

Ενότητα 1: Εισαγωγή στον Α/Σ Προγραμματισμό

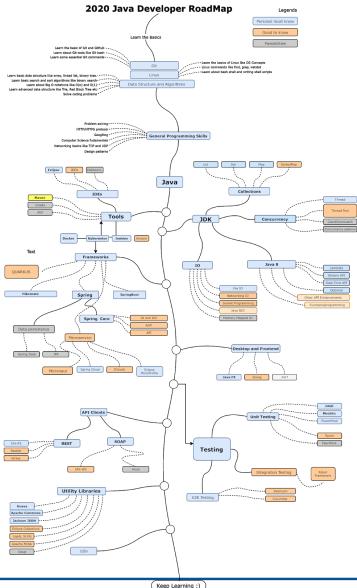
Διάλεξη 5 - 6/6/24

JVM κ. Bytecodes

Αφαίρεση (abstraction)



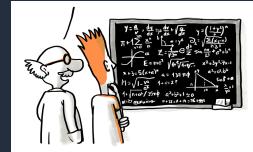
Οδικός χάρτης εκπαίδευσης στη Java



M. Δικαιάκος, ΕΠΛ133

209

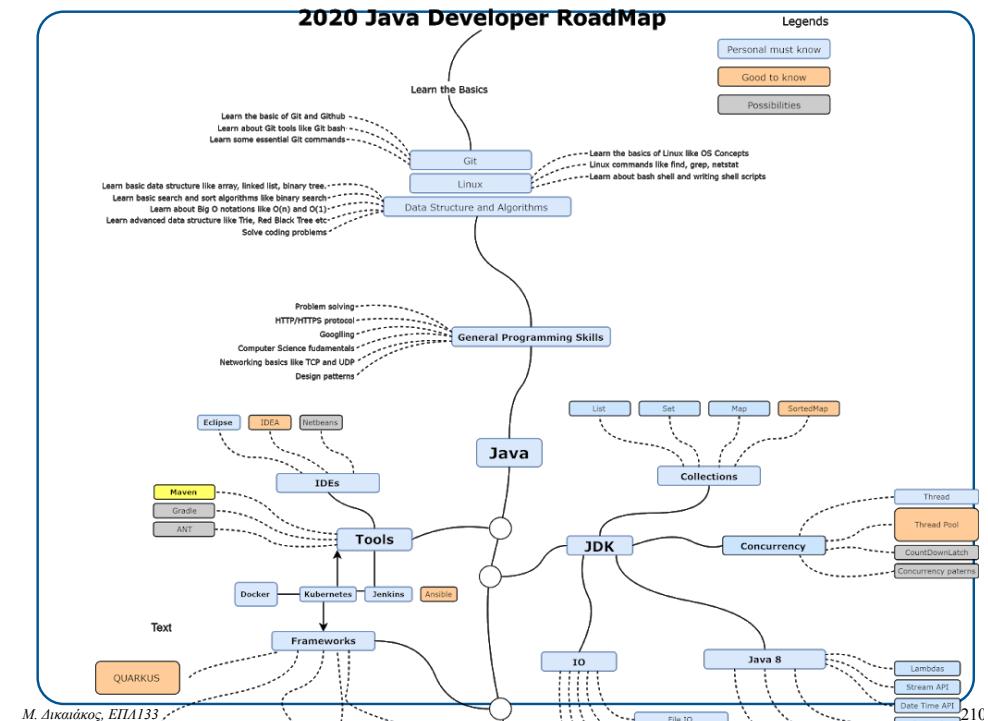
Προηγούμενα

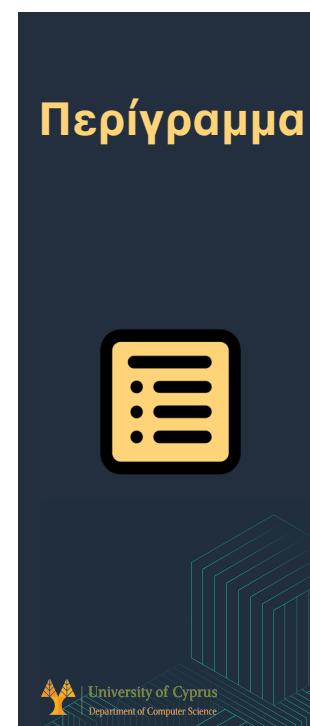
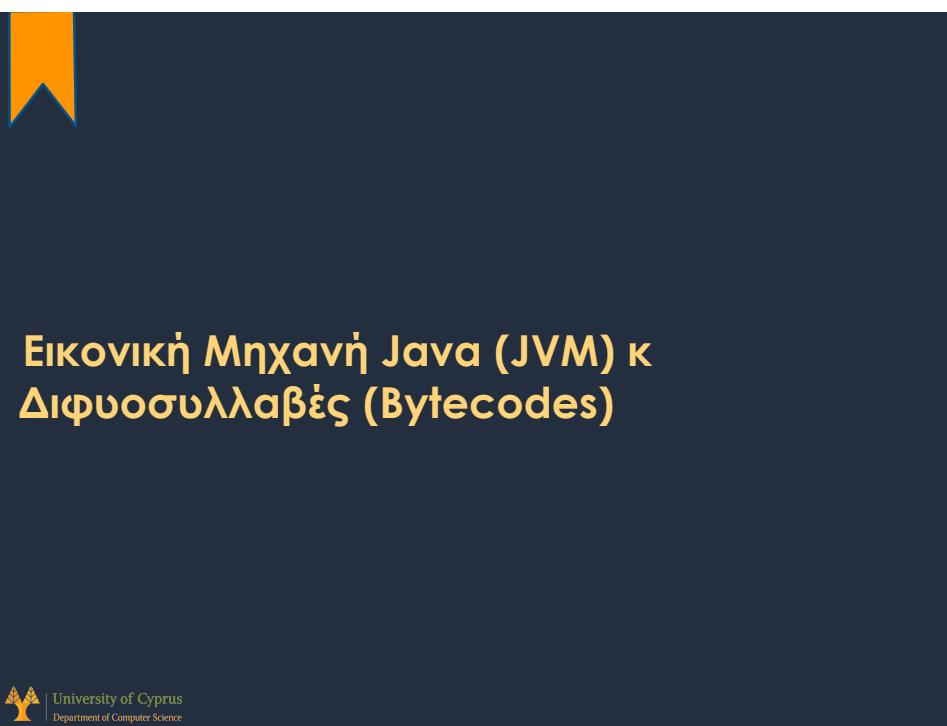
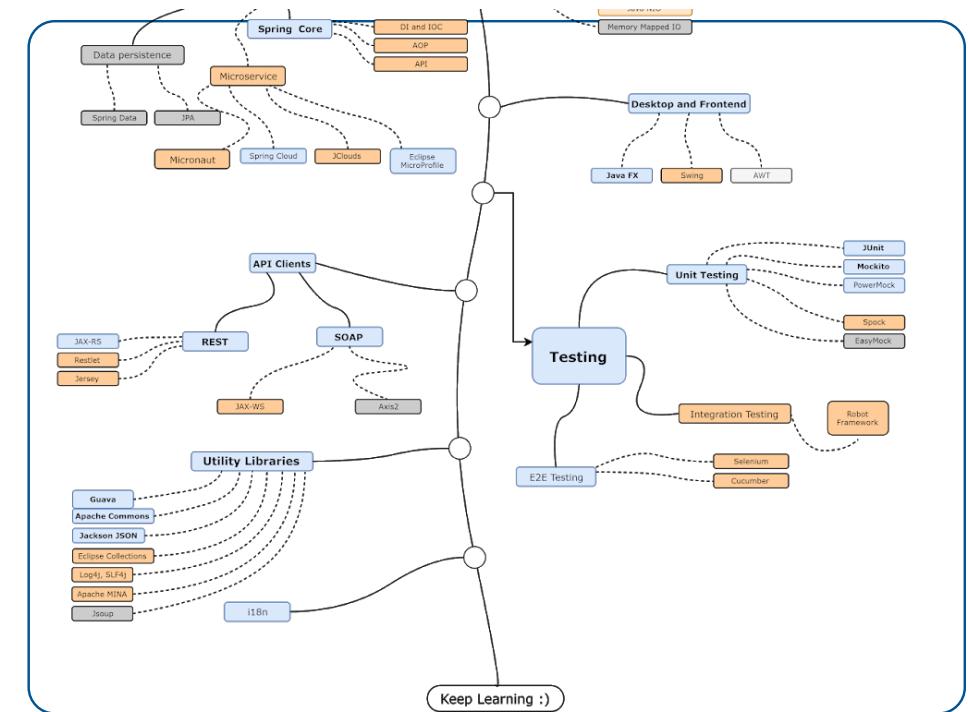
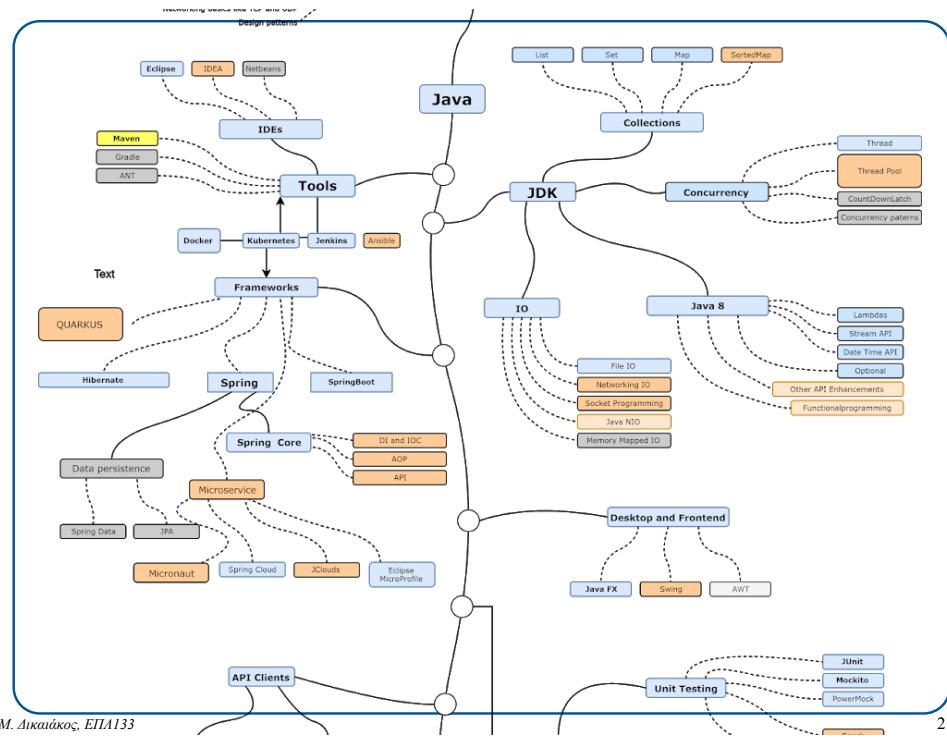


