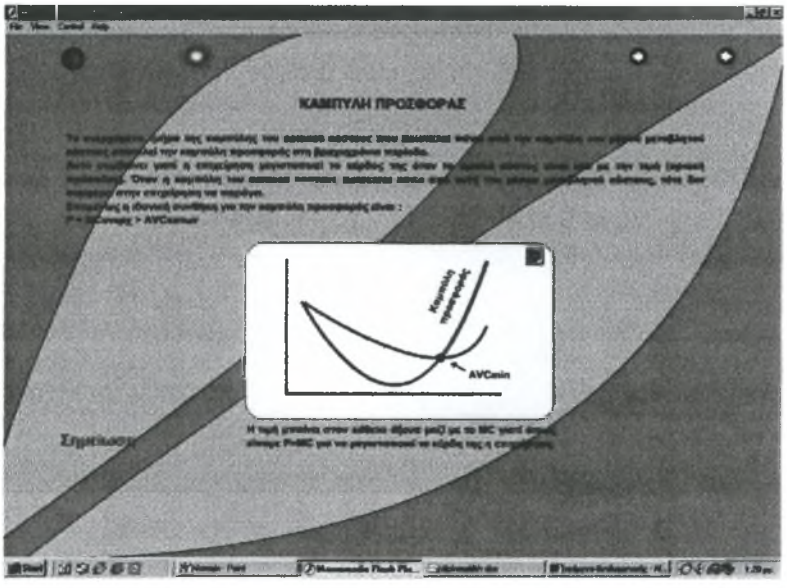
Πανεπιστήμιο Μακεδονίας
Μεταπτυχιακό τμήμα:
Management Information Systems

Πολυμεσική εφαρμογή για την ενίσχυση μαθητών Γ' λυκείου στο μάθημα «Αρχές Οικονομικής Θεωρίας»



Γκίνος Γρηγόρης (Α.Μ. 9/00)

Επιβλέπων καθηγητής : Πομπόρτσης Ανδρέας
Εξεταστής καθηγητής : Οικονομίδης Αναστάσιος

ΠΕΡΙΕΧΟΜΕΝΑ

σελ.

Σε ποιόν απευθύνεται.....2

Στόχος 2

Εξήγηση της εφαρμογής.....3

Εισαγωγικές έννοιες.....4

Ήχοι.....5

1° επίπεδο – είσοδος.....5

2° επίπεδο – περιεχόμενα κεφαλαίου.....6

3° επίπεδο – κυρίως μέρος 6
1° κεφάλαιο.....7
2° κεφάλαιο.....13
3° κεφάλαιο.....29
4° κεφάλαιο.....35
5° κεφάλαιο.....40

Λυμμένες ασκήσεις.....47

Διαγωνίσματα.....49
ερωτήσεις θεωρίας.....62
ασκήσεις.....65

Αξιολόγηση.....80

Το κουμπί «που βρίσκομαι».....85

Βιβλιογραφία.....92

ΣΕ ΠΟΙΟΝ ΑΠΕΘΥΝΕΤΑΙ

Η εργασία αυτή απευθύνεται σε μαθητές τρίτης λυκείου οι οποίοι διαγωνίζονται κάθε Ιούνιο στο μάθημα «Αρχές Οικονομικής Θεωρίας» προκειμένου να εισαχθούν στην ανώτερη και ανώτατη εκπαίδευση. Ασφαλώς και θα μπορούσε να απευθυνθεί και σε οποιονδήποτε άνθρωπο θέλει να λάβει μια ιδέα της οικονομικής επιστήμης αλλά από τη σκοπιά του μαθητή μόνο. Θα λέγαμε ότι είναι αυστηρώς δομημένη και κατασκευασμένη ώστε να ικανοποιήσει τις ανάγκες μαθητών Γ' λυκείου περισσότερο, και λιγότερο κάποιον που θέλει γενικότερα να μάθει τις βασικές αρχές της οικονομικής επιστήμης. Αυτό γιατί και το σχολικό βιβλίο του συγκεκριμένου μαθήματος στοχεύει στην απόδοση κάποιων βασικών γνώσεων και αντιθέτως οι εξετάσεις γίνονται τόσο εξειδικευμένες σε λεπτομέρειες του βιβλίου ακόμα και σε λεπτομέρειες που παρουσιάζονται πολύ απλουστευμένες, σε βαθμό που να μην ισχύουν. Γι' αυτό λοιπόν βρισκόμενη μέσα στα όρια του σχολικού βιβλίου και των πανελληνίων εξετάσεων, θα μπορούσαμε να πούμε ότι αυτή η εφαρμογή εξυπηρετεί περισσότερο έναν μαθητή που προετοιμάζεται να εξεταστεί στην απομνημόνευση γνώσεων και επίλυση ασκήσεων, παρά κάποιον άνθρωπο που θέλει να γνωρίσει γενικότερες έννοιες. Με λυπεί αυτό το γεγονός, και γι'αυτό έχει προσδωθεί στην εργασία, χαρακτήρας πιο ανθρώπινος από αυτόν του σχολικού βιβλίου, προσπαθώντας να ξεφύγει από τα όρια της δίχως νόημα απομνημόνευσης γνώσεων.

ΣΤΟΧΟΣ

Στόχος αυτής της εφαρμογής είναι να μπορέσει να δώσει ένα ισχυρό στήριγμα στον μαθητή έτσι ώστε να ξεφύγει από την λογική της στείρας παρουσίασης γνώσεων του σχολικού βιβλίου. Είναι φανερό ότι ένας μαθητής ή φοιτητής ή οποιοσδήποτε εκπαιδευόμενος, όταν γνωρίζει για πρώτη φορά κάποιες έννοιες, να μην πουλκαταλαβαίνει πλήρως κάποια πράγματα, ή επίσης, όσον αφορά κάποιες θεωρίες που έχουν τεράστια σημασία, ο ίδιος να μην καταλαβαίνει τη σημασία τους. Αυτό το πρόβλημα δεν μπορεί να το ξεπεράσει το βιβλίο, απλά τονίζοντας το χρώμα κάποιων γραμμάτων, όπως ήθισται άλλωστε στα βιβλία του ελληνικού υπουργείου παιδείας. Μέσω μιας multimedia εφαρμογής ένας μαθητής εντυπωσιάζεται από κάποια συμβάντα της εφαρμογής, έτσι ώστε να αυξάνονται οι πιθανότητες να θυμάται μια σημαντική θεωρία. Έπειτα μέσω της διάδρασης, ο χρήστης βλέπει πολλά πράγματα τα οποία θα του ήταν άγνωστα ή θα χρειαζόταν να τα φανταστεί από μόνος του. Επίσης η εφαρμογή προσφέρει στον χρήστη δυνατότητες διάδρασης με προσωμειώσεις τύπων, θεωριών, και νόμων προκειμένου να μπορεί να καταλαβαίνει σ' ένα βαθμό πιό πέρα τι γίνεται με τις διάφορες θεωρίες που απλά παρουσιάζει το βιβλίο. Μεγάλη σημασία για τον μαθητή έχει η χρήση της φωνής. Ασφαλώς και είναι ευρέως αποδεκτό ότι η χρήση φωνητικής διδασκαλίας σε συνδυασμό με τη γραπτή, φέρνει καλύτερα αποτελέσματα απ' ότι μόνη της η γραπτή. Φωνή χρησιμοποιείται σε σημεία που χρίζουν ιδιάιτερης εξήγησης, και το ύφος και ο τόνος της φωνής δίνει μια διαφορετική σημασία σε καθετί που εξηγείται. Όλα αυτά έχουν σαν αποτέλεσμα την καλύτερη κατανόηση της θεωρίας, την αύξηση της ικανότητας του μαθητή στο να επιλύει ασκήσεις και την αντιμετώπιση ενός μεγάλου προβλήματος, -στην περίπτωση του μαθητή που πρωτογνωρίζει μια επιστήμη-, την κατανόηση της σημασίας που έχει για την επιστήμη – ή και για τις πανελλήνιες- η κάθε θεωρία και κάθε νόμος. Όσον αφορά τα διαγωνίσματα, ένα μεγάλο πλεονέκτημα που αποκτά ο μαθητής είναι να έχει τη δυνατότητα ασφαλώς να λύσει περισσότερες ασκήσεις και όχι μόνο όσες του

