

---

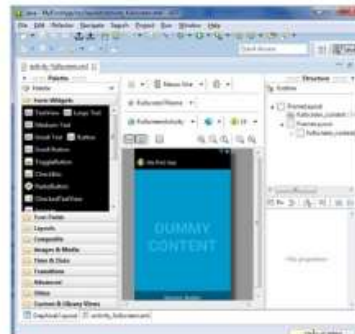
## Κεφάλαιο 7: Υλοποίηση εφαρμογών σε προγραμματιστικά περιβάλλοντα

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

**Ανάπτυξη εφαρμογών για φορητές συσκευές:** τείνουν να αντικαταστήσουν τους υπολογιστές και άλλες συσκευές. Τα δημοφιλέστερα Λειτουργικά Συστήματα είναι το iOS, το Android, το Windows Phone, το Symbian και το BlackBerry. Οι προγραμματιστές χρησιμοποιούν γλώσσες προγραμματισμού (π.χ. Java) και αντιμετωπίζουν προβλήματα που σχετίζονται με τους περιορισμένους πόρους των συσκευών (π.χ. επεξεργαστής, μνήμη), με το μικρό μέγεθος της διεπαφής χρήστη, με θέματα ασφάλειας, με τεχνολογίες αυτόματου προσδιορισμού της θέσης του χρήστη κ.ά. Οι χρήστες μπορούν να κατεβάσουν τις εφαρμογές από ηλεκτρονικά καταστήματα, π.χ. το Google Play για το Android, το App Store για το iOS και το Windows Phone Store για το Windows Phone.



**Εικόνα 7.1.** Δημοφιλή Λειτουργικά Συστήματα για φορητές συσκευές



**Εικόνα 7.2.** Περιβάλλον ανάπτυξης εφαρμογών για φορητές συσκευές με Android



**Εικόνα 7.3.** Το ηλεκτρονικό κατάστημα Google play

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

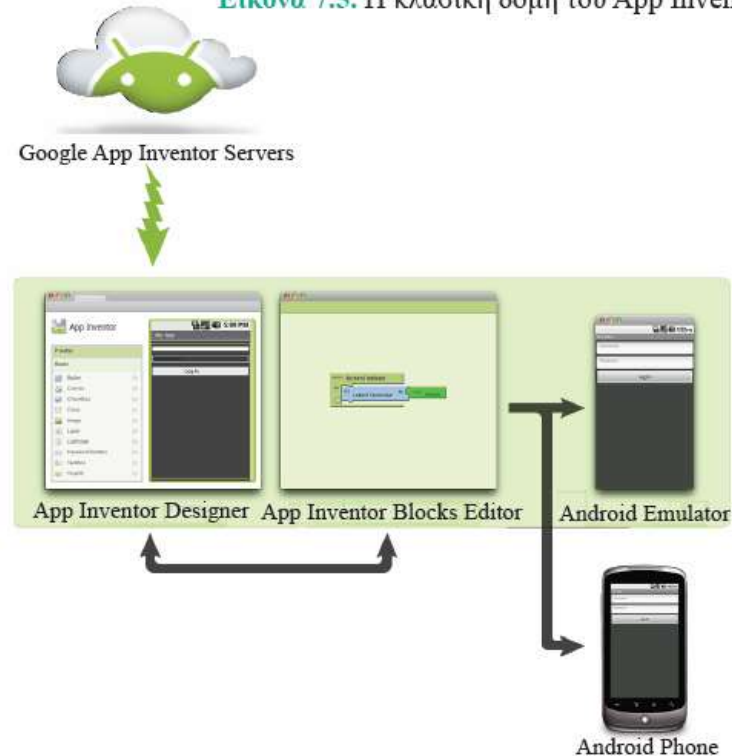
**Το εκπαιδευτικό περιβάλλον ανάπτυξης εφαρμογών App Inventor:** ελεύθερο, διαδικτυακό και οπτικό προγραμματιστικό περιβάλλον με πλακίδια (blocks), όπως και το Scratch. Η εφαρμογή μπορεί να εκτελεστεί και να δοκιμαστεί είτε απευθείας σε συσκευή που είναι συνδεδεμένη με τον υπολογιστή είτε με ενσωματωμένο emulator (προσομοιωτή).



Εικόνα 7.4. Κινητό τηλέφωνο με Android

Το **Android** είναι ένα δημοφιλές, ελεύθερο και ανοικτού κώδικα (open source) Λειτουργικό Σύστημα για φορητές συσκευές. Βασίζεται στον πυρήνα του Linux. Το πρώτο κινητό που κυκλοφόρησε με Android έφτασε στα ράφια των καταστημάτων στις 22 Οκτωβρίου 2008.

Εικόνα 7.5. Η κλασική δομή του App Inventor



Όταν ο χρήστης ολοκληρώσει την εφαρμογή του μπορεί είτε να τη «συσκευάσει», για να παραγάγει το τελικό πρόγραμμα σε μορφή **.apk** (Android application package), προκειμένου να το εγκαταστήσει στην **Android συσκευή** του, είτε ακόμη να το διανείμει δωρεάν ή εμπορικά στο Google Play. Εναλλακτικά, αν δεν υπάρχει διαθέσιμη κάποια συσκευή Android, ο χρήστης έχει τη δυνατότητα να δημιουργήσει και να ελέγξει τη λειτουργία της εφαρμογής του, χρησιμοποιώντας τον προσομοιωτή **Android Emulator**, ο οποίος είναι λογισμικό που εκτελείται τοπικά στον υπολογιστή του και συμπεριφέρεται σαν ένα κινητό τηλέφωνο.

