



Επικοινωνία Ανθρώπου Μηχανής

Δρ. Ακουμιανάκης Δημοσθένης



Μοντέλα προγραμματισμού διεπαφών

Μοντέλα προγραμματισμού (γενικά)

❖ Procedural programming

- Ο κώδικας του προγράμματος εκτελείται ως μια ακολουθία εντολών (sequential order)

❖ Object-oriented

- Όλα είναι αντικείμενα που υλοποιούν μεθόδους

❖ Function-oriented

- Όλα είναι συναρτήσεις (με τη μαθηματική έννοια του όρου)



Μοντέλα προγραμματισμού

❖ Procedural programming

- Ο κώδικας του προγράμματος εκτελείται ως μια ακολουθία εντολών (sequential order)

```
statement 1  
statement 2  
statement 3  
-----  
-----  
statement n
```

❖ Προσέγγιση / δομές

- Εκτέλεση προγράμματος αρχίζει με κάποια *main()* και εκτελείται το ένα statement μετά το άλλο
- Υπάρχουν κατάλληλες δομές για branching με *if*, *for*, ή/και επανάληψη με *while* statement

✓ Όταν απαιτείται *user input*

- γίνεται κλήση της *read()*
- μπλοκάρεται η εκτέλεση του προγράμματος έως ότου ο χρήστης κάνει κάτι συνεχίζεται αμέσως μετά

Μοντέλα προγραμματισμού

❖ Πρόβλημα με procedural programming

- Πως αντιμετωπίζεται το θέμα της αναμονής για είσοδο από πολλαπλές πηγές εισόδου (πχ. πληκτρολόγιο και ποντίκι);
- Αν μπλοκάρουμε μια πηγή μπορεί να χάσουμε είσοδο από την άλλη

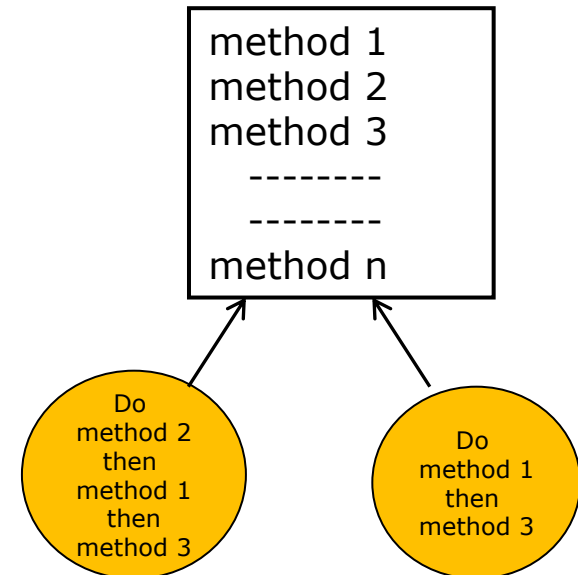
Μοντέλα προγραμματισμού

❖ Event-driven programming

- Η ροή προγράμματος και η ακολουθία των εντολών καθοδηγούνται από γεγονότα (events) που εκπέμπονται

❖ Προσέγγιση / δομές

- Το σύστημα περιμένει την ενεργοποίηση ενός γεγονότος από μια πηγή (πχ. το χρήστη) και αντιδρά σ' αυτά τα γεγονότα με τη χρήση μεθόδων του προγράμματος



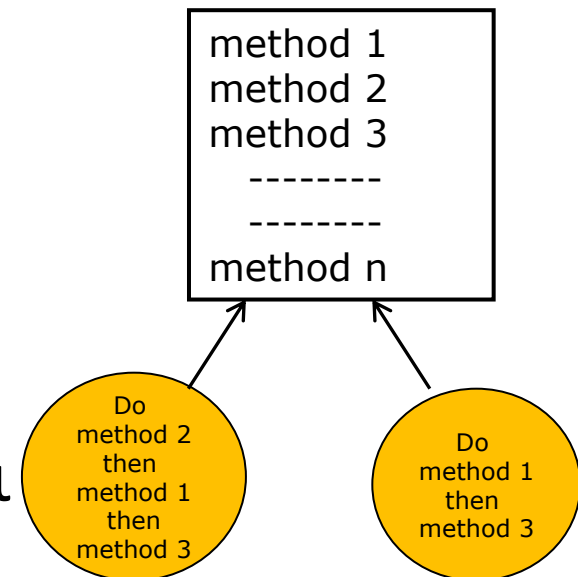
Μοντέλα προγραμματισμού

❖ Event-driven programming

- Η ροή προγράμματος και η ακολουθία των εντολών καθοδηγούνται από γεγονότα (events) που εκπέμπονται