παρέχονται στο σχολικό βιβλίο, αλλά επίσης, είτε τις έχει καταφέρει είτε όχι, να μπορεί να δει και τις λύσεις αυτών. Επίσης μεγάλο πλεονέκτημα προσφέρει η δυνατότητα που έχει ο χρήστης να αξιολογήσει τις γνώσεις του. Όσον αφορά τα διαγωνίσματα είναι γνωστό ότι πολλοί από εμάς αν δεν ξέρουμε μια απάντηση σε κάποιο ερώτημα, θα γυρίσουμε στη θεωρία και θα δώσουμε την απάντηση κάνοντας κατ' αυτό τον τρόπο «ζαβολιά». Έχοντας στόχο την αποφυγή αυτού του γεγονότος, ο χρήστης μόλις εισαχθεί σε ένα διαγώνισμα δεν μπορεί να φύγει μέχρις ότου το ολοκληρώσει και αξιολογηθεί η απόδοσή του. Επίσης όταν ο χρήστης είναι ενήμερος στο σύνολο της θεωρίας μπορεί να ξεκινήσει τον τομέα των διαγωνισμάτων όπου αρχίζει να λύνει πιό εύκολα διαγωνίσματα στην αρχή, και έπειτα, προχωράει στην επόμενη βαθμίδα δυσκολίας, μόνο αν αποδείχθηκε ικανός σε προηγούμενο διαγώνισμα. Μ' αυτό τον τρόπο ο μαθητής δεν ξεκινάει κατ' ευθείαν από τα δύσκολα. Αντιθέτως, σιγά σιγά, αρχίζει να συνηθίζει τις ασκήσεις, και λύνοντας αποκτά την ικανότητα επίλυσης πιό δύσκολου διαγωνίσματος.

ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ

Η εφαρμογή έχει κτιστεί με το flash mx και χρησιμοποιούνται διάφορες δυνατότητες που παρέχει, όπως κουμπιά μπάρες κύλισης, πεδία κειμένου, movie clips, κ.α. Επίσης εκτός από αυτά δεν θα μπορούσαμε να παραλείψουμε την γλώσσα που χρησιμοποιείται για την επιτυχή αλληλοεπίδραση των διαφόρων ταινιών που εμπεριέχονται μέσα στην εφαρμογή, αλλά και για την επιτυχή διάδραση του χρήστη με αυτή. Η γλώσσα που χρησιμοποιείται δεν είναι άλλη από αυτή που χρησιμοποιεί το flash, δηλαδή την Actionscript. Κατά τη διάρκεια της ανάλυσης θα εξηγηθεί και θα περιγραφεί κάθε στοιχείο κώδικα που χρησιμοποιείται και οι ενέργειες αυτού.

Scene είναι ένα μέρος μιας ταινίας. Κάτι σαν τα κεφάλαια ενός βιβλίου. Γι' αυτό δεν είναι τυχαίο ότι κάθε κεφάλαιο αντιπροσωπεύεται από ένα scene.

Κάθε scene αποτελείται από τις στρώσεις (layers). Σε κάθε στρώση τοποθετούνται ομέδες στιγμιοτύπων έτσι ώστε να βρίσκονται όλα μαζί. Πχ. Όλα τα κουμπιά βρίσκονται στην ίδια στρώση όπως επίσης σε άλλη στρώση βρίσκονται οι εικόνες του background κτλ.

Frame είναι ένα καρέ χρόνου που διαρκεί κάποια κλάσματα του δευτερολέπτου και περιέχει διάφορα αντικείμενα και στιγμιότυπα συμβόλων. Σε κάθε frame εκτελούνται διάφορες ενέργειες από κώδικα που βρίσκεται στο ίδιο το καρέ είτε σε κάποιο movie clip.

Movie clip είναι οτιδήποτε δημιουργείται και τρέχει κατά τη διάρκεια της ταινίας. Μπορεί να περιέχει γραφικά, κείμενο, κίνηση, ή διάφορα άλλα οπτικά, ηχητικά στοιχεία και μη.

Το σύμβολο (symbol) είναι το καλούπι για μια σειρά από στιγμιότυπα που υπάρχουν μέσα στην ταινία, και αποθηκεύεται στη βιβλιοθήκη της ταινίας.

Στιγμιότυπο (instance) είναι ένα υπόδειγμα ενός συμβόλου και είναι τα «παιδιά» που έχουν δημιουργηθεί από τη μήτρα symbol, και αυτά είναι που υπάρχουν και αποτελούν μια ταινία.

Static text είναι ένα πεδίο κειμένου το οποίο δεν έχει καμία ιδιότητα και ούτε μπορεί να περιλαμβάνει μπάρα κύλισης. Αυτό χρησιμοποιείται όπου χρειάζεται κείμενο χωρίς ιδιότητες όπως σε ετικέττες, τίτλους και κείμενο που χρησιμεύει ως γραφικό.

Dynamic text είναι ένα πεδίο κειμένου που μπορεί να δεχθεί μπάρα κύλισης (αρκεί να έχει την ιδιότητα scrollable) καθώς επίσης μπορεί να λαμβάνει τιμές ή και κείμενο από διάφορες ενέργειες που προκαλούνται έπειτα από τη διάδραση του

χρήστη. Το κείμενο που περιλαμβάνει έχει τη δυνατότητα να μεταβάλλεται ανάλογα με την εκτέλεση διαφόρων ενεργειών που ορίζονται με κώδικα.

Input text είναι ένα πεδίο κειμένο όπου ο χρήστης εισάγει μια τιμή ή λέξη η οποία προσδίδεται σε μία μεταβλητή που έχει οριστεί σ' αυτό το πεδίο κειμένου. Αυτό το είδος κειμένου χρησιμοποιείται αρκετά στα διαγωνίσματα όπως θα δούμε παρακάτω, γιατί ο χρήστης θα εισάγει την απάντησή του σε ένα τέτοιο πεδίο κειμένου, και θα χρείαζεται να γίνεται έλεγχος της μεταβλητής του πεδίου κειμένου, με την σωστή απάντηση, προκειμένου να αξιολογηθεί ο μαθητής.

ΗΧΟΙ

Στο πρώτο καρέ της εργασίας τοποθετήθηκε κάποιος κώδικας προκειμένου να εισαχθούν διάφοροι ήχοι επεξηγηματικοί κτλ. Αυτοί οι ήχοι βρίσκονται αποθηκευμένοι σε κάποιον φάκελο έξω από το αρχείο της κύριας ταινίας. Μόλις ξεκινά η εφαρμογή, όλοι οι ήχοι που θα χρειαστούν, φορτώνονται σε μια εικονική μνήμη, και καλούνται από την εφαρμογή, όταν χρειαστούν. Για τη φόρτωσή τους χρησιμοποιείται ο παρακάτω κώδικας:

```
zitisi=new Sound();
zitisi.loadSound("sounds/zitisi.mp3", false);
```

η πρώτη γραμμή δημιουργεί ένα αντικείμενο (object) με το όνομα «zitisi». Στη δεύτερη γραμμή ορίζεται ότι στο αντικείμενο «zitisi» θα φορτωθεί το αρχείο ήχου «zitisi.mp3» που βρίσκεται στον φάκελο «sounds». Όπως βλέπουμε οι ήχοι που φορτώνονται είναι συμπιεσμένοι σε μορφή mp3. Η λέξη «false» είναι μια boolean μεταβλητή που δηλώνει αν ο ήχος θα είναι streaming ή αν θα συμπεριφέρεται ως γεγονός. Αν συμπεριφέρεται ως γεγονός, τότε θα μπορεί να παίξει μόνο αφότου φορτωθεί. Αν είναι streaming sound, τότε θα μπορεί να παίξει και κατά τη διάρκεια που φορτώνει. Όταν χρειαστεί να παίξουν, τότε στο αντίστοιχο movie clip ή frame τοποθετείται η εντολή zitisi.start(); που σημαίνει ότι το αντικείμενο «zitisi» που εμπεριέχει τον ήχο «zitisi.mp3», θα ξεκινήσει να παίζει. Μόλις χρειαστεί να σταματήσει, τότε αντίστοιχα παρατίθεται η εντολή zitisi.stop();

Σε πολλά swf η φωνή μπορεί να σταματήσει και αλλιώς. Αν ο χρήστης κλείσει ένα παράθυρο που εμπεριέχει φωνή τότε πρέπει να οριστεί να κλείνει και η φωνή μαζί με το παράθυρο. Αυτό επιτυγχάνεται με την εντολή:

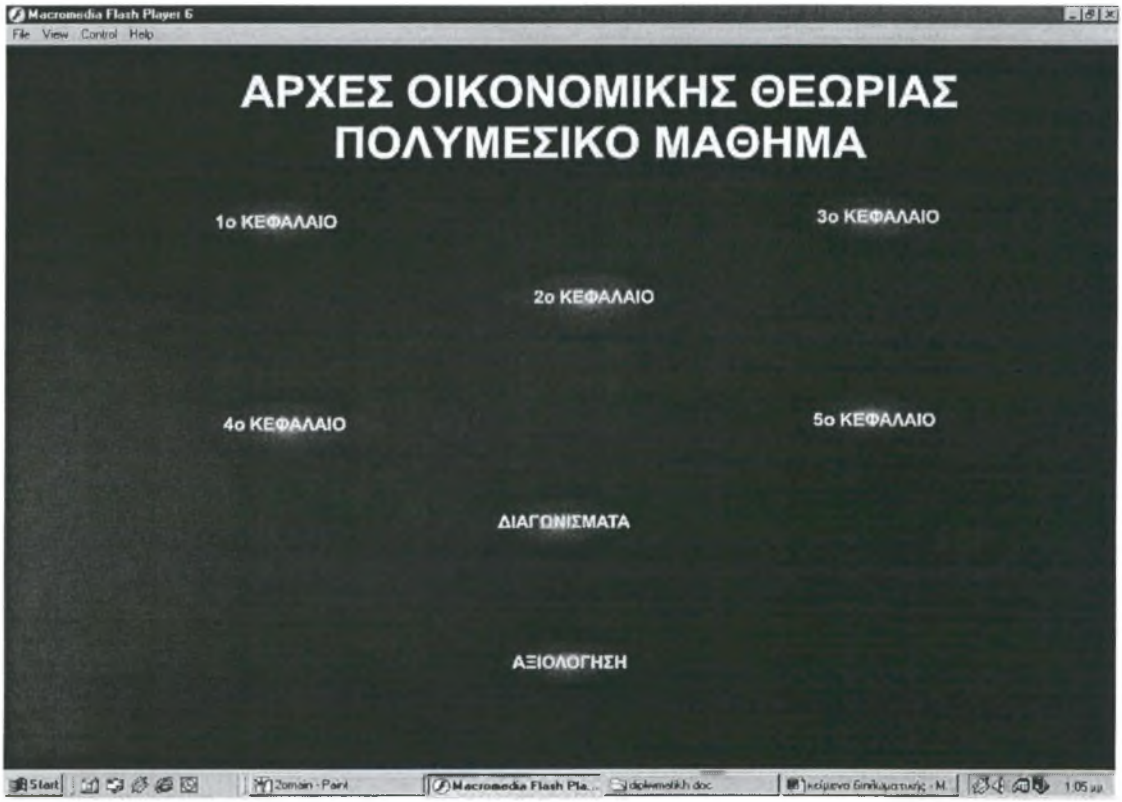
```
eidhkpdscroll.onUnload=function()
{
	kpdkoil.stop();
	kpdkyrth.stop();
	kpdeytheia.stop();
}
```

στην πρώτη γραμμή ορίζεται μια συνάρτηση που αρχίζει μόλις πάψει το movie clip «eidhkpdscroll» να υπάρχει. Τότε ορίζεται να εκτελεστούν οι εντολές μέσα στο βρόγχο οι οποίες αφορούν το σταμάτημα κάποιων ήχων.

Η εφαρμογή αποτελείται από τρία επίπεδα.

ΠΡΩΤΟ ΕΠΙΠΕΔΟ – ΕΙΣΟΔΟΣ

Στο πρώτο επίπεδο δίνονται στον χρήστη οι πιθανές επιλογές που έχει όσον αφορά αυτό που έχει να μελετήσει και να αξιολογηθεί. Παρατίθενται σ' αυτό το επίπεδο κουμπιά, που παρέχουν στο χρήστη τη δυνατότητα να μπει στο περιβάλλον του εκάστοτε κεφαλαίου. Υπάρχουν κουμπιά ισάριθμα με τα κεφάλαια, δηλαδή πέντε. Επίσης υπάρχει κουμπί για την οδήγηση του χρήστη στο περιβάλλον των τελικών διαγωνισμάτων, και ένα κουμπί για να μπορέσει ο χρήστης να μεταβεί στον τομέα της αξιολόγησης της εφαρμογής. Τα κουμπιά σ'αυτό το σημείο είναι σύμβολα που περιέχουν κείμενο και μεταβάλεται το χρώμα του κειμένου ανάλογα με το αν ο χρήστης έχει τον δείκτη του ποντικιού πάνω στο κουμπί.



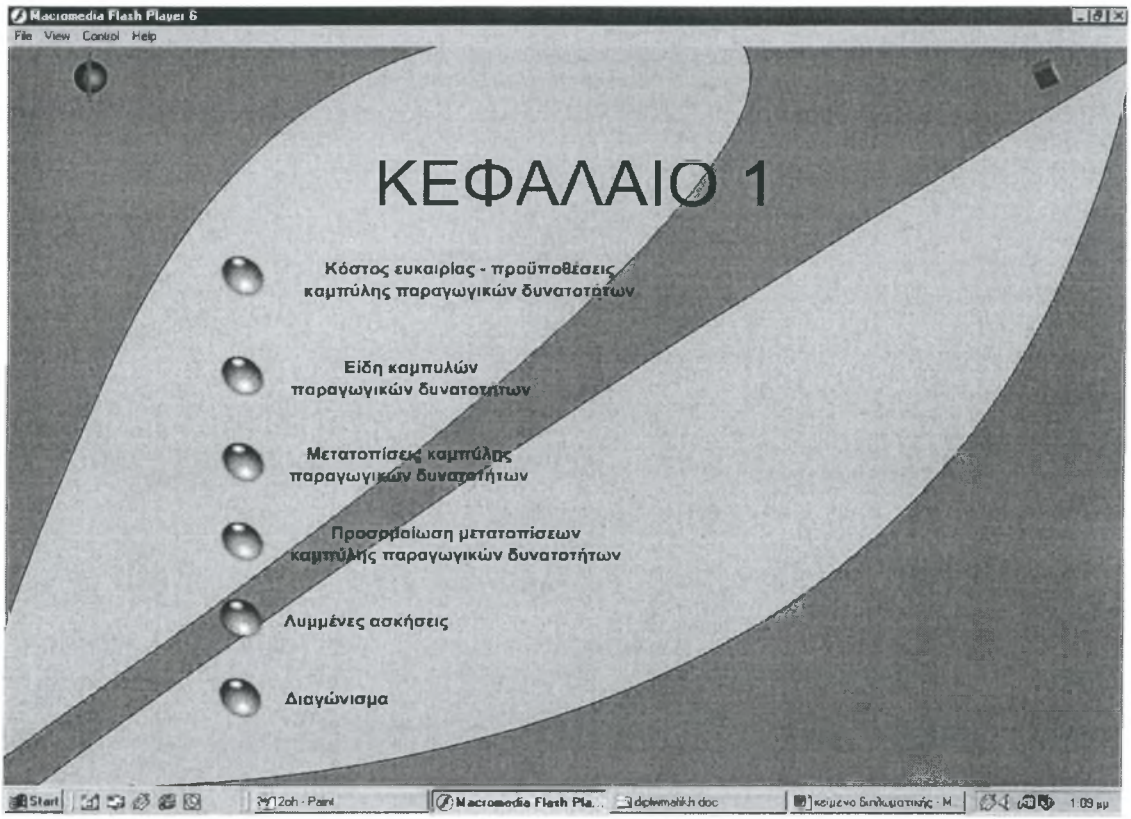
Σε περίπτωση που πιέζεται ένα τέτοιο κουμπί, εκτελείται ένας κώδικας actionscript. Συγκεκριμένα για τα κουμπιά αυτού του επιπέδου ο κώδικας είναι :

```
on (release)
{
gotoAndPlay("2o kefalaio",56);
}
```

η πρώτη γραμμή σημαίνει ότι τη στιγμή που ο χρήστης έχοντας πατήσει το κομπί του ποντικιού, μόλις τ' αφήσει, να εκτελεστεί η ενέργεια μέσα στις αγκύλες. Η γραμμή μέσα στις αγκύλες σημαίνει ο player να παίξει το scene με το όνομα "2o kefalaio" στο frame υπ'αριθμόν 56. Πίσω από τα κουμπιά βρίσκονται κάποια clips που στη συγκεκριμένη περίπτωση εξυπηρετούν μόνο από άποψη ομορφιάς.