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

### Διαδικασία δημιουργίας μιας εφαρμογής στο App Inventor:

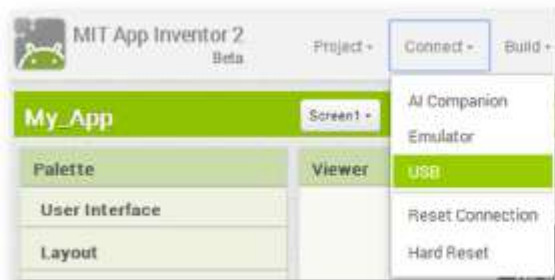
1. Επισκεπτόμαστε τον επίσημο ιστότοπο του App Inventor: <http://appinventor.mit.edu/explore>
2. Φτιάχνουμε λογαριασμό στην Google
3. Επιλέγουμε New Project και δίνουμε ένα όνομα για την εφαρμογή
4. Ανοίγει η καρτέλα Designer όπου ολοκληρώνουμε τη σχεδίαση της εφαρμογής μας και την παραμετροποίηση τους μέσω των ιδιοτήτων τους
5. Ανοίγουμε την καρτέλα Blocks. Ο προγραμματισμός γίνεται στο πλαίσιο Viewer, όπου σύρουμε από το πλαίσιο Blocks τα κατάλληλα πλακίδια
6. Εγκαθιστούμε και ελέγχουμε την εφαρμογή μας:
  - με σύνδεση σε κάποια φορητή συσκευή.
  - ή με Emulator για προσομοίωση φορητής συσκευής στον υπολογιστή μας



Εικόνα 7.6. App Inventor Designer



Εικόνα 7.7. App Inventor Blocks Editor



Εικόνα 7.8. Σύνδεση φορητής συσκευής



Εικόνα 7.9. Ο Emulator

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

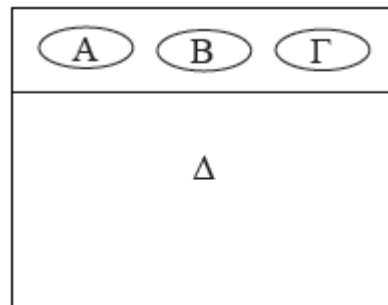
### Project με την εφαρμογή App Inventor

#### Φάση 1η: Ανάλυση

Ζητείται η εφαρμογή να έχει ένα κεντρικό μενού με 3 επιλογές. Η πρώτη επιλογή να ξεκινάει την εκτέλεση προστασίας της οθόνης, η δεύτερη επιλογή να περιέχει την εκτέλεση ενός παιχνιδιού και η τελευταία επιλογή να υπολογίζει τον Μέσο Όρο ενός μαθήματος.

#### Φάση 2η: Σχεδίαση

- Όταν πατηθεί το κουμπί A, εκτελείται η προστασία της οθόνης, όπου εμφανίζεται μια εικόνα ενός ήρεμου σκύλου, και, όταν ο χρήστης αγγίξει την περιοχή (Δ), τότε αλλάζει η εικόνα του σκύλου σε αγριεμένο και ακούγεται ο ανάλογος ήχος.
- Όταν πατηθεί το κουμπί B, εκτελείται το παιχνίδι. Ο χρήστης θα έχει τη δυνατότητα να ζωγραφίζει στην οθόνη του κινητού του.
- Όταν πατηθεί το κουμπί Γ, υπολογίζεται ο Μέσος Όρος του μαθήματος και εμφανίζεται η προαγωγή ή απόρριψη του μαθητή στο συγκεκριμένο μάθημα.



Σχήμα 7.1. Σχεδίαση οθόνης κινητού

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

### Σενάρια εντολών

(A) Για την πρώτη επιλογή προστασίας της οθόνης έχουμε τα παρακάτω σενάρια-ψευδοκώδικα:

Όταν το κουμπί A πατηθεί, τότε

απόκρυψε ό,τι δεν αφορά στον σκύλο και εμφάνισε τον ήρεμο σκύλο.

Όταν ο χρήστης αγγίζει την περιοχή Δ, τότε άλλαξε την εικόνα του σκύλου σε αγριεμένο και παίξε ήχο γαβγίσματος

Όταν ο χρήστης πάψει να αγγίζει την περιοχή Δ, τότε

άλλαξε την εικόνα του σκύλου σε ήρεμο.

(B) Για τη δεύτερη επιλογή του παιχνιδιού σχεδίασης έχουμε τα παρακάτω σενάρια-ψευδοκώδικα:

Όταν το κουμπί B πατηθεί, τότε

απόκρυψε ό,τι δεν αφορά στη ζωγραφική και καθάρισε την περιοχή Δ

Όταν ο χρήστης αγγίζει την οθόνη, τότε ζωγράφισε σε εκείνη τη θέση μια τελεία

Όταν ο χρήστης κινήσει το δάκτυλο του επάνω στην οθόνη, τότε

ζωγράφισε μια γραμμή μεταξύ της τρέχουσας θέσης του δακτύλου και της προηγούμενης.

Όταν κουνηθεί η φορητή συσκευή, τότε καθάρισε την οθόνη.



## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

### Σενάρια εντολών

(Γ) Για την τρίτη επιλογή όπου υπολογίζουμε τον μέσο όρο ενός μαθήματος και εμφανίζεται στην οθόνη το αποτέλεσμα της προαγωγής του μαθητή, έχουμε τα παρακάτω σενάρια:

Ορίζουμε και αρχικοποιούμε τις μεταβλητές:

- A (ο προφορικός βαθμός του A τετραμήνου σε ένα μάθημα) ← 0 (μηδέν)
- B (ο προφορικός βαθμός του B τετραμήνου στο ίδιο μάθημα) ← 0 (μηδέν)
- Γ (ο γραπτός βαθμός στο ίδιο μάθημα) ← 0 (μηδέν)
- MO (ο μέσος όρος βαθμολογίας του μαθήματος) ← 0 (μηδέν)

Επιλέγουμε να γίνουν οι υπολογισμοί και η εμφάνιση των αποτελεσμάτων με τη χρήση διαδικασιών:

**Όταν** το κουμπί Γ πατηθεί, τότε

απόκρυψε το σκύλο,

καθάρισε την περιοχή Δ

εμφάνισε την ετικέτα οδηγιών και το πλαίσιο όπου θα εισάγω τους βαθμούς

εμφάνισε το κουμπί «Τελικό αποτέλεσμα»

**Όταν** πατήσω το κουμπί «Τελικό αποτέλεσμα»,

κάλεσε τη διαδικασία για τον υπολογισμό του μέσου όρου

κάλεσε τη διαδικασία για την εμφάνιση των αποτελεσμάτων

**Διαδικασία: Υπολογισμός του μέσου όρου του μαθήματος**

Υπολόγισε τον μέσο όρο των 2 προφορικών βαθμών  $((A+B) / 2)$  και καταχώρισέ τον στον MO.