❖ Κατ' αυτό τον τρόπο αντιμετωπίζονται

- Αναμονή για πολλαπλές πηγές εισόδου (input events)
- Απόφαση σχετικά με το τι θα γίνει με τον εκάστοτε τύπο γεγονότος



Κατηγορίες γεγονότων

❖ Γεγονότα αντιπροσωπεύουν

- Χειρισμούς του χρήστη σε κάποια συσκευή
 - ✓ Keyboard key press/release, function key press
 - ✓ Mouse move/drag/click
 - ✓ Mouse button press/release
 - ✓ Touchscreen finger tap/drag
 - ✓ Window resize/minimize/restore/close
 - ✓ Network activity or file input/output
- Ανάδραση του συστήματος
 - ✓ Αλλαγή κατάσταση αντικειμένου
 - ✓ Δημιουργία διαλόγου, κλπ.

Πηγές γεγονότων

❖ Γεγονότα παράγονται από

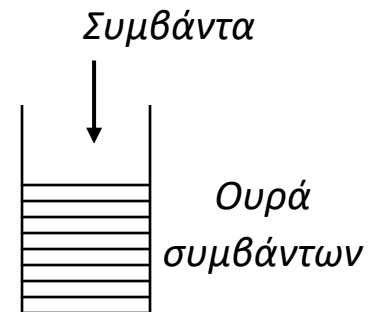
- Προγράμματα λογισμικού που εκτελούνται και παράγουν γεγονότα
- Νήματα (threads) που εκτελούνται παράλληλα και παράγουν ειδοποιήσεις (notifications)
- Συσκευές εισόδου δεδομένων (input devices) που αλλάζουν κατάσταση
- Το δίκτυο (network events)

Ανατομία ενός event

- ❖ Όταν παράγεται ένα event δημιουργείται μια δομή (data structure) με πληροφορία σχετικά με
 - Event origin (mouse button down, key up, etc.)
 - Event target action (viewmodel, listener)
 - Timestamp
 - Επιμέρους στοιχεία (event type-specific)
 - ✓ Mouse: οι συντεταγμένες (x, y coordinates) και ο αριθμός των clicks
 - ✓ Keyboard: κωδικός πλήκτρου (key code 21)
- ❖ Την πληροφορία αυτή αξιοποιούν οι *event handlers* για να χειριστούν το event