ΔΕΥΤΕΡΟ ΕΠΙΠΕΔΟ – ΠΕΡΙΕΧΟΜΕΝΑ ΚΕΦΑΛΑΙΟΥ

Στο δεύτερο επίπεδο βρίσκονται τα περιεχόμενα του κάθε κεφαλαίου. Περιλαμβάνει έναν αριθμό κουμπιών που το κάθε ένα στέλνει τον χρήστη στην αντίστοιχη θεματική ενότητα του κεφαλαίου. Ο αριθμός των κουμπιών διαφέρει σε κάθε κεφάλαιο ανάλογα με τις θεματικές ενότητες που περιέχει το καθένα. Επίσης σ'αυτό το επίπεδο ενυπάρχει κι ένα κουμπί που επιστρέφει τον χρήστη στο κεντρικό μενού, δηλαδή στο πρώτο επίπεδο.



Οι εντολές που κρύβονται πίσω από κάθε κουμπί είναι οι ίδιες όπως και στο πρώτο επίπεδο με τη διαφορά ότι τώρα τα κουμπιά των θεματικών ενοτήτων δεν αναφέρονται σε κάποιο άλλο scene, αλλά στο ίδιο scene, στο αντίστοιχο frame.

```
on (release)
{
gotoAndPlay(40);
}
```

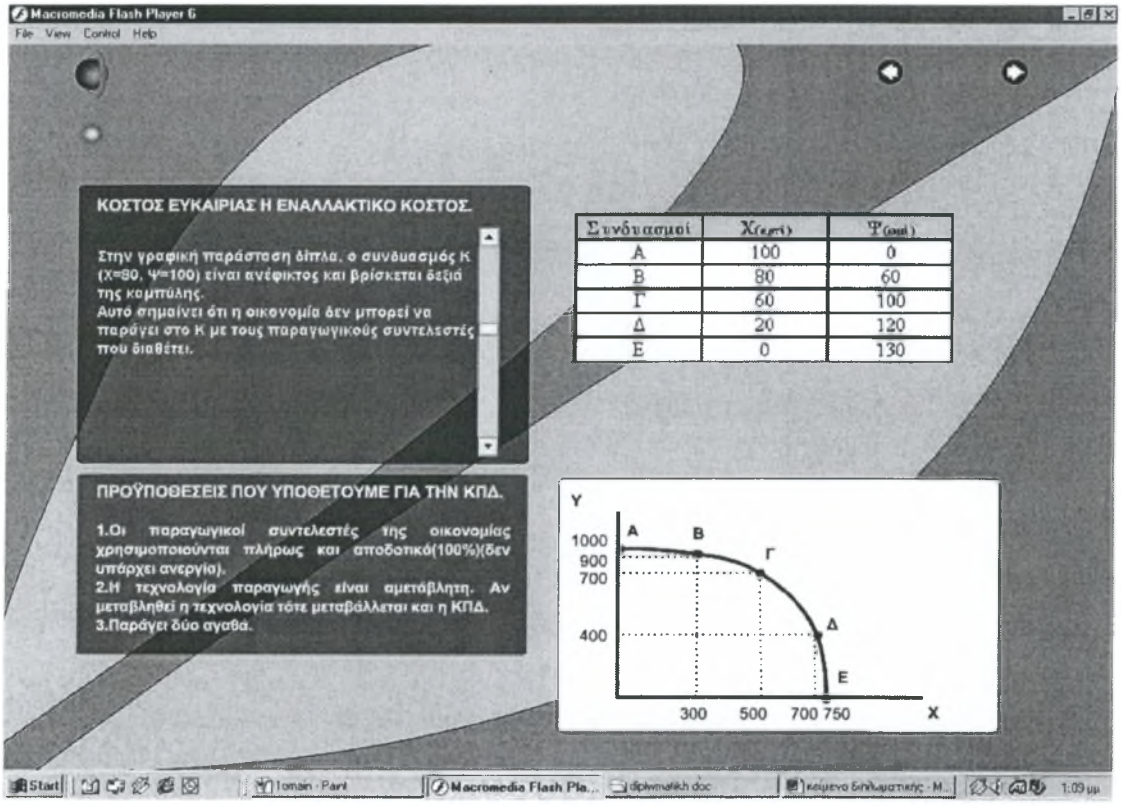
η γραμμή μέσα από τις αγκύλες σημαίνει η ταινία να παίξει στο ίδιο scene, το καρέ υπ' αριθμόν 40.
Το κουμπί που επιστρέφει τον χρήστη στο αρχικό επίπεδο έχει ίδιας δομής κώδικα όπως και των κουμπιών του πρώτου επιπέδου.

ΤΡΙΤΟ ΕΠΙΠΕΔΟ – ΚΥΡΙΩΣ ΜΕΡΟΣ

Αυτό το επίπεδο είναι και το μεγαλύτερο της εφαρμογής. Περιέχει όλη της επεξήγηση της θεωρίας καθώς επίσης και υποδείγματα λυμμένων ασκήσεων. Αυτό το επίπεδο αποτελείται από πέντε scenes. Αυτά είναι τα πέντε κεφάλαια της θεωρίας.

1° κεφάλαιο:

Αποτελείται από δεκαεννιά καρέ τα οποία διαμοιράζονται στη θεωρία και στην εκτέλεση διαφόρων εντολών. Το πρώτο καρέ αναφέρεται στην πρώτη θεματική ενότητα, και αποτελείται από δύο πεδία κειμένου, ένα static text και ένα dynamic text με μια μπάρα κύλισης. Καθώς η μπάρα κατεβαίνει, ο χρήστης παρακολουθεί και διάφορους πίνακες και γραφικές παραστάσεις που επεξηγούν το κείμενο. Ανάλογα με την τοποθεσία του δείκτη της μπάρας κύλισης, εμφανίζονται αντίστοιχα και οι επιπλέον πληροφορίες.



Επίσης η μπάρα κύλισης που υπάρχει δίπλα στο dynamic text έχει κάποιες ιδιότητες να εκτελεί κάποιες ενέργειες που ορίζονται με κώδικα. Οι ενέργειες αυτές προσδίδονται στη συγκεκριμένη μπάρα και όχι σε ολόκληρο το καρέ, άσχετα αν λαμβάνουν χώρα μέσα στο συγκεκριμένο καρέ. Αυτές είναι:

```
onClipEvent (enterFrame)
{
setScrollProperties(5, 1, 71);

        if ( getScrollPosition()>15 && getScrollPosition()<56)
        {
        loadMovie("pinakaskpd1.swf",_parent.invisiblepin);
        }
}
```

η πρώτη γραμμή του κώδικα σημαίνει ότι κάθε φορά που διανύεται ένα καρέ, να εκτελούνται οι ενέργειες που εμπεριέχονται στις αγκύλες. Στη δεύτερη γραμμή ορίζονται οι ιδιότητες της μπάρας κύλισης. Δηλαδή σ' αυτή την περίπτωση ορίζεται ότι το τετραγωνάκι που αποτελεί τον δείκτη τοποθεσίας του scrollbar, να καταλαμβάνει 5 θέσεις στην μπάρα, η default θέση του να είναι στη θέση 1 της μπάρας, και η μπάρα να αποτελείται από 71 θέσεις. Η τρίτη γραμμή περιλαμβάνει τον βρόγχο if και σημαίνει ότι αν ο δείκτης της μπάρας κύλισης βρίσκεται μεταξύ της θέσης 15 και 56, τότε να εκτελεστούν οι ενέργειες μέσα στον βρόγχο. Η εντολή getScrollProperties λαμβάνει την θέση στην οποία βρίσκεται ο δείκτης της μπάρας. Αυτές οι ενέργειες που αναφέρονται στην επόμενη γραμμή, είναι να φορτωθεί η ταινία με το όνομα "pinakaskpd1.swf", επάνω στο movieclip με το όνομα invisiblepin. Η ταινία δεν αποτελείται μόνο από ένα αρχείο swf αλλά από πολλά, χάριν αποφυγής υπαρφόρτωσης του υπολογιστή. Είναι προτιμότερο ολόκληρη η εφαρμογή να είναι χωρισμένη σε μικρότερα swf αρχεία και να ανοίγουν μόνο όταν χρειάζονται, παρά να ανοίγουν όλα μαζί και να υπάρχουν, είτε χρειάζονται είτε όχι, και έτσι να βαρύνεται το σύστημα. Η εντολή loadMovie ανοίγει το συγκεκριμένο swf, και έπειτα όταν αυτό είναι αχρείαστο, κλείνει με την εντολή unloadMovie. Το πρόθεμα _parent μπροστά από το όνομα του movie clip δείχνει ότι είμαστε ένα επίπεδο πιο βαθιά στην εφαρμογή, δηλαδή βρισκόμαστε μέσα στο component «scrollbar» και επειδή θέλουμε οι ενέργειες να συμβούν στο καρέ, δηλαδή έξω από το component scrollbar, πρέπει να προσδιορίσουμε ότι θέλουμε να εκτελεστούν στο συγκεκριμένο καρέ, στο συγκεκριμένο scene που βρισκόμαστε στην κυρίως ταινία, δηλαδή στη γονική ταινία. Δηλαδή η κυρίως ταινία αυτή τη στιγμή είναι η γονική τανία για το «παιδί της», το component scrollbar.