 University of Cyp
Department of Computer Sci
M. Δικαίακος, ΕΠΛ133

- Ιστορία και βασικά χαρακτηριστικά της Java
 - Γιατί η Java;
 - Μεταγγλωτιστές (compilers)
 - Διερμηνείς (interpreters)
 - Νήμα εκτέλεσης (thread)
 - Παραλληλισμός και Πολυνηματική εκτέλεση προγραμμάτων
 - Γλώσσες προγραμματισμού (Programming Languages και Πλατφόρμες/Πλαίσια (Platforms/Frameworks))
 - Η πλατφόρμα της Java
 - Οδικός χάρτης προγραμματιστών/τριάντα Java

M. Δικαιάκος, ΕΠΛ133





- Γιατί η Java;
- Αντικειμενοστρεφείς Γλώσσες Προγραμματισμού
- Ιστορικά στοιχεία Java
- Μεταγλώττιση κ. Διερμηνεία
- Νήμα Εκτέλεσης και Πολυνηματική Εκτέλεση
- Η πλατφόρμα της JAVA
- JVM
- Bytecodes

Εικονικές Μηχανές (virtual machines)

- **Εικονική Μηχανή**: λογισμικό προσομοίωσης υπολογιστικού συστήματος, το οποίο “τρέχει” προγράμματα σαν πραγματικό Υ/Σ.
- **JAVA Virtual Machine (JVM)**: μια εικονική μηχανή που υλοποιεί το σύνολο εντολών ενός Η/Υ με αρχιτεκτονική στοίβας (stack-oriented architecture) και δέχεται προς εκτέλεση μια μορφή ενδιάμεσου κώδικα, τις δυφιοσυλλαβές Java (**Java bytecode**).
 - 1.1 δισ. επιτραπέζιοι Η/Υ τρέχουν JAVA (2013)
 - 930 εκατομύρια JRE καταφορτώσεις / έτος (2013)
 - 3 δισ. κινητά τηλέφωνα τρέχουν JAVA (2013).

Μ. Δικαιάκος, ΕΠΛ133

215

Java Virtual Machine

ORACLE®

- Εικονική Υπολογιστική Μηχανή
- **Ακρογωνιαίος λίθος της πλατφόρμας** και βάση:
 - Της ανεξαρτησίας της Java από Υλικό & Λ.Σ.
 - Του μικρού μεγέθους του μεταγλωττισμένου κώδικα.
 - Της προστασίας των χρηστών από κακόπιστα προγράμματα.

Μ. Δικαιάκος, ΕΠΛ133

217

3 όψεις του JVM

- **Προδιαγραφή** ([specification](#)): ένα έγγραφο που καθορίζει τις τυπικές (formal) προϋποθέσεις τις οποίες πρέπει να καλύπτει μια έγκυρη υλοποίηση του JVM.
 - Η ύπαρξη μιας μοναδικής προδιαγραφής διασφαλίζει ότι όλες οι υλοποιήσεις του JVM είναι **διαλειτουργικές** ([interoperable](#)).
- **Υλοποίηση** ([implementation](#)): Προγραμματιστικός κώδικας ο οποίος ανταποκρίνεται στις προϋποθέσεις της προδιαγραφής JVM.
- **Στιγμιότυπο** ([instance](#)): Μια διεργασία ([process](#)) η οποία «τρέχει» το πρόγραμμα του JVM, και εκτελεί προγράμματα Java μεταγλωττισμένα σε **Java bytecodes**.

