενες θέσεις μνήμης – Εκχώρηση κατά την μεταγλώττιση Δυναμικές Δομές Δεδομένων → Τυχαίες θέσεις μνήμης – Δυναμική Παραχώρηση μνήμης	
(~ 3.4) Στοιβά Δεδομένων → Last In First Out (LIFO) → Ώθηση (Push) --- Απώθηση (Pop) (~ 3.5) Ουρά Δεδομένων → First In First Out (FIFO) → Εισαγωγή (Enqueue) --- Εξαγωγή (Dequeue)	
(~6.3) Στοιχεία μίας τεχνητής Γλώσσας : 1. Αλφάβητο 2. Λεξιλόγιο 3. Γραμματική (Τύποι – Συντακτικό) 4. Σημασιολογία	
Τεχνικές Σχεδίασης Προγραμμάτων (~ 6.4 – 6.5)	1. Ιεραρχική σχεδίαση 2. Τμηματικός προγραμματισμός 3. Δομημένος προγραμματισμός 4. Αντικειμενοστραφής Σχεδίαση
Πλεονεκτήματα Δομημένου Προγραμματισμού (~ 6.4)	1. Δημιουργία απλούστερων προγραμμάτων. 2. Άμεση μεταφορά των αλγορίθμων σε προγράμματα. 3. Διευκόλυνση ανάλυσης του προγράμματος σε τμήματα. 4. Περιορισμός των λαθών κατά την ανάπτυξη του προγράμματος. 5. Εύκολη ανάγνωση και κατανόηση του προγράμματος από τρίτους. 6. Ευκολότερη διόρθωση και συντήρηση.
Μεταφραστικά προγράμματα (~ 6.7) →	1. Μεταγλωττιστές (Compilers) – Πιο γρήγορη Εκτέλεση 2. Διερμηνευτές (Interpreters) – Πιο άμεση Διόρθωση
Μειονεκτήματα Χρήσης Πινάκων (~9.2)	1. Δεσμεύουν πολλές θέσεις μνήμης από την αρχή του προγράμματος. 2. Περιορίζουν τις δυνατότητες του προγράμματος (Στατικό μέγεθος μνήμης)
Τυπικές Επεξεργασίες Πινάκων (~9.4)	1. Υπολογισμός αθροίσματος στοιχείων του πίνακα (ανά στήλη, γραμμή). 2. Εύρεση του μέγιστου ή ελάχιστου στοιχείου του πίνακα. 3. Ταξινόμηση των στοιχείων του πίνακα (Φυσαλίδα). 4. Αναζήτηση ενός στοιχείου του πίνακα (Σειριακή – Δυναδική). 5. Συγχώνευση δύο ταξινομημένων πινάκων

Χαρακτηριστικά των Υποπρογραμμάτων (~ 10.2)	1. Κάθε υποπρόγραμμα με μία είσοδο και μία έξοδο. 2. Κάθε υποπρόγραμμα πρέπει να είναι ανεξάρτητο από τα άλλα. 3. Κάθε υποπρόγραμμα όχι πολύ μεγάλο και εκτελεί μία μόνο λειτουργία.
Πλεονεκτήματα του Τμηματικού Προγραμματισμού (~ 10.3)	1. Διευκολύνει την ανάπτυξη του αλγορίθμου και του προγράμματος. 2. Διευκολύνει την κατανόηση και διόρθωση του προγράμματος. 3. Λιγότερος χρόνος και προσπάθεια στη συγγραφή του προγράμματος. 4. Επεκτείνει τις δυνατότητες των γλωσσών προγραμματισμού (βιβλιοθήκες).
Κανόνες Παραμέτρων (~ 10.5.3) Τυπικές (Δήλωση υποπρογράμματος) Πραγματικές (Κλήση υποπρογράμματος)	1. Ο αριθμός πραγματικών και τυπικών παραμέτρων πρέπει να είναι ίδιος. 2. Κάθε πραγματική παράμετρος αντιστοιχεί στην τυπική που βρίσκεται στην αντίστοιχη θέση. 3. Η τυπική παράμετρος και η αντίστοιχη της πραγματική πρέπει να είναι του ιδίου τύπου.
Εμβέλεια Μεταβλητών – Σταθερών (~ 10.6)	1. Απεριόριστη (Καθολικές) 2. Περιορισμένη (Τοπικές) 3. Μερικώς περιορισμένη εμβέλεια
Κατηγορίες Λαθών Προγραμματισμού (~13.1)	1. Συντακτικά Λάθη (κατά την Υλοποίηση). 2. Αντικανονικού Τερματισμού (κατά την Εκτέλεση – Runtime error). 3. Λογικά Λάθη (κακός σχεδιασμός – ανεπιθύμητα αποτελέσματα).
Διπλά Συνδεδεμένες Λίστες <i>Δυναμικές Δομές Σειριακής Προσπέλασης</i>	Πιο εύκολη Αναζήτηση και Ταξινόμηση . / Μεγαλύτερη πολυπλοκότητα στην διαχείριση των κόμβων, παραπάνω μνήμη .
Πλεονεκτήματα (+) & Μειονεκτήματα (-) των Λιστών έναντι των Πινάκων	1. Δυναμικό μέγεθος, 2. Ευκολία εισαγωγής και διαγραφής οπουδήποτε, 3. Μη αναγκαιότητα αρχικής δήλωσης του μεγέθους. 1. Η Τυχαία Πρόσβαση δεν επιτρέπεται (Αδύνατη Δυναμική Αναζήτηση). 2. Μεγαλύτερη επιβάρυνση στην Μνήμη.
Δένδρα Απόφασης → Κόμβος == Χαρακτηριστικό (Ιδιότητα) / Ακμή == Απόφαση (Κανόνας) / Φύλλο == Αποτέλεσμα	
Αντικειμενοστραφής Προγραμματισμός • Αυτόνομες Οντότητες (Αντικείμενα) • Δεδομένα Αντικειμένου (Ιδιότητες) • Ενέργειες, συμπεριφορά (Μέθοδοι)	Αφαιρετικότητα (Abstract) → Κλάσεις (Class) → Αντικείμενα (Objects) Κληρονομικότητα (Υπερκλάση – Υποκλάση – Ιεραρχία) Ενθυλάκωση (Encapsulation) --- Πολυμορφισμός
Εκσφαλμάτωση Λογικών Λαθών στις Δομές Επανάληψης (Περατότητα)	Συνθήκη επανάληψης ή τερματισμού --- Αρχικοποίηση συνθήκης Ενημέρωση συνθήκης εντός της επανάληψης Εντολές που βρίσκονται μέσα στην επανάληψη
Εκσφαλμάτωση Λογικών Λαθών στους Πίνακες (Καθοριστικότητα)	Μέγεθος των πινάκων κατά την δήλωση τους. Δείκτες των πινάκων κατά την προσπέλαση τους. Μη Υπέρβαση των ορίων των πινάκων.
➤ Σενάρια Ελέγχου (test cases) Περιγραφή: Δεδομένα Εισόδου Ανάλυση: Αναμενόμενα Αποτελέσματα	Τεχνική Ελέγχου «Μαύρο Κουτί» → Αγνοείται εντελώς ο Κώδικας ! Τα σενάρια ελέγχου προκύπτουν από τις Προδιαγραφές του προγράμματος. 1. Δημιουργία Ισοδύναμων Διαστημάτων Τιμών. 2. Έλεγχος Ακραίων Τιμών κάθε Διαστήματος.