Υπολόγισε τον μέσο όρο  $((MO+Γ) / 2)$  και καταχώρισέ τον στον MO.

**Διαδικασία: Εμφάνιση των αποτελεσμάτων**

Εμφάνισε τον μέσο όρο του μαθήματος.

Αν ο MO είναι μεγαλύτερος ή ίσος από 10, τότε

εμφάνισε ότι ο μαθητής πέρασε το μάθημα

αλλιώς

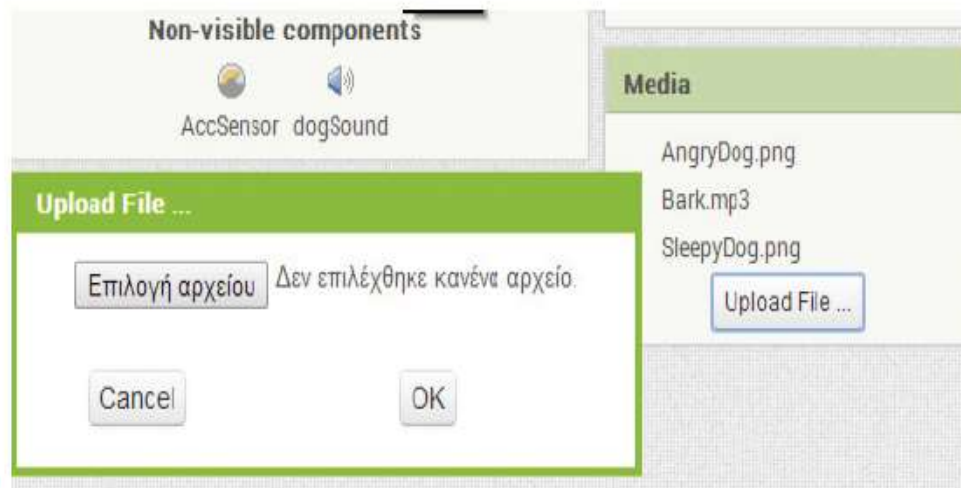
εμφάνισε ότι ο μαθητής δεν πέρασε το μάθημα.

Τέλος\_αν

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

### Φάση 3η: Υλοποίηση

1. Δημιουργούμε ένα νέο project με όνομα «Fun & Learn»
2. Θα χρησιμοποιήσουμε 2 εξωτερικά αρχεία: μια εικόνα του σκύλου και έναν ήχο (γάβγισμα), οπότε, με το κουμπί Upload File του πλαισίου Media (Εικόνα 7.10), ανεβάζουμε τα σχετικά αρχεία (Προσοχή: το συνολικό μέγεθος των αρχείων δεν πρέπει να υπερβαίνει τα 5 MB)



**Εικόνα 7.10.** Ανέβασμα εξωτερικών αρχείων που θα χρησιμοποιηθούν



## 7.1 Προγραμματισμός εφαρμογών

### Φάση 3η: Υλοποίηση

3. Εισάγουμε τα παρακάτω στοιχεία στο αντικείμενο Screen1 του πλαισίου Viewer. Αλλάζουμε τις ρυθμίσεις για κάθε αντικείμενο όπως στον πίνακα:



**Εικόνα 7.11.** Τα πλαίσια Viewer και Components μετά την εισαγωγή των αντικειμένων

Πίνακας 7.1 Τα συστατικά μέρη της εφαρμογής και οι ιδιότητές τους			
Από την ομάδα του πλαισίου Palette	Επιλέγουμε το αντικείμενο και το μεταφέρουμε στο πλαίσιο Viewer	Αλλάζουμε το Όνομα με το κουμπά Rename	Μεταβάλλουμε τις ιδιότητες - Properties
	Screen1 (Είναι ήδη εγκατεστημένο, οπότε αλλάζουμε μόνο τις ιδιότητες)		BackgroundColor: LightGrey Screen Orientation: Portrait Scrollable: (No) Title: Κεντρικό μενού
Layout	Horizontal Arrangement	Horizontal Arrangement1	PhotoArrangement Width: Fill Parent AlignHorizontal: Center
User Interface	Button (το μεταφέρουμε μέσα στο πλαίσιο Horizontal Arrangement)	Buttonscreensaver	BackgroundColor: Cyan Shape: Rounded Text: Screensaver
User Interface	Button (το μεταφέρουμε μέσα στο πλαίσιο Horizontal Arrangement)	Buttongame	BackgroundColor: Green Shape: Rounded Text: Game
User Interface	Button (το μεταφέρουμε μέσα στο πλαίσιο Horizontal Arrangement)	ButtonMO	BackgroundColor: Red Shape: Rounded Text: Μέσος Όρος
User Interface	Label	Label1	Visible: Hidden
User Interface	Slider	Slider1	MaxValue: 20.0 MinValue: 1.0 ThumbPosition: 5 Visible: hidden Width: Fill parent
User Interface	Slider	Slider2	MaxValue: 20.0 MinValue: 1.0 ThumbPosition: 10 Visible: hidden Width: Fill parent
User Interface	Slider	Slider3	MaxValue: 20.0 MinValue: 1.0 ThumbPosition: 15 Visible: hidden Width: Fill parent
User Interface	Label	Label2	Visible: Hidden
User Interface	Button	Button1	Text: Τελικό αποτέλεσμα TextAlignment: center Visible: hidden Width: Fill parent
Drawing and Animation	Canvas	Canvas	Paint Color: Blue Width: Fill Parent Height: Fill Parent
Sensors	Accelerometer Sensor	AccSensor	
Drawing and Animation	ImageSprite	DogSprite	Interval: 10 Picture: SleepyDog.jpg Rotates: (No) Visible: (Yes)
Media	Sound	DogSound	Source: Bark.mp3 MinimumInterval: 300