Μ. Δικαιάκος, ΕΠΛ133

216

Βασικά χαρακτηριστικά JVM

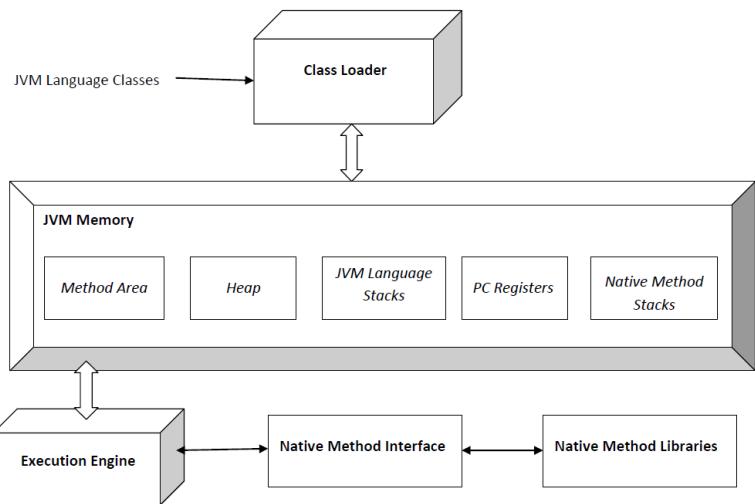
ORACLE®

- Δική της «γλώσσα μηχανής» (instruction set) - τις δυφιοσυλλαφές (byte code).
 - **Εκτελεί διφιοσυλλαβές** για την εκτέλεση προγράμματος Java.
 - Δυφιοσυλλαβές: μεταγλωττισμένη μορφή του κώδικα Java.
- **Διαχείριση Εικονικής Μνήμης**: [κατανέμει δυναμικά μνήμη](#) στα προγράμματα βάση των αναγκών τους και [ανακτά μνήμη](#) που καταλαμβάνεται από αχρηστευμένα αντικείμενα (garbage collection).
- **Ανεξάρτητη από την υλοποίηση**: Δεν προϋποθέτει συγκεκριμένες επιλογές τεχνολογίας υλοποίησης, υλικού Η/Υ ή λειτουργικού συστήματος. Αυτό επιτρέπει στα προγράμματα Java να είναι ανεξάρτητα από υλικό/λειτουργικό, καθώς η JVM κρύβει τις λεπτομέρειες του υποκείμενου συστήματος.
- **Δεν λειτουργεί υποχρεωτικά σαν διερμηνέας** ([not inherently interpreted](#)): Ενώ μπορεί να διερμηνεύσει το bytecode, οι σύγχρονες JVM συνήθως χρησιμοποιούν την τεχνική της **Just-In-Time (JIT)** μεταγλωττισης, που μετατρέπει το bytecode σε τελικό κώδικα ενώρα εκτέλεσης για καλύτερη απόδοση. Επιπλέον, οι εντολές της JVM μπορεί επίσης να εκτελεστούν μέσω μικροκώδικα ή απευθείας σε επεξεργαστή ειδικού σκοπού (Application-Specific Integrated Circuits - ASICs).

Μ. Δικαιάκος, ΕΠΛ133

218

Αρχιτεκτονική JVM



M. Δικαιάκος, ΕΠΛ133

219

Φορτωτής κλάσης - Class Loader

- Αναγνωρίζει και φορτώνει ο, τιδήποτε συμμορφώνεται με τη μορφή αρχείου κλάσης Java.
 - Μια υλοποίηση classloader μπορεί να αναγνωρίζει και άλλους δυαδικούς μορφότυπους (binary formats), αλλά πρέπει υποχρεωτικά να αναγνωρίζει αρχεία κλάσης.
- Εκτελεί 3 βασικές δραστηριότητες με αυτή τη σειρά:
 - **Φόρτωση (loading):** βρίσκει και εισάγει τα δυαδικά δεδομένα για κάθε τύπο (κλάση) του προγράμματος
 - **Σύνδεση (linking):** εκτελεί επαλήθευση, προετοιμασία και (προαιρετικά) επίλυση:
 - **Επαλήθευση (verification):** διασφαλίζει την ορθότητα του εισαγόμενου τύπου
 - **Προετοιμασία (preparation):** δέσμευση και αρχικοποίηση μνήμης για πεδία δεδομένων κάθε κλάσης.
 - **Επίλυση (resolution):** μετατροπή συμβολικών παραπομπών (references) από τον τύπο σε άμεσες αναφορές.
 - **Αρχικοποίηση (initialization):** καλεί κώδικα Java ο οποίος αρχικοποιεί τις μεταβλητές κλάσης στις σωστές αρχικές τους τιμές.

M. Δικαιάκος, ΕΠΛ133

220

Περίγραμμα



- Γιατί η Java;
- Αντικειμενοστρεφείς Γλώσσες Προγραμματισμού
- Ιστορικά στοιχεία Java
- Μεταγλώττιση κ. Διερμηνεία
- Νήμα Εκτέλεσης και Πολυνηματική Εκτέλεση
- Η πλατφόρμα της JAVA
- JVM
- Bytecodes

Δυφιοσυλλαβές (bytecodes)

ORACLE®

- Είναι ο μεταγλωττισμένος κώδικας Java, ο οποίος θα εκτελεστεί από το JVM.
- Αναπαρίσταται σε δυαδική μορφή (binary format), ανεξάρτητη υλικού και ΛΣ, η οποία αποθηκεύεται συνήθως (όχι αποκλειστικά) σε αρχείο με συγκεκριμένη μορφή, το **class file format**.
- Το **class file format** ορίζει επακριβώς την αναπαράσταση των κλάσεων και των διαπροσωπειών Java.

```
13: iload_3
14: ifeq    26
17: iload   5
19: iconst_1
20: iadd
21: istore  4
23: goto   30
26: iload   6
28: istore  4
30: return
```

JVM και bytecodes

ORACLE®

- Ο JVM δεν γνωρίζει τίποτα για τη Java, τη σύνταξη και την σημασιολογία της!
 - Γνωρίζει μόνο ένα συγκεκριμένο μορφότυπο, αυτών των αρχείων class (class file format).
- Ένα class file περιλαμβάνει εντολές JVM (bytecodes - διφυοσυλλαβές), πίνακα συμβόλων (symbol table), και άλλες δευτερεύουσες πληροφορίες.
- Οποιαδήποτε γλώσσα προγραμματισμού με λειτουργικότητα που μπορεί να εκφραστεί/αποτυπωθεί σε ένα έγκυρο class file, μπορεί να εκτελεστεί από τον JVM.
- Για λόγους ασφάλειας, ο JVM επιβάλλει αυστηρούς συντακτικούς και δομικούς περιορισμούς στις εντολές ενός class file.

M. Δικαιάκος, ΕΠΛ133

223

Μοντέλο εκτέλεσης JAVA

Η JAVA συνδυάζει μεταγλωττιση και της διερμηνεία:

- Πηγαίος κώδικας μεταγλωττίζεται σε JVM bytecode.
- Τα bytecode μπορούν να εκτελεσθούν άμεσα από τον διερμηνέα της JVM.
- Ο διερμηνέας παρακολουθεί επίσης πόσο συχνά εκτελείται το κάθε τμήμα bytecode (run-time profiling) και προωθεί συχνά εκτελούμενο κώδικα (hot spots) στον «έγκαιρο» μεταγλωττιστή (just-in-time - JIT) compiler.
- Ο JIT μετατρέπει το bytecode σε τελικό κώδικα, και τον τρέχει.