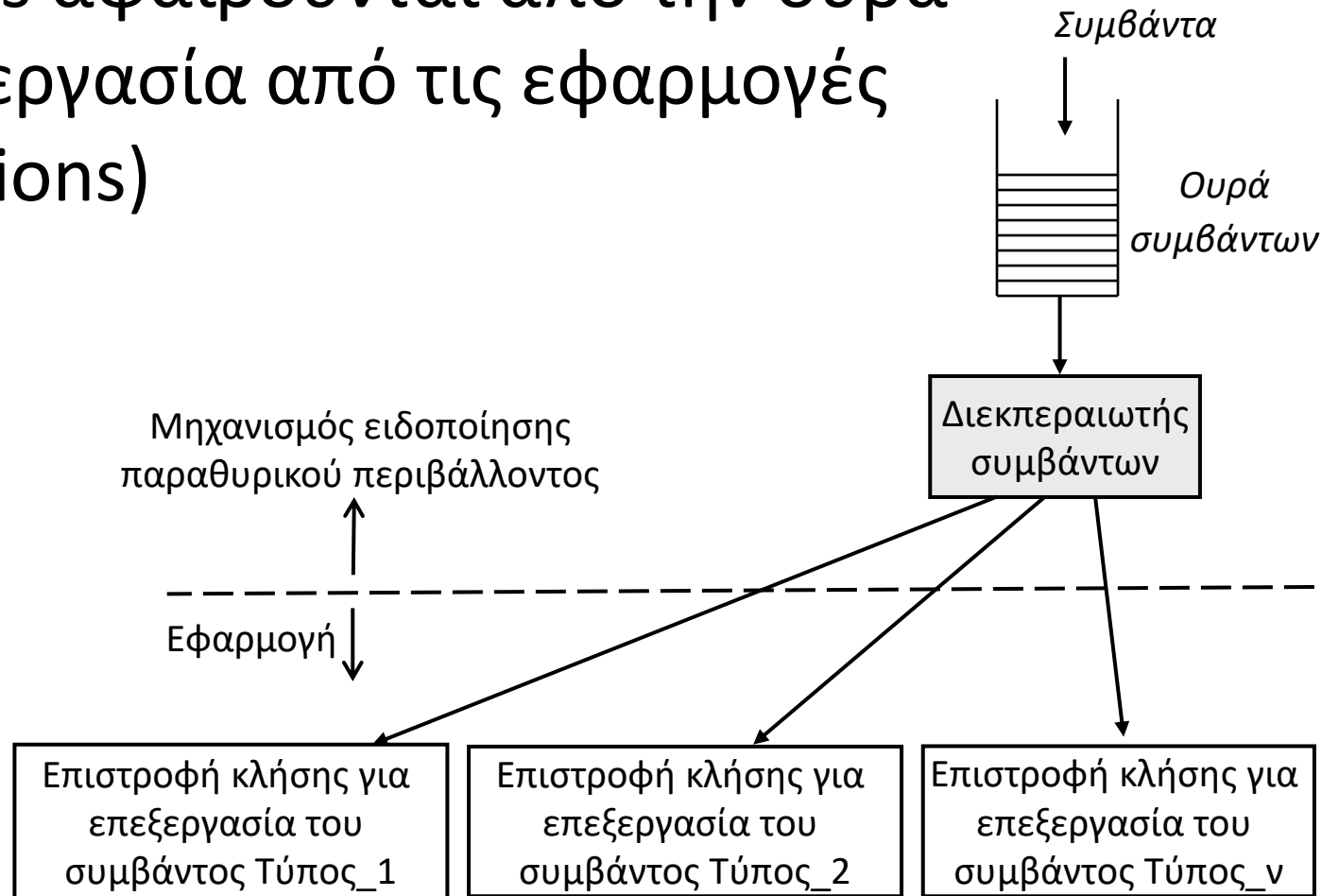
Διαχείριση γεγονότων

❖ Τα events τοποθετούνται σε μια ουρά



Διαχείριση γεγονότων (συν.)

- ❖ Τα events αφαιρούνται από την ουρά για επεξεργασία από τις εφαρμογές (applications)



Event processing

❖ Event loop

```
main(){  
  ... set up application data structures...  
  ... set up GUI....  
  // enter event loop  
  while(true){  
    Event e = get_event();  
    process_event(e); // event dispatch routine  
  }  
}
```

- ❖ Συνήθως ο προγραμματιστής δεν ελέγχει την παραπάνω δομή απλά χειρίζεται `process_event()` method

Event handling

❖ Σε τι συνίσταται ο προγραμματισμός

- Καθορισμός του κώδικα που θα εκτελεστεί όταν προκύψει το event (callbacks)
- Εργαλειοθήκες (built-in components)
 - ✓ Προσφέρουν έτοιμο κώδικα διαχείρισης των περισσότερων γεγονότων μιας συσκευής εισόδου
 - Buttons handle mouse up/down to change graphic
 - Text boxes update their contents on key press
 - ✓ Παράγουν ‘high-level’ events από low-level events
 - Text boxes παράγουν “change” events όταν αλλάζουν τιμή ή χάνουν το focus
 - Sliders παράγουν “change” events όταν το thumb αλλάζει θέση