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

### Φάση 3η: Υλοποίηση

4. Επιλέγουμε από πάνω δεξιά το κουμπί Blocks και μεταφερόμαστε στον App Inventor Blocks Editor. Για να προγραμματίσουμε για ένα αντικείμενο, το επιλέγουμε από το πλαίσιο Blocks και από το συρτάρι εντολών που ανοίγει επιλέγουμε την εντολή και τη μεταφέρουμε στο πλαίσιο Viewer:

(Α) Δημιουργούμε τα παρακάτω σενάρια εντολών για την επιλογή προστασίας της οθόνης.

```
when Buttonscreensaver . TouchDown
do
  call Camvas . Clear
  set Label1 . Visible to false
  set Label2 . Visible to false
  set Slider1 . Visible to false
  set Slider2 . Visible to false
  set Slider3 . Visible to false
  set Button1 . Visible to false
  set DogSprite . Visible to true
```

```
when Camvas . TouchDown
  x y
do
  set DogSprite . Picture to "AngryDog.png"
  call dogSound . Play
```

```
when Camvas . TouchUp
  x y
do
  set DogSprite . Picture to "SleepyDog.png"
```

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

(B) Δημιουργούμε τα παρακάτω σενάρια για την επιλογή του παιχνιδιού σχεδίασης. Όπου ακουμπάει ο χρήστης ζωγραφίζει μια κουκκίδα και, όταν σύρει το δάκτυλο, ζωγραφίζει γραμμή.

```
when Buttongame . TouchDown
do
  set DogSprite . Visible to false
  set Label1 . Visible to false
  set Label2 . Visible to false
  set Slider1 . Visible to false
  set Slider2 . Visible to false
  set Slider3 . Visible to false
  set Button1 . Visible to false
  call Camvas . Clear
```

```
when AccSensor . Shaking
do
  call Camvas . Clear
```

```
when Camvas . Touched
  x y touchedSprite
do
  call Camvas . DrawCircle
    x get x
    y get y
    r Camvas . LineWidth
```

```
when Camvas . Dragged
  startX startY prevX prevY currentX currentY draggedSprite
do
  call Camvas . DrawLine
    x1 get prevX
    y1 get prevY
    x2 get currentX
    y2 get currentY
```

Για να μπορούμε όμως να σχεδιάσουμε κάτι άλλο από την αρχή, θα πρέπει να καθαρίσουμε την οθόνη. Αυτό γίνεται, αν κουνήσουμε τη συσκευή.



## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

(Γ) Δημιουργούμε τα παρακάτω σενάρια για την επιλογή του υπολογισμού του Μέσου Όρου (ΜΟ) και των αποτελεσμάτων προαγωγής του μαθητή σε ένα μάθημα.

### Αντιστοίχιση των μεταβλητών

Βαθμός Α τετραμήνου - A  
Βαθμός Β τετραμήνου - B  
Βαθμός γραπτού - G  
Μέσος όρος μαθήματος - mo

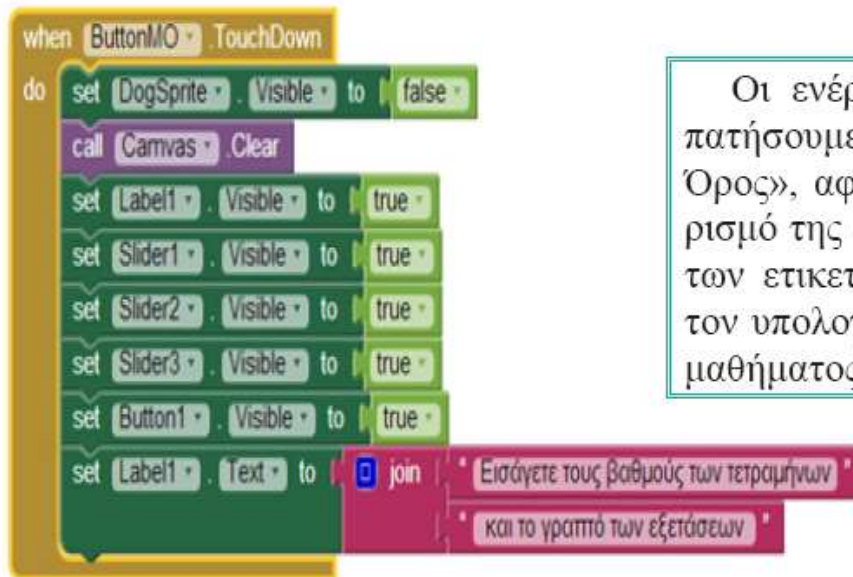
initialize global A to 0

initialize global B to 0

initialize global G to 0

initialize global mo to 0

Δημιουργία και αρχικοποίηση των μεταβλητών A, B, G, mo.



```
when ButtonMO TouchDown
do
  set DogSprite Visible to false
  call Canvas Clear
  set Label1 Visible to true
  set Slider1 Visible to true
  set Slider2 Visible to true
  set Slider3 Visible to true
  set Button1 Visible to true
  set Label1 Text to join "Εισάγετε τους βαθμούς των τετραμήνων "
  "και το γραπτό των εξετάσεων "
```

Οι ενέργειες που κάνουμε, όταν πατήσουμε το κουμπί «Μέσος Όρος», αφορούν κυρίως στον καθαρισμό της οθόνης και στην εμφάνιση των ετικετών και του κουμπιού για τον υπολογισμό του μέσου όρου του μαθήματος.