M. Δικαιάκος, ΕΠΛ133

224

Μοντέλο εκτέλεσης JAVA

- Ο JIT μπορεί να κάνει βελτιστοποιήσεις στον κώδικα που θα μεταφράσει, αξιοποιώντας τις πληροφορίες που συγκεντρώνονται από την παρακολούθηση (monitoring) του τρεχούμενου κώδικα.
 - Δυνατότητα καλύτερης βελτιστοποίησης της επίδοσης από παραδοσιακούς μεταγλωττιστές (γιατί:)
- Η αρχιτεκτονική της JAVA επιτρέπει:
 - την εκτέλεση κώδικα σε οποιαδήποτε μηχανή στην οποία υπάρχει η JVM και
 - την γρήγορη εκτέλεση σε οποιαδήποτε μηχανή υπάρχει και ο μεταγλωττιστής JIT
 - Σοβαρή βελτιστοποίηση γίνεται για τα hot spots του κώδικα.

M. Δικαιάκος, ΕΠΛ133

225

Μοντέλο εκτέλεσης JAVA

Java source code

Java compiler

JVM bytecode

JVM interpreter

JIT compiler

machine code

M. Δικαιάκος, ΕΠΛ133

226

JVM πέρα από τη JAVA

Versions of non-JVM languages		Languages designed expressly for JVM
Language	On JVM	Language
Erlang	Erjang	Java
JavaScript	Rhino	BBj
Pascal	Free Pascal	BeanShell
PHP	Quercus	Ceylon
Python	Jython	Clojure
REXX	NetRexx ^[5]	Fantom
Ruby	JRuby	Kotlin
Scheme	Kawa	Groovy
Tcl	Jacl	MIDletPascal
		Scala
		Xtend

Μ. Δικαιάκος, ΕΠΛ133

227

Σε ποιό στοιχείο της πλατφόρμας της JAVA στηρίζεται η ιδιότητά της «write once, run everywhere»

- Στη χρήση του διερμηνέα javac σε συνδυασμό με τον μεταγλωττιστή JIT
- Στη χρήση της εικονικής μηχανής JAVA (JVM)
- Στην αξιοποίηση ολοκληρωμένων περιβαλλόντων ανάπτυξης όπως το Eclipse.
- Στην αντικειμενοστρέφεια της JAVA



ΕΠΛ233

229

Πλεονεκτήματα / Μειονεκτήματα JVM

■ Πλεονεκτήματα:

- Ασφάλεια
- Εύκολος προγραμματισμός δικτύων (Network-centric programming)
- Ανεξαρτησία αρχιτεκτονικής

■ Μειονεκτήματα:

- Επίδοση(performance) - όταν γίνεται χρήση διερμηνέα, η εκτέλεση προγραμμάτων είναι πιο αργή από την εκτέλεση ιθαγενή κώδικα (native code)
- Δυσκολία άμεσης διαχείρισης μνήμης από τον προγραμματιστή

Μ. Δικαιάκος, ΕΠΛ133

228

Αφαίρεση (abstraction)

Αφαίρεση



Ο ανθρώπινος νου, όταν επεξεργάζεται απλές ιδέες, πραγματοποιεί τρεις βασικές λειτουργίες:

- **Συνδυασμό** και **Σύνθεση απλών ιδεών**: έτσι δημιουργούνται οι **σύνθετες ιδέες**.
- **Αντιπαραβολή δύο ιδεών**, απλών ή σύνθετων, χωρίς να γίνεται σύνθεση τους: έτσι προκύπτουν οι **ιδέες για τις σχέσεις** ανάμεσα σε ιδέες.
- **Διαχωρισμό** κάποιων ιδεών από όλες τις υπόλοιπες που τις συνοδεύουν στην πραγματική τους ύπαρξη: αυτό ονομάζεται **αφαίρεση**, και έτσι δημιουργούνται όλες οι γενικές ιδέες.

John Locke, An Essay Concerning Human Understanding (1690)

231

Αφαίρεση (abstraction)

- «(ΦΙΛΟΣΟΦΙΑ): **νοητική διαδικασία** κατά την οποία **απομονώνονται** τα **κοινά στοιχεία** συνόλου **ομοειδών πραγμάτων**, πράγμα που επιτρέπει τον **καθορισμό** της **γενικής έννοιας** στην οποία υπάγονται αυτά.»
- «(ΦΙΛΟΣΟΦΙΑ): διάκριση του **ουσιώδους** από το **επουσιώδες**, κατανόηση **γενικών**, αφηρημένων **έννοιών**»
- »(ΤΕΧΝΗ): η απομάκρυνση του καλλιτέχνη από τη ρεαλιστική απεικόνιση της πραγματικότητας και η απόδοση του αισθητικού αντικειμένου με τρόπο αφηρημένο που βασίζεται στην προβολή αυθύπαρκτων μορφών, χρωμάτων κλπ.»



M. Δικαιάκος, ΕΠΛ133

232

Αφαιρετικός (abstract)

- **Έννοιολογική επεξεργασία** από την οποία προκύπτουν γενικοί κανόνες και έννοιες
 - Η επεξεργασία αφορά σε **χρήση** και **ταξινόμηση παραδειγμάτων, σημαίνοντων, πρωταρχικών εννοιών, συστημάτων ή λειτουργιών/μεθόδων.**
- «**Αφαίρεση** / **αφαιρετική έννοια** αποκαλείται επίσης το αποτέλεσμα της επεξεργασίας αυτής:
 - **Συνοψίζει** τα υπό περιγραφή παραδείγματα, σημαίνοντα, συστήματα, φαινόμενα, λειτουργίες/ μεθόδους.
- Πηγή: Wikipedia (<https://en.wikipedia.org/wiki/Abstraction>)



M. Δικαιάκος, ΕΠΛ133

234

Αφαιρετικότητα (abstraction)

- Η ιδιότητα του αφαιρετικού - επιγραμματικότητα
- Απόδοση πολλών νοημάτων με λίγα λόγια
- Εκδήλωση παραμερισμού στοιχείων του περιβάλλοντος ή της νοήσεως

Μ. Δικαιάκος, ΕΠΛ133

235

Αφαίρεση - πώς πραγματοποιείται;

- Η διαδικασία της αφαίρεσης συνήθως εμπεριέχει το φίλτράρισμα του πληροφοριακού περιεχομένου μιας έννοιας, ενός συνόλου παραδειγμάτων, παρατηρούμενων φαινομένων, σημαινόντων, συστημάτων, λειτουργιών με στόχο:
 - την **απάλεψη** πληροφοριών που **δεν ενδιαφέρουν** ή **δεν είναι σημαντικές**
 - την **επιλογή** και **συμπερίληψη** εκείνων των στοιχείων / πληροφοριών που **είναι χρήσιμες** / **σχετικές** για κάποιο συγκεκριμένο σκοπό

Μ. Δικαιάκος, ΕΠΛ133

236

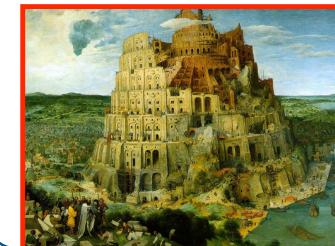
Αφαίρεση: πού χρησιμοποιείται;

- Τέχνες
- Πληροφορική
- Μηχανική
- Γλωσσολογία
- Μαθηματικά
- Μουσική
- Νευρολογία
- Φιλοσοφία
- Ψυχολογία