Σαν ενέργειες του καρέ που περιέχει όσα προαναφέρθηκαν, προστίθεται η εντολή stop(); έτσι ώστε όταν η ταινία περάσει από αυτό το καρέ να σταματήσει και να μην συνεχίσει στο επόμενο. Επίσης για κάθε ένα movie clip στο οποίο φορτώνεται κάποιο εξωτερικό swf, διακρίνουμε την εντολή invisible._visible=false; Που σημαίνει ότι το σγκεκριμένο movie clip να μην είναι ορατό στον χρήστη.

Σε όλα τα frames αυτού του επιπέδου μπορούμε να διακρίνουμε το κουμπί που υπάρχει σε ολόκληρο το δεύτερο επίπεδο. Είναι εκείνο που εκτελεί την ενέργεια ο χρήστης να οδηγείται στο κεντρικό μενού δηλαδή στο πρώτο επίπεδο. Επίσης παρατηρούμε ακόμα ένα κουμπί διαφορετικό από το προηγούμενο. Αυτό εκτελεί την ενέργεια της μεταφοράς του χρήστη στο δεύτερο επίπεδο, δηλαδή στο μενού του κάθε κεφαλαίου. Επίσης κάθε καρέ αυτού του επιπέδου περιλαμβάνει δύο κουμπιά που το ένα στέλνει τον χρήστη στην επόμενη θεωρία και το άλλο τον επιστρέφει στο προηγούμενο καρέ. Οι κώδικες αυτών των κουμπιών είναι :

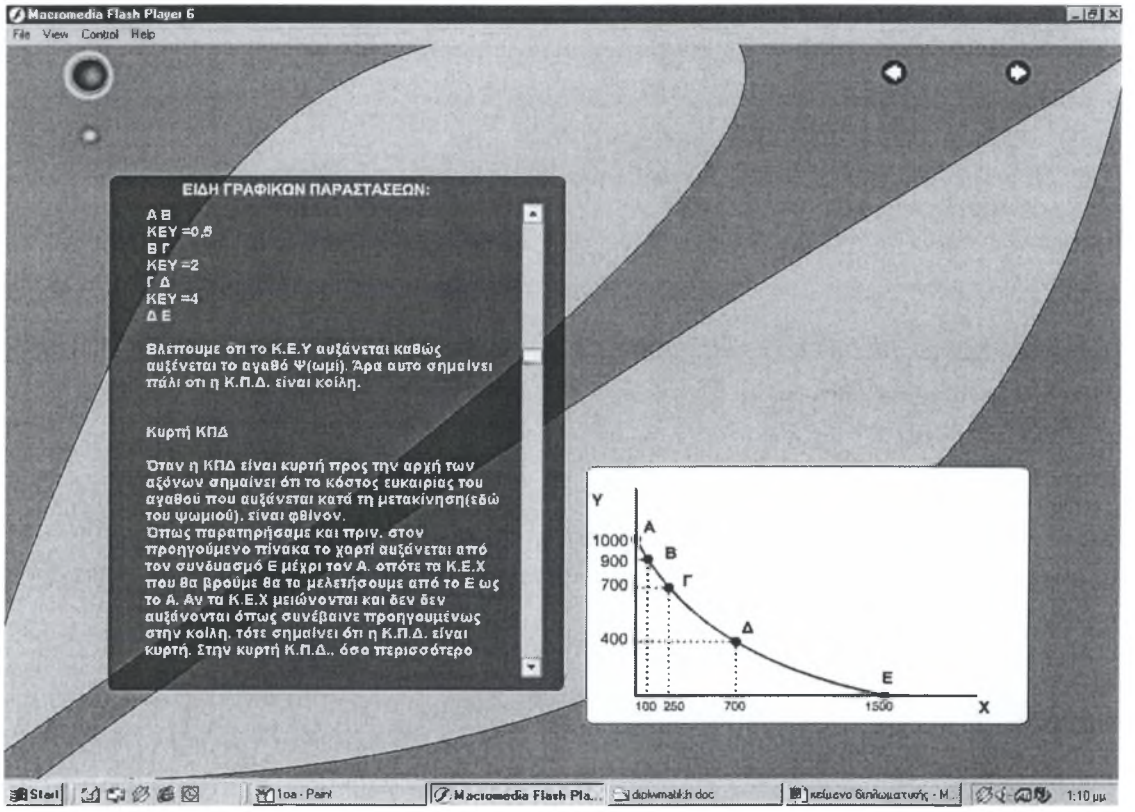
```
on (release)
{
prevFrame();
}
```

για το κουμπί που οδηγεί πίσω και,

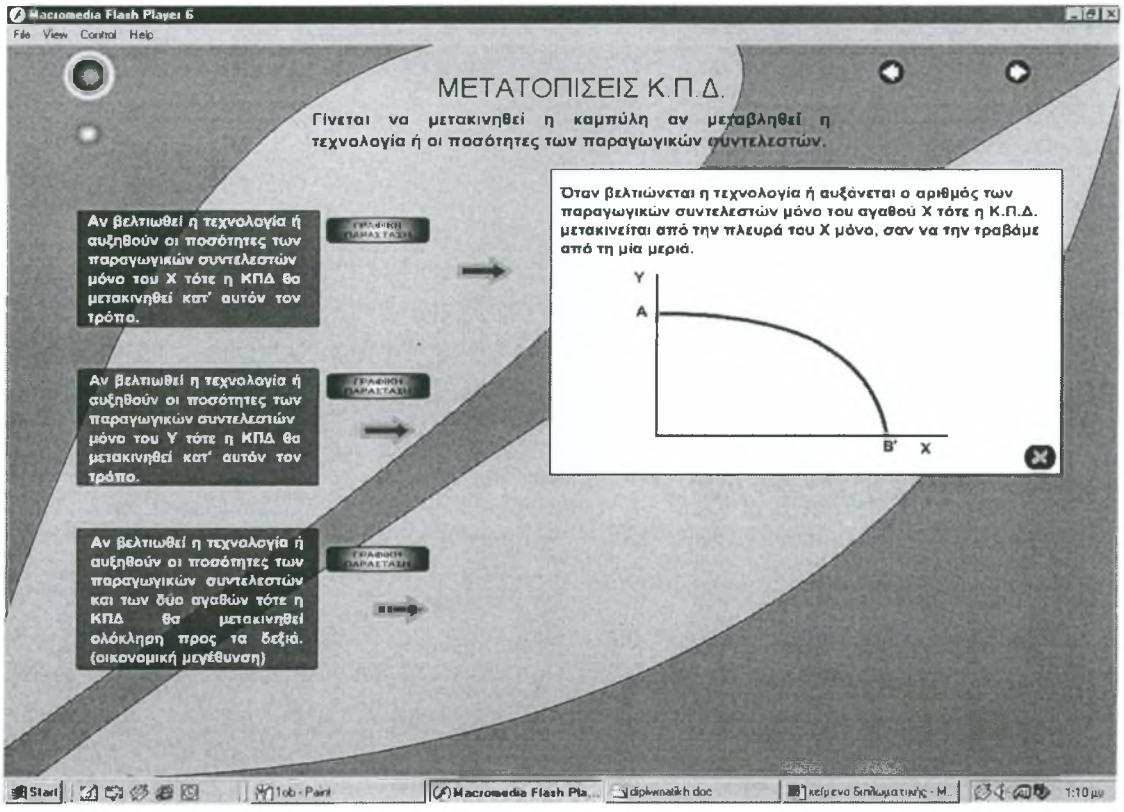
```
on (release)
{
nextFrame();
}
```

για το κουμπί που οδηγεί στο επόμενο καρέ.