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

```
when Slider1 . PositionChanged
  thumbPosition
do set global A to get thumbPosition
  set Label1 . Text to get global A

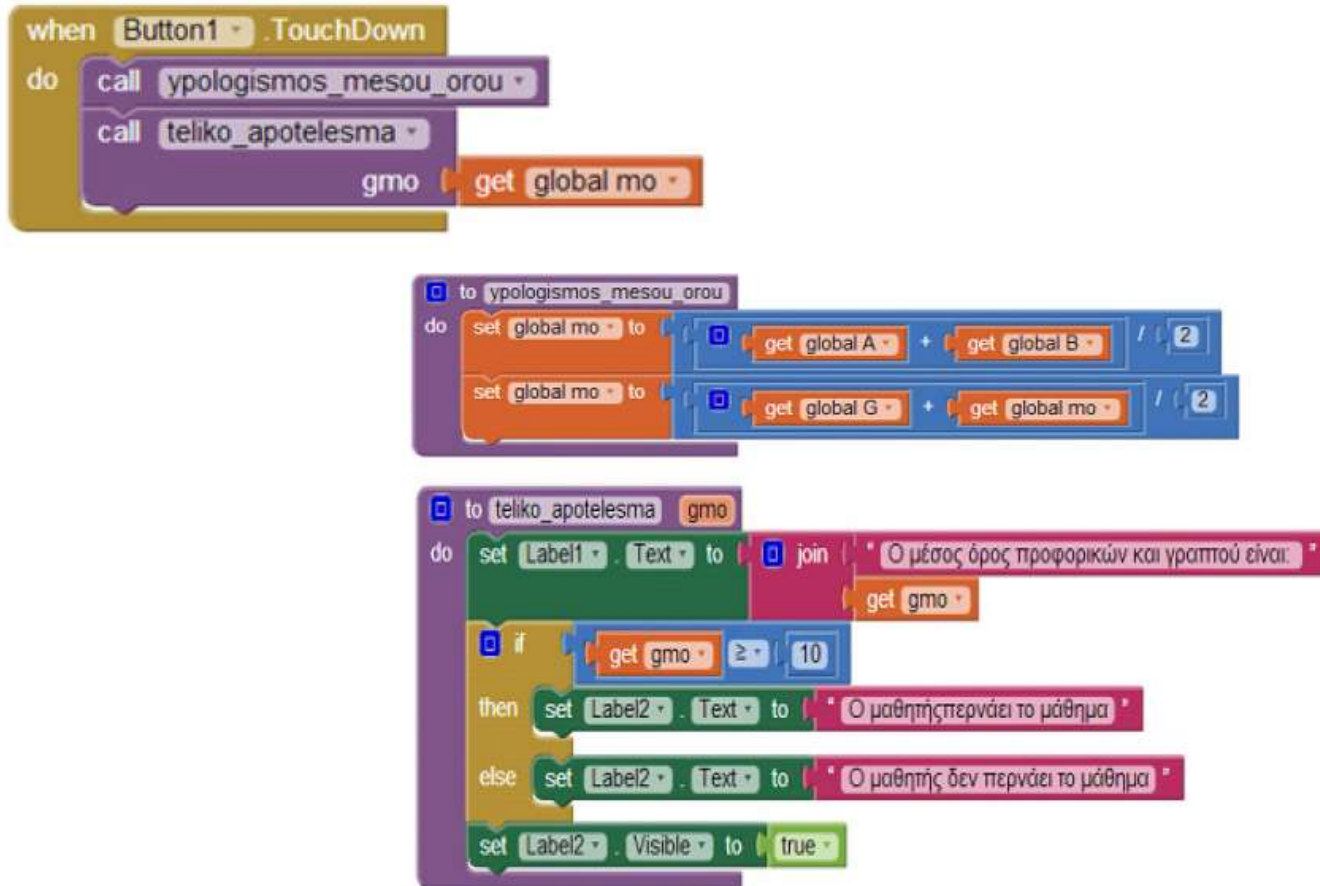
when Slider2 . PositionChanged
  thumbPosition
do set global B to get thumbPosition
  set Label1 . Text to get global B

when Slider3 . PositionChanged
  thumbPosition
do set global G to get thumbPosition
  set Label1 . Text to get global G
```

Με τον διπλανό κώδικα επιτυγχάνουμε, μετακινώντας την μπάρα πάνω σε μια κλίμακα από 1 έως 20, να ενημερώνονται οι αντίστοιχες μεταβλητές που αφορούν στους 3 βαθμούς του μαθήματος:  
Slider1 για τη μεταβλητή A  
Slider2 για τη μεταβλητή B  
Slider3 για τη μεταβλητή G

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

Όταν πατήσουμε το κουμπί «Τελικό αποτέλεσμα», τότε καλούμε τις Procedures (διαδικασίες) για τον υπολογισμό του μέσου όρου του μαθήματος και για την εμφάνιση του τελικού αποτελέσματος προαγωγής ή απόρριψης του μαθητή στο μάθημα. Αξίζει να σημειωθεί ότι κατά την κλήση της διαδικασίας «teliko\_apotelesma» περνάμε ως όρισμα τη μεταβλητή mo, η οποία μεταβιβάζει την τιμή της στην gmo.






## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές


### Φάση 4η: Λειτουργία

**Option One - RECOMMENDED**  
Build apps with an Android device and WiFi Connection (preferred): [Instructions](#)  
If you have a computer, an Android device, and a WiFi connection, this is the easiest way to test your apps.




Build your project on your computer      Test it in real-time on your device

**Option Two**  
Don't have an Android device? Use the Emulator: [Instructions](#)  
If you don't have an Android phone or tablet handy, you can still use App Inventor. Have a class of 30 students? Have them work primarily on emulators and share a few devices.



Build your project on your computer      Test it in real-time on your computer with the onscreen emulator

**Option Three**  
No WiFi? Build apps with an Android device and USB Cable: [Instructions](#)  
Some firewalls within schools and organizations disallow the type of WiFi connection required. If WiFi doesn't work for you, try USB.



Build your project on your computer      Test it in real-time on your device

**Εικόνα 7.12.** Οι τρόποι σύνδεσης για την εκτέλεση της εφαρμογής σε φορητές συσκευές ή προσομοίωση στον υπολογιστή (<http://appinventor.mit.edu/explore/ai2/setup.html>)

## 7.1 Προγραμματισμός εφαρμογών για φορητές συσκευές

### Φάση 5η: Συντήρηση

Προτάσεις για αλλαγές και βελτιώσεις:

1. Να κινείται ο σκύλος επάνω στην οθόνη.
2. Να γίνεται λήψη μιας φωτογραφίας από την κάμερα του κινητού και ο χρήστης να ζωγραφίζει πάνω στην εικόνα αυτή.
3. Να προστεθούν δυνατότητες αποθήκευσης και εκτύπωσης της ζωγραφιάς.
4. Να έχουμε δυνατότητα επιλογής χρώματος.
5. Να δημιουργηθεί μια λίστα με τα μαθήματα της τάξης, ώστε να φαίνεται και ο τίτλος του μαθήματος.
6. Να ενημερώσουμε την εφαρμογή σε οποιαδήποτε περίπτωση αλλαγής του τρόπου υπολογισμού του μέσου όρου.