Μ. Δικαιάκος, ΕΠΛ133

237

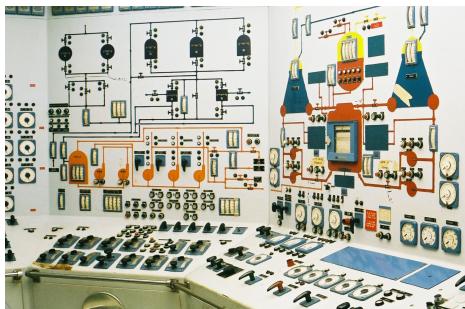
Ποιό από τα πιο κάτω αποτελεί «αφηρημένη τέχνη» (abstract art);



ΕΠΛ133

238

Αφαιρετικότητα και πραγματικός κόσμος



M. Δικαιάκος, ΕΠΛ133

239

Αφαίρεση στις Γλώσσες Προγραμματισμού

- Οι υπολογιστικές διεργασίες (computational processes) είναι αφαιρετικές οντότητες/ υπολογιστικές δομές (abstract beings) που «υπάρχουν» και «εξελίσσονται» στους Η/Υ.
- Κατά την εξέλιξη τους, οι διεργασίες επεξεργάζονται άλλες αφαιρετικές οντότητες/δομές, που ονομάζονται **δεδομένα**.
- Η εξέλιξη μιας διεργασίας κατευθύνεται από μια **σειρά κανόνων** που ονομάζεται **πρόγραμμα**.
- Οι άνθρωποι δημιουργούν **προγράμματα** για να κατευθύνουν τις υπολογιστικές διεργασίες.

Structure and Interpretation of Computer Programs



SECOND EDITION
Unofficial Textbook Format 2nd edition
Harold Abelson and
Gerald Jay Sussman
with Julie Sussman
foreword by Alan J. Perlis

M. Δικαιάκος, ΕΠΛ133

Αφαίρεση

Αφαίρεση και Πληροφορική

 University of Cyprus
Department of Computer Science

Αφαίρεση στις Γ/Π

- Μια ισχυρή Γ/Π είναι κάτι περισσότερο από ένα μέσο για την παροχή εντολών προς εκτέλεση σε Η/Υ.
- Η Γ/Π χρησιμεύει ως **πλαίσιο** οργάνωσης **στοιχείων** (elements) για τον καθορισμό των υπολογιστικών διεργασιών με τις οποίες θα **επιλυθούν** σύνθετα προβλήματα.
- Έτσι, όταν περιγράφουμε/μαθαίνουμε μια Γ/Π, δίνουμε ιδιαίτερη προσοχή στα **μέσα** που παρέχει η γλώσσα για το συνδυασμό **απλών στοιχείων** σε άλλα πιο σύνθετα.
- Κάθε ισχυρή γλώσσα Γ/Π έχει τρεις μηχανισμούς για να το πετύχει αυτό:
 - **Αρχέγονες εκφράσεις** (primitive expressions), οι οποίες αντιπροσωπεύουν τα απλούστερα στοιχεία με τα οποία ασχολείται η γλώσσα.
 - **Μηχανισμούς σύνθεσης** (means of combination - τελεστές, εκφράσεις), με τους οποίους δομούνται σύνθετα στοιχεία από άλλα απλούστερα, και
 - **Μηχανισμούς Αφαίρεσης** (means of abstraction), με τους οποίους μπορούμε να ονομάσουμε/ταυτοποιήσουμε σύνθετα στοιχεία ώστε να τα χειριζόμαστε ως μονάδες.

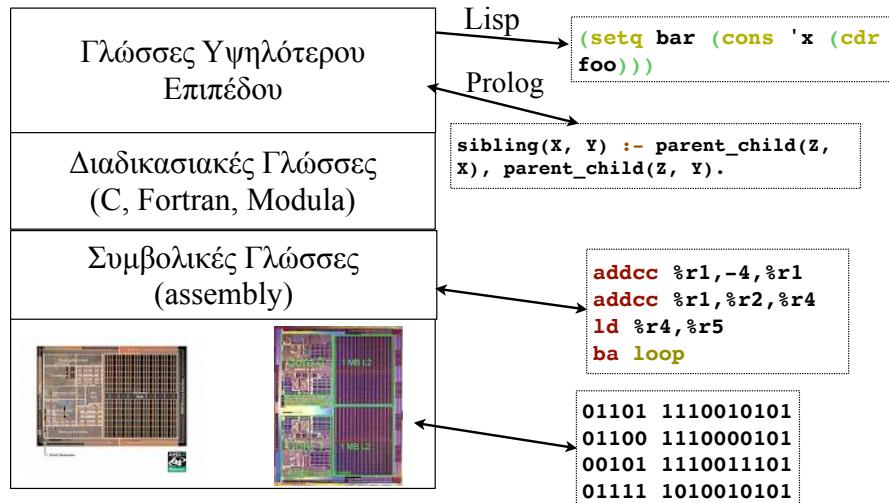
Structure and Interpretation of Computer Programs, Abelson and Gerald Jay Sussman with Julie Sussman, MIT Press, 1996 242

Αφαίρεση στις Γ/Π

- Στον προγραμματισμό ασχολούμαστε με δύο είδη στοιχείων:
 - Δεδομένα (data): οι πληροφορίες που θέλουμε να επεξεργαστούμε και να μετασχηματίσουμε
 - Διαδικασίες (procedures): περιγραφές των κανόνων για την επεξεργασία των δεδομένων.
- Έτσι, οποιαδήποτε ισχυρή Γ/Π θα πρέπει να μπορεί να περιγράφει αρχέγονα δεδομένα και αρχέγονες διαδικασίες και θα πρέπει να έχει τρόπους για τη σύνθεση και την αφαίρεση διαδικασιών και δεδομένων.

Structure and Interpretation of Computer Programs, Abelson and Gerald Jay Sussman with Julie Sussman, MIT Press, 1996