Στο επόμενο καρέ του 1ου κεφαλαίου τοποθετήθηκε πάλι κείμενο στατικό αλλά και δυναμικό. Στην μπάρα κύλισης του δυναμικού πεδίου κειμένου ορίζονται κάποιες ενέργειες για την φόρτωση εξωτερικών αρχείων swf, αλλά επίσης δηλώνονται οι ενέργειες για την έναρξη φωνής όταν ο δείκτης της μπάρας κύλισης βρίσκεται σε κάποιες συγκεκριμένες θέσεις. Μόλις ο δείκτης φύγει από αυτές τις θέσεις, η φωνή θα σταματήσει.



Στο τρίτο καρέ του 1ου κεφαλαίου υπάρχουν κάποια στατικά πεδία κειμένου, τρία κουμπιά και κάποια movie clips. Τα κουμπιά μόλις πατηθούν φορτώνουν εξωτερικά swf που εμπεριέχουν clips και αναφέρονται στην θεματική ενότητα του κεφαλαίου.



Επειδή τα κουμπιά είναι τρία, ανοίγουν και τον αντίστοιχο αριθμό swf, οπότε υπήρχε το πρόβλημα να μην παίζει το ένα πάνω στο άλλο. Αν είναι φορτωμένο ένα swf τότε πατώντας ένα άλλο κουμπί από τα τρία, αυτομάτως κλείνει το προηγούμενο swf, και ανοίγει το επόμενο. Για να εκτελείται αυτό, κάθε ένα από τα τρία κουμπιά περιέχει τον κώδικα:

unloadMovie("όνομα του movie clip στο οποίο φορτώνεται"); και αφορά το swf που πρόκειται να κλείσει, και

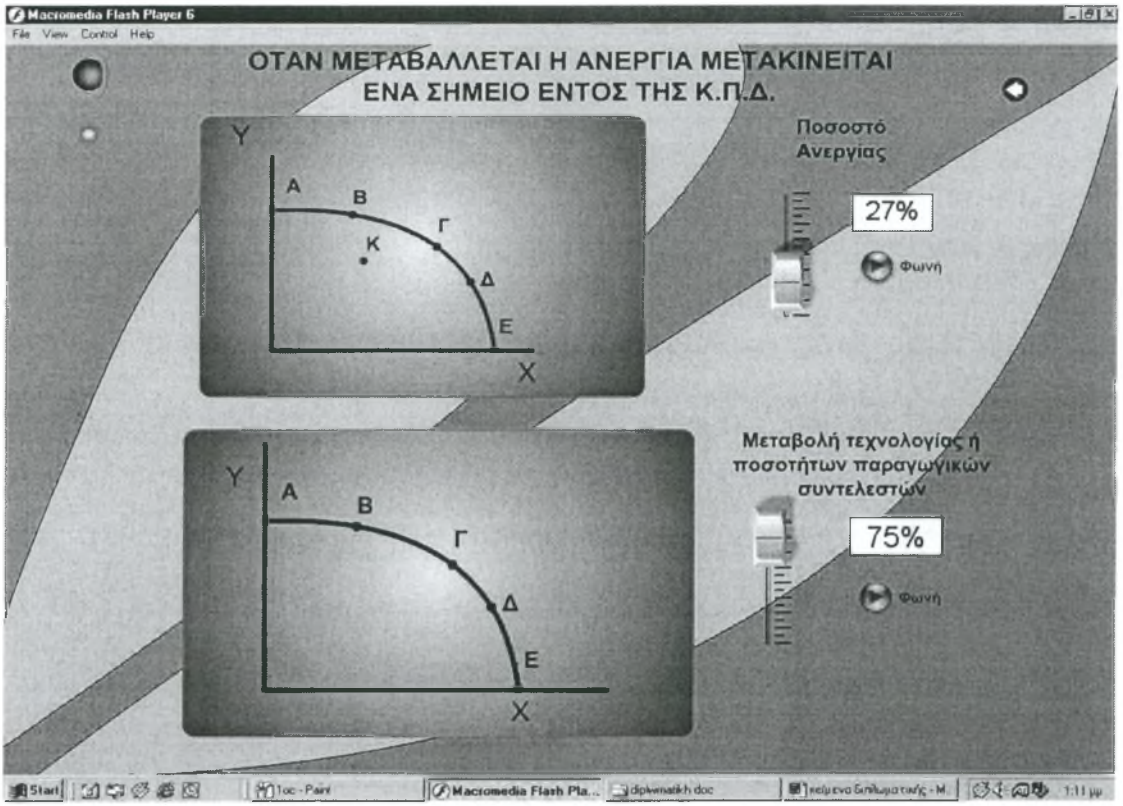
loadMovie("αρχείο.swf",όνομα movie clip); που αφορά το swf που θα φορτωθεί.

Κάθε φορά που ο δείκτης του ποντικιού βρίσκεται πάνω από το κουμπί τότε το movie clip που εμπεριέχει ένα βέλος, αρχίζει και παίζει με συνέπεια το βέλος να μετακινείται. Η εντολή για την εκτέλεση της ενέργειας βρίσκεται μέσα στο κουμπί και είναι

arrow1.gotoAndStop(8);

Δηλαδή η ενέργεια είναι να παίξει το καρέ υπ' αριθμόν 8 του movie clip «arrow1».

Το τέταρτο καρέ του 1^{ου} κεφαλαίου περιέχει δύο γραφικές παραστάσεις και δύο κουμπιά κύλισης. Μόλις ο χρήστης φέρει τον δείκτη του ποντικιού πάνω από το κυλιόμενο κουμπί, αρχίζει η αντίστοιχη επεξηγηματική φωνή.



Ο κώδικας για την εκτέλεση της εντολής είναι:

```
anerg.vertFader.butto.onRollOver=function()
{
        int(fl);
        if(fl!==1)
        {
        kpdanergia.start();
        fl=1;
        }
}
```

η συνάρτηση περιλαμβάνει τη μεταβλητή fl και ορίζει ότι η φωνή θα παίξει μόνο μια φορά.

Αν ο δείκτης βρεθεί έξω από το κουμπί τότε η φωνή σταματά αλλά εμφανίζεται ένα κουμπί που δίνει τη δυνατότητα στο χρήστη, αν θέλει να ξανακούσει την φωνητική επεξήγηση. Το κουμπί της φωνητικής επεξήγησης γίνεται ορατό με την εκτέλεση της εντολής

```
anerg.vertFader.butto.onRollOut=function()
{
        kpdanergia.stop();
        fwnh._visible=true;
}
```

η πρώτη γραμμή ορίζει την διαδρομή του αντικειμένου πάνω στο οποίο μας ενδιαφέρει να βρίσκεται ο δείκτης του ποντικού. Το συγκεκριμένο κουμπί με το όνομα «butto» είναι μέρος του movie clip «vertFader» κι αυτό με τη σειρά του είναι μέρος του movie clip «anerg». Οπότε μόλις ο δείκτης του ποντικιού εξέλθει από το

κουμπί, η σταματάει το φωνητικό αντικείμενο «kpdanergia» και εμφανίζεται το κουμπί «fwnh».