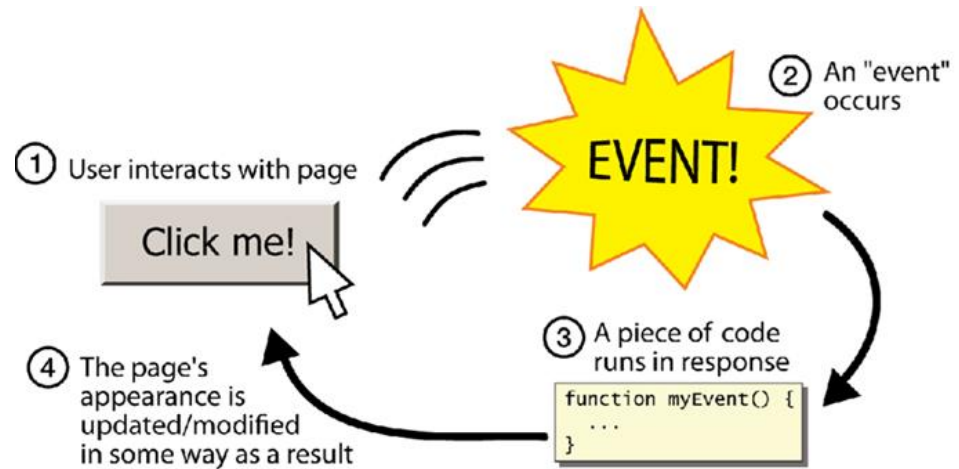
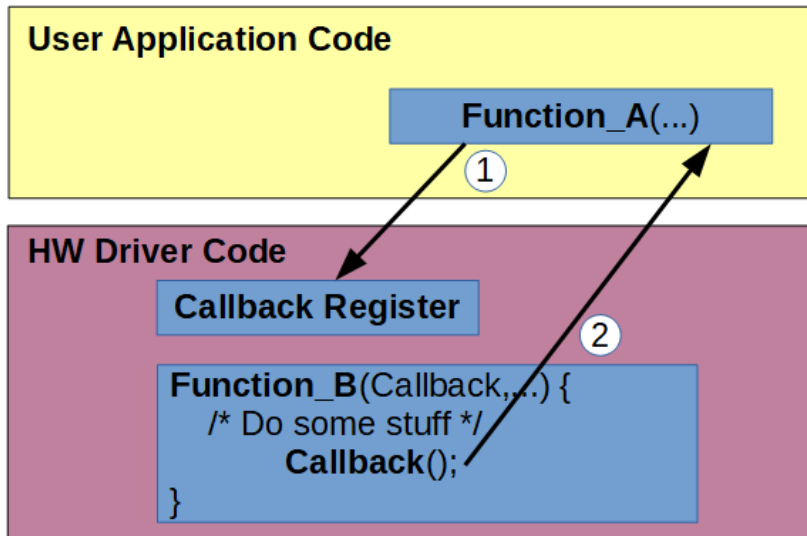
Event driven programs vs. languages

- ❖ Event-driven λογισμικό μπορεί να δημιουργηθεί με όλες τις γλώσσες προγραμματισμού
 - Προφανώς υπάρχουν περιπτώσεις που το καθιστούν ευκολότερο (γλώσσες που υποστηρίζουν high-level abstractions)
- ❖ Παραδείγματα
 - Java events είναι αντικείμενα
 - Python υποστηρίζει και object-oriented και procedural programming
- ❖ Κατά καιρούς υπήρξαν διαφορετικές εκδοχές για το πως μπορεί να υποστηριχθεί

Callback (ρουτίνα ανάκλησης)