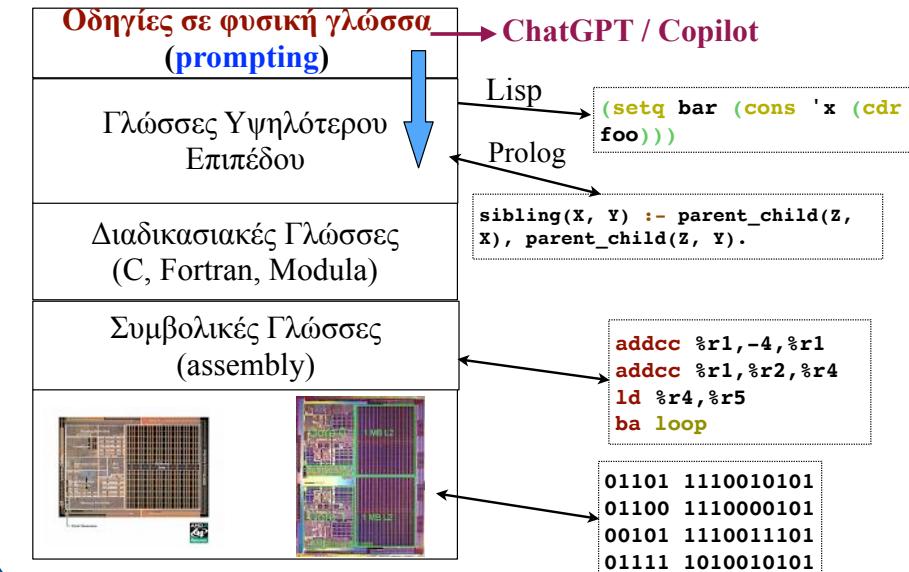
Αφαιρετικότητα στον προγραμματισμό Η/Υ



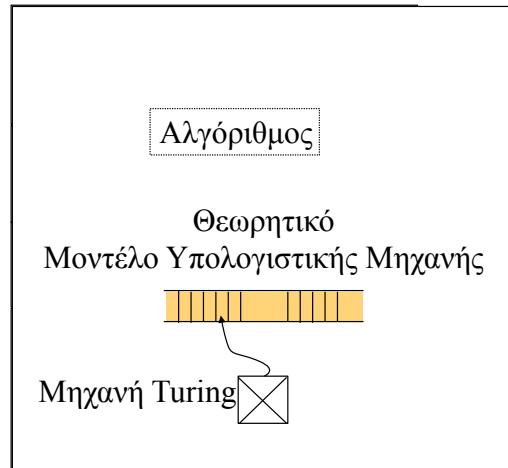
Αφαιρετικότητα στον προγραμματισμό Η/Υ

- Διαδικασιακές Γ.Π. (procedural P.L.: Fortran, BASIC, C)
 - Αποτελούν αφαιρετική αναπαράσταση των δυνατοτήτων των συμβολικών γλωσσών (assembly), οι οποίες με την σειρά τους αποτελούν μια αφαιρετική αναπαράσταση των λειτουργιών του υλικού του Η/Υ.
 - Ο προγραμματιστής καλείται να δημιουργήσει μια σύνδεση ανάμεσα στο μοντέλο μηχανής και στο μοντέλο του προβλήματος που προσπαθεί να επιλύσει.
- LISP, APL, PROLOG: επιβάλλουν συγκεκριμένη μοντελοποίηση του πραγματικού κόσμου (π.χ. σαν σύνολο από λίστες, σαν αλγορίθμους, ή σαν αλυσίδες αποφάσεων).

Αφαιρετικότητα στον προγραμματισμό Η/Υ



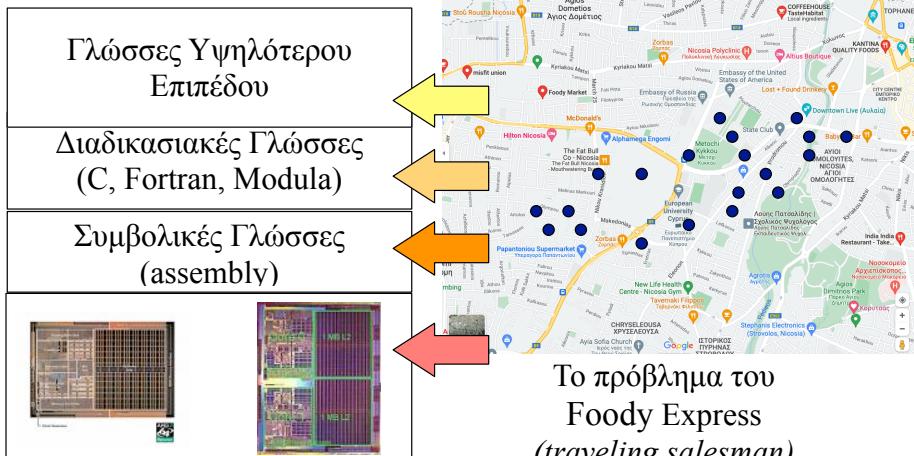
Αφαιρετικότητα στον προγραμματισμό Η/Υ



Μ. Δικαίακος, ΕΠΛ133

247

Πλεονεκτήματα αφαιρετικότητας



Μ. Δικαίακος, ΕΠΛ133

249

Πλεονεκτήματα αφαιρετικότητας

Ποια διαδρομή είναι η συντομότερη;

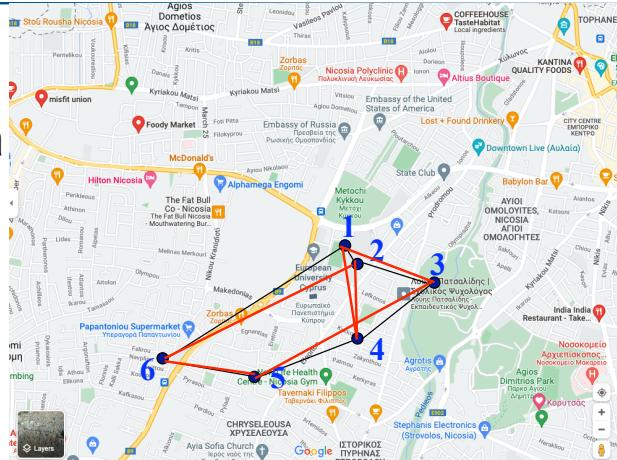
1, 2, 3, 4, 5, 6, 1

ή

1, 4, 2, 6, 5, 3, 1

ή

....



Το πρόβλημα του Foody Express: βρες την πιο σύντομη διαδρομή
(traveling salesman)

Μ. Δικαίακος, ΕΠΛ133

248

Αφαιρετικότητα δεδομένων

Data Abstraction

- Διαφοροποιεί:
 - τις βασικές ιδιότητες δομών δεδομένων από τις λεπτομέρειες της υλοποίησής τους
 - το τι μπορούμε να πράξουμε με τα δεδομένα από το πώς έχουν αυτά αναπαρασταθεί/κωδικοποιηθεί στον Η/Υ
- Π.χ. μπορούμε να αναπαραστήσουμε αριθμούς με πολλούς τρόπους:
 - Ελληνική αρίθμηση: α, β, γ, δ, ε, στ, ζ, η, θ, ι
 - Αραβική αρίθμηση: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
 - Λατινική αρίθμηση: I, II, III, IV, V, VI, VII, VIII, IX, X
 - Κινεζική αρίθμηση: ○, 一, 二, 三, 四, 五, 六, 七, 八, 九, 十
 - Ψηφιακή αρίθμηση: 0, 1
- Πράξεις πάνω στους αριθμούς ακολουθούν τους ίδιους κανόνες ανεξάρτητα του πώς αναπαριστώνται οι αριθμοί.

Μ. Δικαίακος, ΕΠΛ133

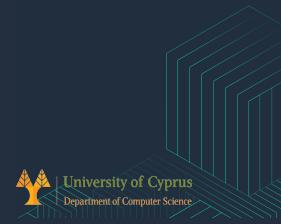
250

Πλεονεκτήματα data abstraction

- Αποκρύπτει από τους προγραμματιστές ή τους χρήστες των δομών δεδομένων επουσιώδεις λεπτομέρειες (που δεν είναι ιδιαίτερα σημαντικές για την εργασία τους).
 - Για την ανάπτυξη των περισσοτέρων εφαρμογών συνήθως μας ενδιαφέρει πώς και γιατί μπορούμε να χρησιμοποιήσουμε κάποιες δομές δεδομένων και όχι πώς αυτές υλοποιούνται ή τι εσωτερική αναπαράσταση έχουν.
- Διευκολύνει το μετασχηματισμός ενό σύνθετου προβλήματος σε σύνολο απλούστερων υποπροβλημάτων (**διαίρει και βασίλευε**).
- Διευκολύνει την **αλλαγή υλοποίησης** μιας λειτουργίας χωρίς να επηρεάζεται η συνολική λειτουργικότητα ενός συστήματος.

M. Δικαιάκος, ΕΠΛ133

251



Αφαίρεση

Αφαίρεση στην Αντικειμενοστρέφεια

«Ο σκοπός της αφαιρετικής διαδικασίας **δεν είναι η ασάφεια** αλλά να δημιουργήσει ένα **νέο σημασιολογικό επίπεδο** στο οποίο μπορεί κανείς να είναι **απολύτως ακριβής**»

"The purpose of abstraction is not to be vague, but to create a new semantic level in which one can be absolutely precise."