**Εικόνα 7.13.** Διεπαφή χρήσης της εφαρμογής

## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

### Αντικειμενοστραφής προγραμματισμός (object-oriented programming)

Διασπά ένα πρόβλημα στα συστατικά του στοιχεία. Κάθε στοιχείο μετατρέπεται σε ένα αυτοτελές **αντικείμενο** (object), το οποίο περιέχει τις δικές του εντολές και τα δεδομένα που σχετίζονται με αυτό το αντικείμενο.

■ Μια **κλάση** (class) είναι ένα πρότυπο (καλούπι) που χρησιμοποιείται για τη δημιουργία ενός αντικειμένου. Κάθε αντικείμενο που δημιουργείται από την ίδια κλάση έχει παρόμοια, αν όχι ίδια, χαρακτηριστικά. Ένα αντικείμενο αποτελεί ένα μοναδικό και συγκεκριμένο **στιγμιότυπο** (instance) της κλάσης στην οποία ανήκει. Πρώτα ορίζονται οι κλάσεις και μετά δημιουργούνται τα αντικείμενα. Τα χαρακτηριστικά μιας κλάσης αντικειμένων ονομάζονται **ιδιότητες** (properties) και οι διαδικασίες που ορίζουν τις συμπεριφορές της ονομάζονται **μέθοδοι** (methods). Οι μέθοδοι στις οποίες εκτελούνται μόνο εντολές και δεν επιστρέφεται κάποια τιμή ονομάζονται **διαδικασίες** (procedures), ενώ οι μέθοδοι στις οποίες επιστρέφεται κάποια τιμή ονομάζονται **συναρτήσεις** (functions).

■ **Κληρονομικότητα**: η διεργασία μέσω της οποίας μια κλάση μπορεί να αποκτήσει (κληρονομήσει) τις ιδιότητες και μεθόδους μιας άλλης κλάσης. Έτσι δημιουργείται μια ιεραρχική ταξινόμηση. π.χ. κλάση Μέσο μεταφοράς, υποκλάση Όχημα, υποκλάση Αυτοκίνητο.



Δημοφιλείς γλώσσες αντικειμενοστρεφούς προγραμματισμού είναι η Java, η C++ και η Python.

**Κλάση:** ρομπότ

**Ιδιότητες:**

θερμοκρασία, θέση, ταχύτητα

**Μέθοδοι:**

- εκκίνηση εξερεύνησης
- έλεγχος τρέχουσας θερμοκρασίας
- αναφορά τρέχουσας θέσης
- ορισμός ταχύτητας

## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές



### Το περιβάλλον προγραμματισμού Alice

Είναι ένα 3D (**τρισδιάστατο**) περιβάλλον προγραμματισμού για τη δημιουργία κινούμενων γραφικών (animation) για την αφήγηση μιας ιστορίας, την ανάπτυξη διαδραστικών παιχνιδιών ή τη δημιουργία βίντεο που μπορεί να διαμοιραστεί στο Διαδίκτυο.

Ακολουθεί την **αντικειμενοστραφή** προσέγγιση προγραμματισμού. Στο Alice, 3D αντικείμενα (π.χ. σκηνικά, άνθρωποι, ζώα, φυτά, οχήματα) σχηματίζουν έναν εικονικό κόσμο και ο προγραμματιστής δημιουργεί οπτικά ένα πρόγραμμα με σύρσιμο και ταίριασμα κατάλληλων πλακιδίων (tiles ή blocks). Στο Alice έχουμε προγραμματισμό οδηγούμενο από γεγονότα (event-driven programming). Κάθε φορά που ο χρήστης κάνει κλικ με το ποντίκι ή πατάει ένα πλήκτρο, δημιουργείται ένα γεγονός που προκαλεί μια απάντηση. Για παράδειγμα, αν κάνουμε «κλικ σε ένα όχημα» (γεγονός), αυτό «αρχίζει να κινείται»(απάντηση). Ο χειρισμός των γεγονότων γίνεται με κατάλληλες μεθόδους.

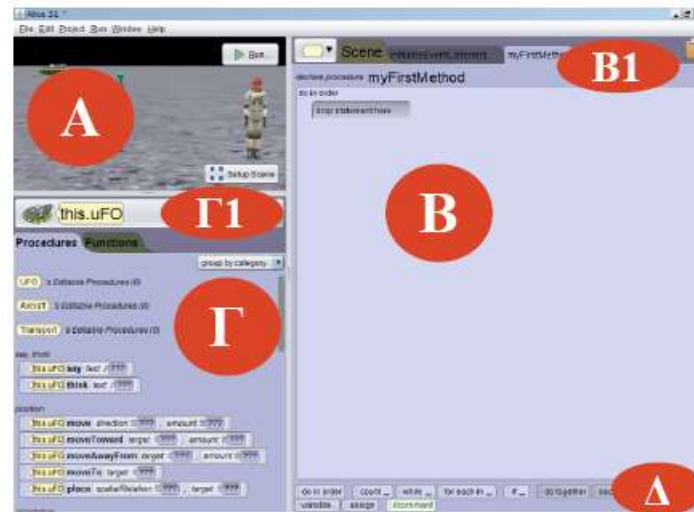
## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

### Διαδικασία δημιουργίας μιας εφαρμογής στο Alice

1. Το κατεβάζουμε από: <http://www.alice.org/> και το εγκαθιστούμε
2. Select Project και επιλέγουμε την αρχική σκηνή (Templates) του εικονικού μας κόσμου.
3. Το παράθυρο όπου προγραμματίζουμε χωρίζεται σε 4 μέρη:
  - A: Σκηνή (Scene), όπου αναπαριστάται ο εικονικός κόσμος που σχεδιάζουμε.
  - B: Συντάκτης Κώδικα (Code Editor), όπου προγραμματίζουμε μεταφέροντας τις μεθόδους από τους χώρους Γ και Δ, αφού πρώτα επιλέξουμε τη σωστή καρτέλα (περιοχή B1) στην οποία συντάσσουμε τον κώδικα.
  - Γ: Μέθοδοι (Procedures, Functions). Τις χρησιμοποιούμε, για να αποκτήσει συμπεριφορές το εκάστοτε αντικείμενο που έχουμε επιλέξει στο Γ1.