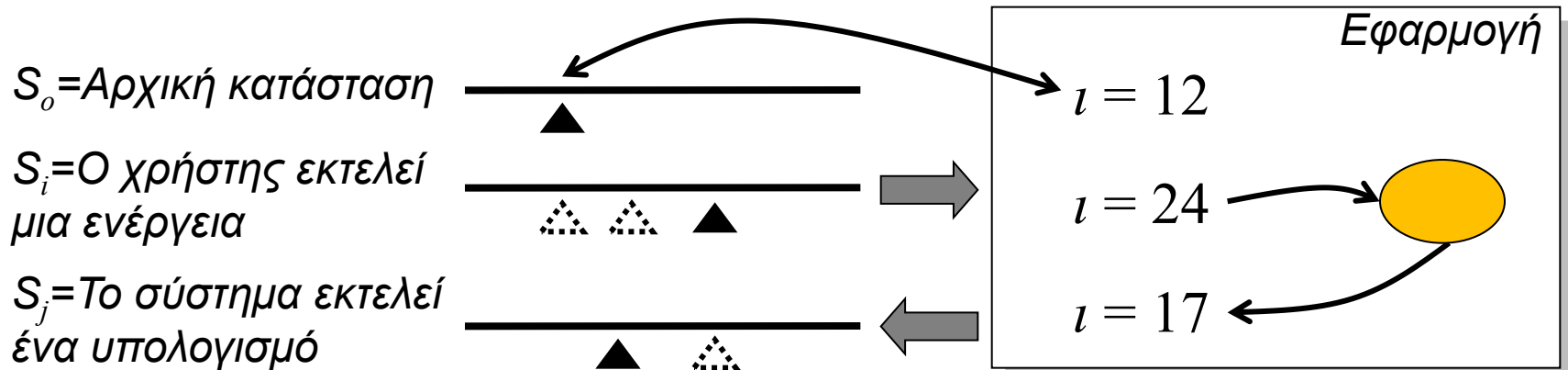
❖ Τυπικός ορισμός

- Τμήμα κώδικα που καλείται από άλλο τμήμα κώδικα
 - ✓ Function passed as an argument to another function



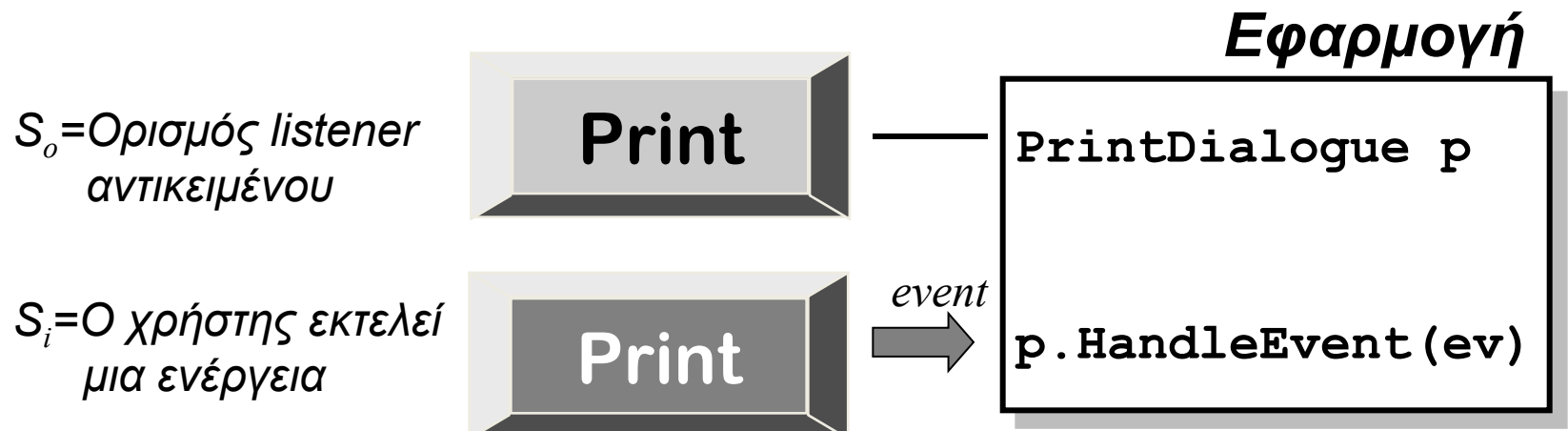
Active variables – Ενεργές μεταβλητές

- ❖ Αντί κλήσης ρουτίνας, το αντικείμενο συσχετίζεται με μια μεταβλητή της εφαρμογής
- ❖ Ένας ελεγκτής παρακολουθεί και αναλαμβάνει να ενημερώσει την μεταβλητή (με νέα τιμή) κάθε φορά που η κατάσταση του διαδραστικού αντικειμένου αλλάζει και αντίστροφα να ενημερώσει το διαδραστικό αντικείμενο κάθε φορά που η μεταβλητή παίρνει νέα



Αντικείμενα τύπου 'listener'

- ❖ Αντί της υλοποίησης μιας callback function, η εφαρμογή αντιστοιχεί ένα αντικείμενο τύπου listener που δέχεται και επεξεργάζεται τα γεγονότα του διαδραστικού αντικειμένου



Τέλος για σήμερα - Ερωτήσεις