Edsger Dijkstra

M. Δικαιάκος, ΕΠΛ133

Αντικείμενα στον πραγματικό κόσμο

- Τα αντικείμενα του φυσικού κόσμου διαθέτουν:
 - Ιδιότητες (properties)
 - Διακριτές Καταστάσεις (distinct states)
 - Συμπεριφορές (behaviors)





Σε πόσες «καταστάσεις» (states) υπορεί να βρεθεί ο συγκεκριμένος λαμπτήρας; ٣٣ · ٢٠

00:20

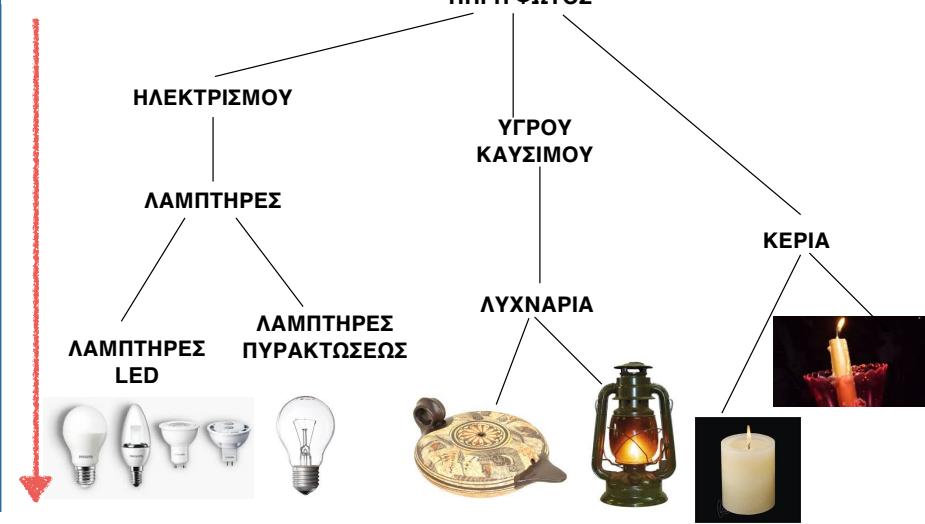
- **Μία**
 - **Δύο**
 - **Τρεις**
 - **Εξαρτάται**

EPA133

255

– ΙΕΡΑΡΧΙΕΣ, ΑΦΑΙΡΕΤΙΚΟΤΗΤΑ και ΔΕΝΔΡΑ

Πιο GENIKO



Matlab EE4132

Ομαδοποίηση και ιεραρχικότητα

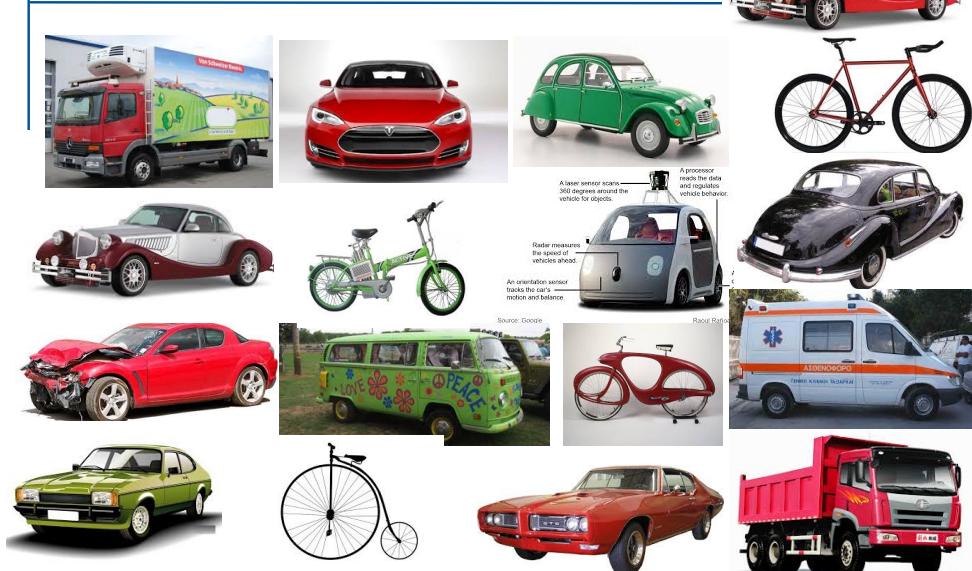
- Τα αντικείμενα του φυσικού κόσμου διαθέτουν:
 - Ιδιότητες (properties)
 - Διακριτές Καταστάσεις (distinct states)
 - Συμπεριφορές (behaviors)



M. Δικαιάκος, ΕΠΛ133

256

Αντικείμενα και κλάσεις στον πραγματικό κόσμο



M. Δικαιάκος, ΕΠΛ133

258



259

Αυτοκίνητα: αντικείμενα και κλάσεις

■ Αντικείμενα: Αυτοκίνητα:

- Διακριτές οντότητες
- Έχουν ιδιότητες (fields): χρώμα, αριθμός επιβατών, έτος κατασκευής, ισχύς μηχανής κλπ.
- Έχουν συμπεριφορές, όπως «κίνηση εμπρός», «κίνηση πίσω», «στροφή» κλπ (μέθοδοι)
- Παρουσιάζουν κοινά χαρακτηριστικά

■ Κλάσεις: Κατηγορία «Αυτοκίνητο»

- **Έννοια** που αντιπροσωπεύει όλα εκείνα τα αντικείμενα που διαθέτουν μηχανή, 4 (;) ρόδες, τιμόνι (;) και μπορούν να μετακινηθούν από μέρος σε μέρος.

M. Δικαιάκος, ΕΠΛ133

260

Αντικείμενα και Κλάσεις

- **Κλάση/κατηγορία (class)** «οικογένεια» αντικειμένων με κοινό σύνολο ιδιοτήτων και συμπεριφορών.
- Αντικείμενα (objects): «**στιγμιότυπα**» (instances) μιας κλάσης.
- Κλάσεις: «**σχεδιότυπα**» κάποιων παρόμοιων αντικειμένων.

261

Ιδιότητες και Συμπεριφορές

- Ιδιότητες-Πεδία Δεδομένων (Properties or fields)
 - Μεταβλητές που ανήκουν/χαρακτηρίζουν/ συναποτελούν ένα αντικείμενο
 - Π.χ.: ιδιοκτήτης ενός αυτοκινήτου, χρώμα, τύπος και μοντέλο ενός κινητού τηλεφώνου, μηχανή
- Συμπεριφορές - Μέθοδοι (methods)
 - **Λειτουργίες** που μπορεί να πραγματοποιηθούν σε ένα αντικείμενο ή διαμέσου αυτού
 - Π.χ.:
 - μετακίνησης εμπρός ένα αυτοκίνητο, μετακίνησης πίσω, στρίψε
 - κάνε ένα τηλεφώνημα, απάντηση σε εισερχόμενο τηλεφώνημα, παίξε μουσική, στείλε μήνυμα

M. Δικαιάκος, ΕΠΛ133

262

Ποιά η σχέση ιδιοτήτων και καταστάσεων ενός αντικειμένου;