Εικόνα 7.14. Επιλογή σκηνής του εικονικού κόσμου



Εικόνα 7.15. Παράθυρο ανάπτυξης του προγράμματος



## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

4. Σχεδιάζουμε τον εικονικό μας κόσμο. Επιλέγουμε στη σκηνή A το κουμπί Setup Scene:

- Αρχικά από τον χώρο X επιλέγουμε αντικείμενα και τα τοποθετούμε στη σκηνή (Ψ). Τα αντικείμενα είναι ομαδοποιημένα σε κατηγορίες - κλάσεις (Biped-δίποδα, Flyer-ιπτάμενα, Prop-υποστηρικτικά, Quadruped-τετράποδα, Swimmer-υποθαλάσσια και Transport-μέσα μεταφοράς). Για ορισμένα αντικείμενα (άνθρωποι) έχουμε τη δυνατότητα να επιλέξουμε την ενδυμασία τους.
- Τοποθετούμε τα αντικείμενα στην επιθυμητή θέση στη σκηνή Ψ.
- Ρυθμίζουμε τις αρχικές ιδιότητες του κάθε αντικειμένου στο χώρο Z.

5. Για να επιστρέψουμε και να προγραμματίσουμε, επιλέγουμε το κουμπί Edit Code από τη σκηνή Ψ.

6. Για να δούμε τα αποτελέσματα της εκτέλεσης του κώδικα επιλέγουμε από τη σκηνή A το κουμπί Run.



Εικόνα 7.16. Παράθυρο σχεδίασης του εικονικού κόσμου



## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

### Project με την εφαρμογή Alice

**Σενάριο:** Υπάρχει ένας αστροναύτης (*AdultPerson*) στη Σελήνη, αντιλαμβάνεται ότι πλησιάζει ένα σκάφος και στρέφεται προς εκείνη την κατεύθυνση. Όμως το σκάφος είναι Άγνωστης Ταυτότητας Ιπτάμενο Αντικείμενο (*UFO*). Γίνεται η προσσελήνωση, κατεβαίνει ένας εξωγήινος (*Alien*) και πλησιάζει τον αστροναύτη. Ο αστροναύτης καλεί σε βοήθεια το κέντρο ελέγχου και ζητάει από τον χρήστη να τον απομακρύνει από τον *Alien*. Αφού δεν έχει επιτευχθεί ο επιθυμητός διάλογος μεταξύ τους, ο εξωγήινος απάγει τον αστροναύτη στο σκάφος και αναχωρούν για τον πλανήτη του.

1. Από τα templates επιλέγουμε το Moon. File->Save As και για κάθε επόμενη φορά File->Save.
2. Πατάμε το κουμπί Setup Scene. Εισάγουμε τον αστροναύτη από την κλάση Biped, χρησιμοποιώντας τη μέθοδο new Adult(...). Αφού του φορέσουμε τη στολή, τον τοποθετούμε στη σκηνή Ψ μπροστά και δεξιά. Τον μετονομάζουμε σε astronaut.
3. Εισάγουμε τον εξωγήινο από την κλάση Biped, χρησιμοποιώντας την μέθοδο new Alien(). Τον τοποθετούμε στη μέση της σκηνής και ορίζουμε την ιδιότητα Opacity (ορατότητα) σε 0.0 Αρχικά δεν φαίνεται ο alien, για να μπορέσουμε να τον εμφανίσουμε αργότερα και να δημιουργείται η εντύπωση ότι βγαίνει από το UFO.
4. Εισάγουμε το UFO από την κλάση Transport και την υποκλάση Aircraft, χρησιμοποιώντας τη μέθοδο new UFO(). Τον τοποθετούμε στη σκηνή Ψ πίσω και αριστερά.
5. Μπορούμε να μετακινήσουμε, περιστρέψουμε, ανυψώσουμε, αλλάξουμε μέγεθος κ.ά. στα αντικείμενά μας με τις επιλογές Default, Rotation, Translation, Resize καθώς επίσης και με τα μπλε βελάκια που βρίσκονται μπροστά και στο κέντρο της σκηνής Ψ.

## 7.2 Αντικειμενοστραφής π

6. Πατάμε το κουμπί Edit Code.

7. Συντάσσουμε τον κώδικα της εικόνας στη δική μας μέθοδο MyFirstMethod (B1). Μετά από το σύρσιμο μιας μεθόδου στον συντάκτη κώδικα τροποποιούμε τις παραμέτρους της. Επιπλέον δυνατότητες ορισμού παραμέτρων μάς δίνει η επιλογή του add detail.