Όταν ο χρήστης μετακινεί το κυλιόμενο κουμπί, τότε μετακινείται και ένα σημείο στην αντίστοιχη γραφική παράστση. Αυτό εξηγείται από την εκτέλεση μιας σειράς ενεργειών που έχουν να κάνουν με συντεταγμένες.

```
onClipEvent (load)
{
inity=_y;
        left = _x;
        right = _x;
        top = _y-50;
        bottom = _y+50;
}

onClipEvent (enterFrame)
{
_root.circle._x = _y*(-1)+218;
_root.circle._y = _y+196;
_root.kappa._x = _y*(-1)+221;
_root.kappa._y = _y+175;
_root.pointer.text = _y+50 + "%";
}
```

Η πρώτη γραμμή ορίζει ότι με το που φορτώνεται το movie clip, να οριστούν οι συντεταγμένες του κουμπιού μέσα στο movie clip. Ο κώδικας μέσα στις καμπύλες ορίζει ότι το κουμπί θα έχει την ίδια τετμημένη x δηλαδή δεν θα μπορεί να μετακινηθεί δεξιά ή αριστερά. Όσον αφορά την τεταγμένη y, το κουμπί θα μπορεί να μετακινείται μέχρι κατά 50 θέσεις προς τα πάνω ή μέχρι κατά 50 θέσεις προς τα κάτω. Οι θέσεις αυτές αντιστοιχούν με τον αριθμό pixels, δηλαδή το κουμπί θα μπορεί να μετακινηθεί μέχρι 50 pixels προς τα κάτω ή μέχρι 50 pixels προς τα πάνω. Έπειτα η εντολή onClipEvent (enterFrame) έχει εξηγηθεί, και η πρώτη γραμμή μέσα στον βρόγχο δηλώνει ότι το movie clip «circle» που βρίσκεται στην κεντρική ταινία (_root) θα έχει τετμημένη , την τεταγμένη του κουμπιού πολλαπλασιασμένη με το -1 συν 218. Επίσης ορίζεται και η τεταγμένη του «circle» η οποία θα παίρνει σαν τιμή, την τεταγμένη του κουμπιού συν 50 pixels. Εκτός από το σημείο, ορίζονται και οι συντεταγμένες του γράμματος (που είναι ένα κλιπ) που αντιπροσωπεύει το σημείο, με τον ίδιο τρόπο. Η τελευταία γραμμή του κώδικα ορίζει ότι το δυναμικό πεδίο κειμένου «pointer», θα λαμβάνει σαν κείμενο την τεταγμένη του κουμπιού συν 175.

Το άλλο κυλιόμενο κουμπί εκτελεί λίγο διαφορετικές εντολές. Αυτό όταν μετακινείται πάνω ή κάτω, η γραφική παράσταση μεγαλώνει ή μικραίνει αντίστοιχα επεξηγώντας την ανάλογη θεωρία του κεφαλαίου. Ανάλογα παίζει και η επεξηγηματική φωνή.

```
onClipEvent (enterFrame) {
_root.kpdscaling._x = _y*(1.25)+227;
_root.kpdscaling._y = (_y+50)*(-1.0)+531;
```

```
_root.kpdscaling._xscale = (_y+50)*(1.3);
_root.kpdscaling._yscale = (_y+50)*(1.37);
_root.pointer2.text = _y*2 + "%";
}
```

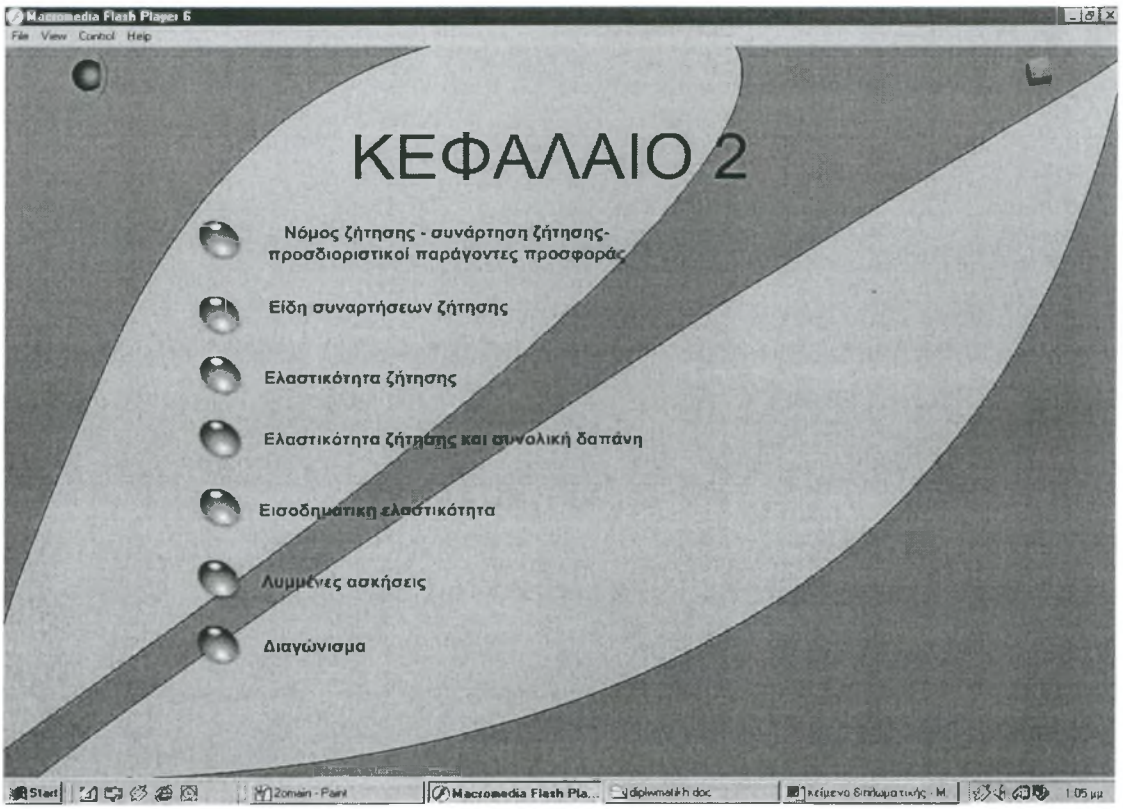
εδώ ο κώδικας είναι όμοιος με τον προηγούμενο, εκτός από την τρίτη και τέταρτη γραμμή μέσα στον βρόγχο. Στην τρίτη γραμμή ορίζεται ότι το movie clip «kpdscaling» που βρίσκεται στην κυρίως ταινία, θα αλλάζει μέγεθος οριζοντίως, τόσο όσο είναι και η τιμή της τεταγμένης του κουμπιού, συν 50 και πολλαπλασιασμένα επί το 1,3. Όμοια θα αλλάζει μέγεθος καθέτως όσο είναι η τιμή της τεταγμένης του κουμπιού, συν 50 και πολλαπλασιασμένα επί 1,37. Μ' αυτόν τον τρόπο όσο αυξάνεται η τιμή της τεταγμένης του κουμπιού θα μεγενθύνεται οριζοντίως και καθέτως το movie clip «kpdscaling», και τούμπαλιν. Κατα τα άλλα οι λοιπές εντολές εκτελούν τις ίδιες ενέργειες με το άλλο κυλιόμενο κουμπί. Και στα δυο κυλιόμενα κουμπιά διακρίνουμε τον κώδικα:

```
on (press) {
startDrag("", false, left, top, right, bottom);
_root._root.kappa._visible = false;
}
on (release) {
stopDrag();
_root._root.kappa._visible = true;
}
```