00 : 80

- Οι ιδιότητες του αντικειμένου είναι υποσύνολο των καταστάσεων στις οποίες μπορεί να βρεθεί το αντικείμενο
- Οι καταστάσεις είναι υποσύνολο των ιδιοτήτων του αντικειμένου
- Δεν υπάρχει σχέση μεταξύ ιδιοτήτων και καταστάσεων
- Μια κατάσταση αντιστοιχεί σε συνδυασμούς τιμών που μπορεί να λάβουν οι ιδιότητες του αντικειμένου

ΕΠΑ/133

263

Ποιά από τις παρακάτω προτάσεις είναι σωστή;

00 : 40

00 : 20

- Ένα αντικείμενο μπορεί να έχει ως ιδιότητα ένα άλλο αντικείμενο
- Η τιμή μιας ιδιότητας αντικειμένου μπορεί να είναι άλλο αντικείμενο
- Ένα αντικείμενο αποτελείται μόνο από ιδιότητες ή μόνο από συμπεριφορές
- Τίποτε από τα πιο πάνω

ΕΠΑ/133

265

Ποιά από τις παρακάτω προτάσεις είναι σωστή;

00 : 40

00 : 20

- Η κλάση είναι ένα σύνολο αντικειμένων με κοινές ιδιότητες και συμπεριφορές
- Η κλάση είναι ένα σχεδιότυπο απαράλλακτων αντικειμένων
- Το αντικείμενο είναι σχεδιότυπο· η κλάση είναι στιγμιότυπο
- Καμιά από τις πιο πάνω

ΕΠΑ/133

264

Ποιά από τις παρακάτω προτάσεις είναι σωστή;



00 : 40 00 : 20

- Ιδιότητα του αυτοκινήτου που βλέπετε είναι το «χρώμα πράσινο».
- Ιδιότητα του αυτοκινήτου που βλέπετε είναι το «χρώμα»
- Ιδιότητα του αυτοκινήτου που βλέπετε είναι: «μάρκα Citroen 2CV»
- Καμιά από τις πιο πάνω

ΕΠΑ/133

266

Αντικείμενα και Κλάσεις στις Γ.Π.

- Βασικά συστατικά στοιχεία των προγραμμάτων Α/Σ Γ.Π. είναι οι **κλάσεις**: μια μορφή «τύπων», όπως τα struct της C, ενισχυμένων όμως με λειτουργικότητες («συμπεριφορές»), κώδικα, έλεγχο πρόσβασης σε πεδία δεδομένων κλπ.
- Οι κλάσεις, χρησιμοποιούνται κυρίως για να κατασκευάζουμε **αντικείμενα**, τα οποία αποτελούν και τη βασική δομή δεδομένων του Α/Σ Προγραμματισμού.
 - Κατά την εκτέλεση του, ένα αντικειμενοστρεφές προγράμμα μπορεί να ιδωθεί σαν ένα σύνολο αντικειμένων που αλληλεπιδρούν μεταξύ τους εξυπηρετώντας το ένα το άλλο.

Μ. Δικαιάκος, ΕΠΛ133

267

Κλάσεις

- Στον πραγματικό κόσμο υπάρχουν αντικείμενα που ανήκουν στην ίδια **κατηγορία** - είναι του ίδιου **τύπου**.
 - Π.χ. έχουμε χιλιάδες ποδήλατα της ίδιας μάρκας και του ίδιου μοντέλου, τα οποία έχουν κατασκευαστεί με βάση το ίδιο σχέδιο κι έχουν τα ίδια συστατικά στοιχεία.
- Στον Α/Σ προγραμματισμό, λέμε ότι ένα αντικείμενο-ποδήλατο, αποτελεί **στιγμιότυπο (instance)** μια **κλάσης αντικειμένων** που είναι γνωστά ως ποδήλατα.
- Η **κλάση** είναι το **σχεδιότυπο/σχεδιάγραμμα** (ο τύπος) από το οποίο κατασκευάζονται τα αντικείμενα-ποδήλατα.
- Η κλάση καθορίζει την κατάσταση και τη συμπεριφορά των αντικειμένων της.

Μ. Δικαιάκος, ΕΠΛ133

268

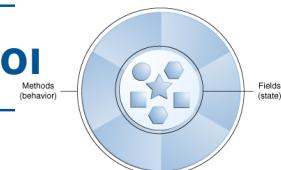
Κατάσταση, Συμπεριφορά, Ταυτότητα

- Τα αντικείμενα στον Α/Σ προγραμματισμό έχουν τρία βασικά χαρακτηριστικά:
 - **Κατάσταση (state)**
 - Τα δεδομένα που αποθηκεύονται στο εσωτερικό του αντικειμένου
 - **Συμπεριφορά (behavior)**
 - Οι ενέργειες που μπορεί να διεκπεραιώσει το αντικείμενο: συναρτήσεις που διαθέτει το αντικείμενο
 - **Ταυτότητα (identity)**
 - Το “κλειδί” μέσω του οποίου μπορούμε να αποκτήσουμε πρόσβαση στο αντικείμενο

Μ. Δικαιάκος, ΕΠΛ133

269

Πεδία Δεδομένων και Μέθοδοι



Ένα αντικείμενο λογισμικού:

- Αποθηκεύει στο εσωτερικό του πληροφορίες που είναι αποτέλεσμα των υπολογισμών-λειτουργιών του.
 - Η αποθήκευση γίνεται στα **πεδία δεδομένων** του αντικειμένου(fields/variables).
 - Οι αποθηκευμένες πληροφορίες αποτελούν την **κατάσταση (state)** του αντικειμένου.
- Δίνει πρόσβαση στη **συμπεριφορά** του μέσω **μεθόδων** (functions in some programming languages). Οι μέθοδοι:
 - λειτουργούν πάνω στην εσωτερική κατάσταση ενός αντικειμένου
 - αποτελούν τον βασικό μηχανισμό για επικοινωνία ανάμεσα σε αντικείμενα.

Μ. Δικαιάκος, ΕΠΛ133

270

Ταυτότητα αντικειμένων

- Η **ταυτότητα** ενός αντικειμένου λογισμικού: identifier (προσδιοριστής) μέσω του οποίου αποκτούμε πρόσβαση στο αντικείμενο.
- Αντιστοιχεί στη **διεύθυνση** στην οποία βρίσκεται αποθηκευμένο το αντικείμενο
- Στη Java, ο προσδιοριστής ενός αντικειμένου λέγεται **χειριστήριο** (handle) και μέσω αυτού:
 - αποκτούμε πρόσβαση στα πεδία του αντικειμένου (κατάσταση)
 - μπορούμε να ενεργοποιήσουμε τις ενέργειες του

Συστατικά στοιχεία αντικειμένων



Πολυπλοκότητα αντικειμένων

- Η πολυπλοκότητα της κατάστασης και της συμπεριφοράς αντικειμένων διαφορετικού τύπου μπορεί να διαφέρει. Π.χ.:
 - Ένας σκύλος έχει **κατάσταση** (name, color, breed, hungry) και **συμπεριφορά** (barking, fetching, wagging tail).
 - Ένα ποδήλατο επίσης έχει **κατάσταση** (current gear, current pedal cadence, current speed) και **συμπεριφορά** (changing gear, changing pedal cadence, applying brakes).
- Ο καθορισμός της κατάστασης και της συμπεριφοράς αντικειμένων του πραγματικού κόσμου είναι ένας πολύ καλός τρόπος για να ξεκινήσετε να σκέφτεστε με αντικειμενοστρεφή τρόπο.