Scene initializeEventListeners myFirstMethod UFO abduction

declare procedure myFirstMethod

do in order

- this.astronaut say "Ένα μικρό βήμα για τον άνθρωπο, αλλά ένα μεγάλο για την ανθρωπότητα" add detail
- this.astronaut think "Τι θόρυβος είναι αυτός!!!" add detail
- this.astronaut turn RIGHT, 0.4 add detail

do together

- this.astronaut say "Κέντρο ελέγχου, βλέπω ένα διαστημόπλοιο να πλησιάζει." add detail
- this.ufo moveTo this.alien, duration 5.0 add detail

- this.alien setOpacity 1.0 add detail
- this.alien moveToward this.astronaut, 10.0 add detail
- this.alien say "&#@%" add detail
- this.astronaut moveAwayFrom this.alien, 1.0 add detail

count up to 2

- this.astronaut say "Κέντρο βοήθεια ένας εξωγήινος μου μιλάει!!!" add detail
- this.alien say "&#@%" add detail

loop

- this.astronaut say "ΒΟΗΘΕΙΑ χρήστη πάρα τα βελόκια να με σώσεις!!!", duration 2.0 add detail
- this.alien say "%#@ 10" add detail

WholeNumber xronos ← 10

while xronos > 0 is true

- this.alien say "!" + xronos add detail
- xronos ← xronos - 1

loop

- this.alien moveTo this.astronaut add detail
- this.astronaut setVehicle this.alien
- this.alien turnToFace this.ufo, duration 1.0 add detail
- this.alien moveTo this.ufo, duration 2.0 add detail
- this.alien setOpacity 0.0, duration 2.0 add detail
- this.astronaut setOpacity 0.0, duration 2.0 add detail

do together

- this.ufo abduction
- this.camera setVehicle this.ufo

### Πίνακας 7.2 Επεξήγηση εντολών στην MyFirstMethod

Ο astronaut «μιλάει» και «σκέφτεται» το αντίστοιχο κείμενο που είναι γραμμένο.

Στρίβει προς μια κατεύθυνση. Το 1.0 είναι μια πλήρης περιστροφή 360°.

Οι μέθοδοι say και moveTo μέσα στο do together εκτελούνται ταυτόχρονα. Η διάρκεια (duration) είναι σε δευτερόλεπτα.

Εμφανίζεται ο alien, μετακινείται μπροστά στον astronaut και του μιλάει, ενώ αυτός απομακρύνεται προς τα πίσω.

Επανάληψη του διαλόγου 2 φορές (δομή count up to).

Δίνεται χρονικό περιθώριο στον astronaut να συνετιστεί.

Μετά την κλήση για βοήθεια ορίζουμε μια μεταβλητή χρόνος με αρχική τιμή 10.

Στη δομή επανάληψης while ο alien αναφέρει τον εναπομείναντα χρόνο. Η μεταβλητή χρόνος μειώνεται κατά ένα σε κάθε επανάληψη, μέχρι να μηδενιστεί.

Οπότε και ο alien κινείται προς τον astronaut.

Με τη μέθοδο setVehicle τα δύο αντικείμενα αντιμετωπίζονται σαν ένα (ομαδοποίηση), δηλαδή η οποιαδήποτε αλλαγή στη συμπεριφορά του alien προκαλεί την ίδια συμπεριφορά και στον astronaut.

Εξαφανίζουμε (κάνουμε αόρατους) τους alien, astronaut.

Εκτελούνται παράλληλα: η κλήση της procedure: abduction (απαγωγή) και η ομαδοποίηση της κάμερας με το UFO.

## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

8. Για να δημιουργήσουμε μια procedure, επιλέγουμε το εξάγωνο από την περιοχή B1->UFO->Add UFO Procedure. Πληκτρολογούμε το όνομα της διαδικασίας abduction, οπότε δημιουργούνται 2 νέες καρτέλες (UFO, abduction). Εισάγουμε τις μεθόδους όπως φαίνονται στην εικόνα 7.18.



Εικόνα 7.18. Ο κώδικας της procedure-διαδικασίας που αφορά στην απαγωγή

## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

9. Για να δώσουμε τη δυνατότητα να χειρίζεται ο χρήστης με το δεξί και αριστερό βελάκι του πληκτρολογίου τον astronaut, επιλέγουμε καρτέλα InitializeEventListeners->Add Event Listeners->Keyboard-> addArrowKeyPressListeners. Στη συνέχεια μεταφέρουμε τη μέθοδο ελέγχου if\_ από τον χώρο Δ στον συντάκτη κώδικα (B) και δημιουργούμε τον κώδικα όπως φαίνεται στην εικόνα 7.19. Η λογική που εφαρμόζουμε είναι «Αν (if) πατηθεί το αριστερό πλήκτρο, τότε (then) να κινηθεί ο astronaut αριστερά, αλλιώς, αν (if) πατηθεί το δεξί βελάκι, τότε (then) να κινηθεί ο astronaut δεξιά, αλλιώς (else) να εκφράσει τη δυσαρέσκειά του. Ο λόγος που επιλέξαμε για να υλοποιηθεί αυτός ο κώδικας σε αυτή την καρτέλα και όχι εντός της μεθόδου MyFirstMethod είναι ότι ο κώδικας που γράφεται σε αυτό τη σημείο εκτελείται από την αρχή εκτέλεσης της εφαρμογής μέχρι το τέλος αυτής.

10. Οποιαδήποτε στιγμή μπορούμε να εκτελέσουμε τον κώδικα και να ελέγξουμε τα αποτελέσματα που εμφανίζονται με τη μορφή βίντεο.

11. Δεν ξεχνάμε κατά διαστήματα και στο τέλος να αποθηκεύσουμε την εφαρμογή μας.



Εικόνα 7.19. Ο κώδικας στη μέθοδο InitializeEventListeners για τον χειρισμό του αστροναύτη από τον χρήστη



## 7.2 Αντικειμενοστραφής προγραμματισμός σε 3D περιβάλλον συσκευές

### Ερωτήσεις - Δραστηριότητες:

1. Εμπλουτίστε την εφαρμογή με καλύτερες κινήσεις των αντικειμένων, όπως π.χ. όταν μετακινείται ο αστροναύτης δεξιά ή αριστερά να στρίβει και αναλόγως.
2. Επεκτείνετε την εφαρμογή, εισάγοντας έναν ακόμη αστροναύτη, και εμπλουτίστε την αλληλεπίδραση μεταξύ όλων των αντικειμένων.
3. Αφού πρώτα περιηγηθείτε σε έτοιμα προγράμματα που έχουν δημιουργηθεί με το Alice, συζητήστε σε ομάδες μια δική σας ιστορία που νομίζετε ότι είναι εφικτό να υλοποιηθεί στο συγκεκριμένο περιβάλλον. Συμβουλευτείτε τον καθηγητή σας και υλοποιήστε την.