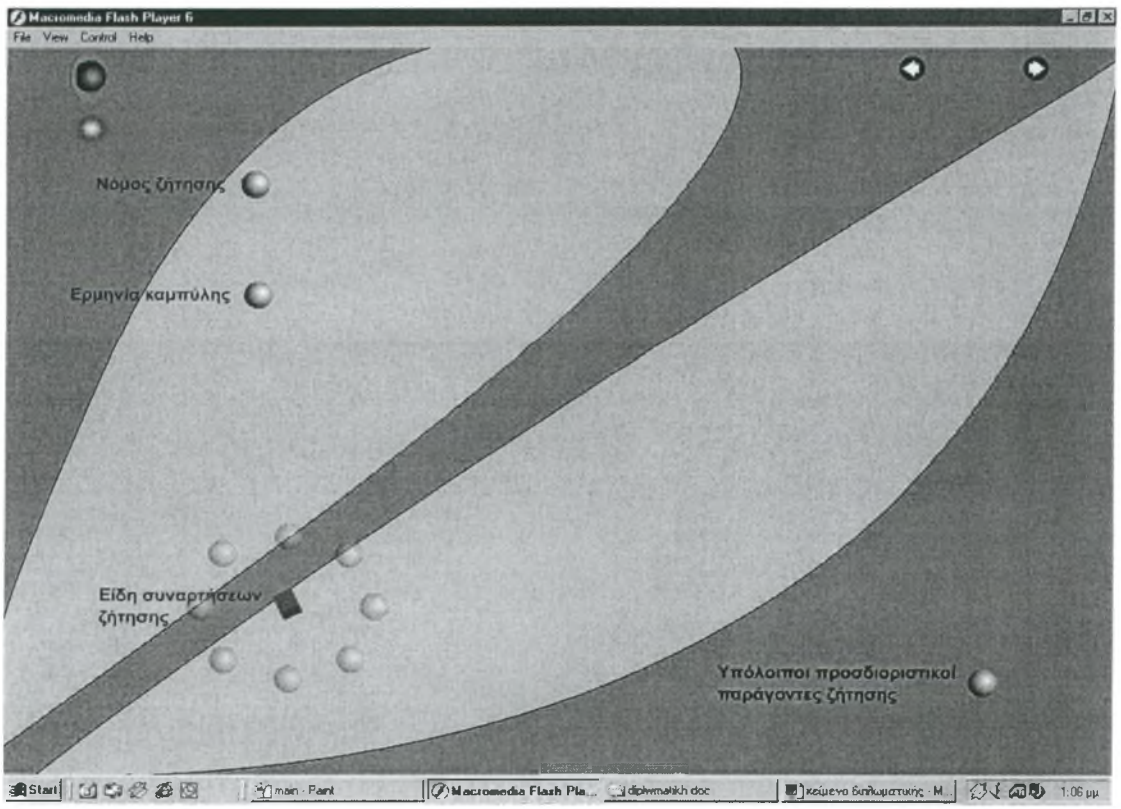
η πρώτη γραμμή μέσα στον βρόχγο της εντολής on (press) καθιστά το κουμπί ικανό να σύρεται δεξιά, αριστερά, πάνω, και κάτω. Επίσης κάθε φορά που το κουμπί θα κυλιέται, τότε το movie clip «kappa» θα γίνεται αόρατο και όταν το κουμπί θα ακινητοποιείται, το movie clip θα ξαναεμφανίζεται.

2° κεφάλαιο:

Αφού ο χρήστης περάσει από το εισαγωγικό επίπεδο του κειμένου, όπου θα μπορεί να διαλέξει την θεματική ενότητα που θέλει να μελετήσει, προχωράμε στα καρέ του κεφαλαίου.

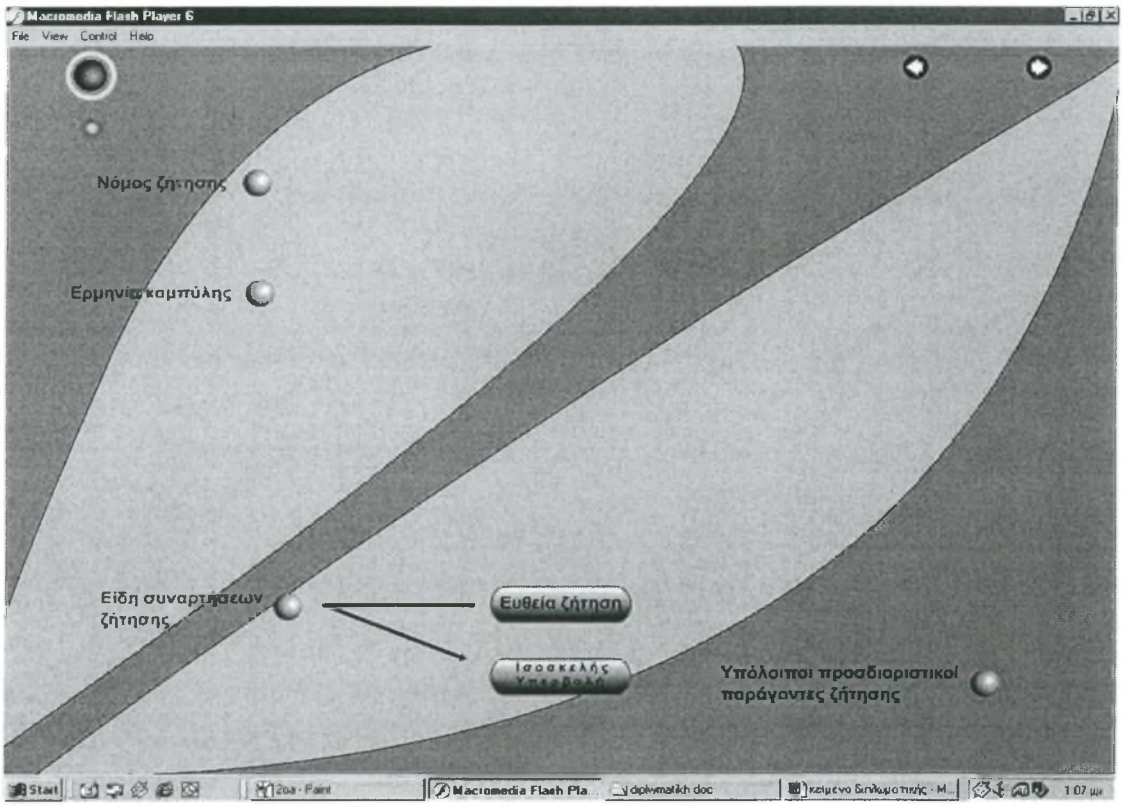


Το δεύτερο κεφάλαιο αρχίζει με την παρουσίαση κάποιων κουμπιών που με το πάτημά τους ανοίγουν εξωτερικά swf αρχεία. Κάθε ένα κουμπί αντιστοιχεί και σε κάποια θεωρία του βιβλίου.

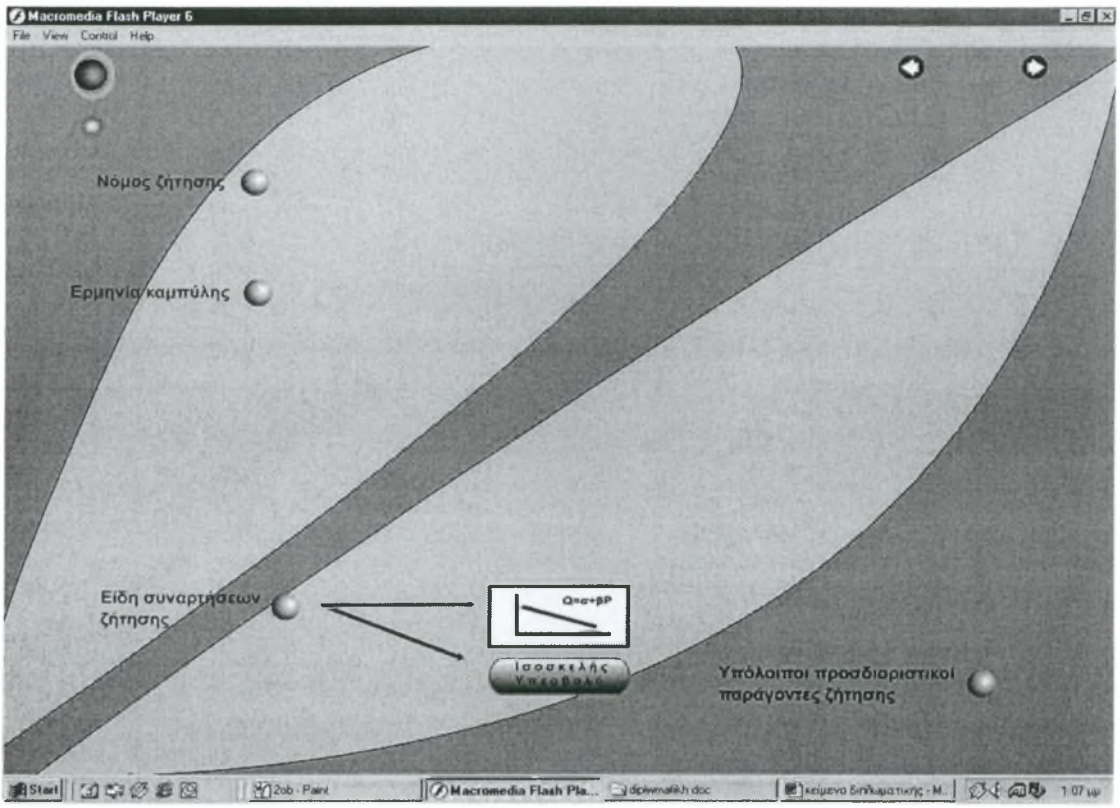


Με το πάτημα Με το πάτημα του κουμπιού «Είδη συναρτήσεων ζήτησης», παίζει ένα movie clip προκειμένου να προκαλέσει εντύπωση και να θυμηθεί ο

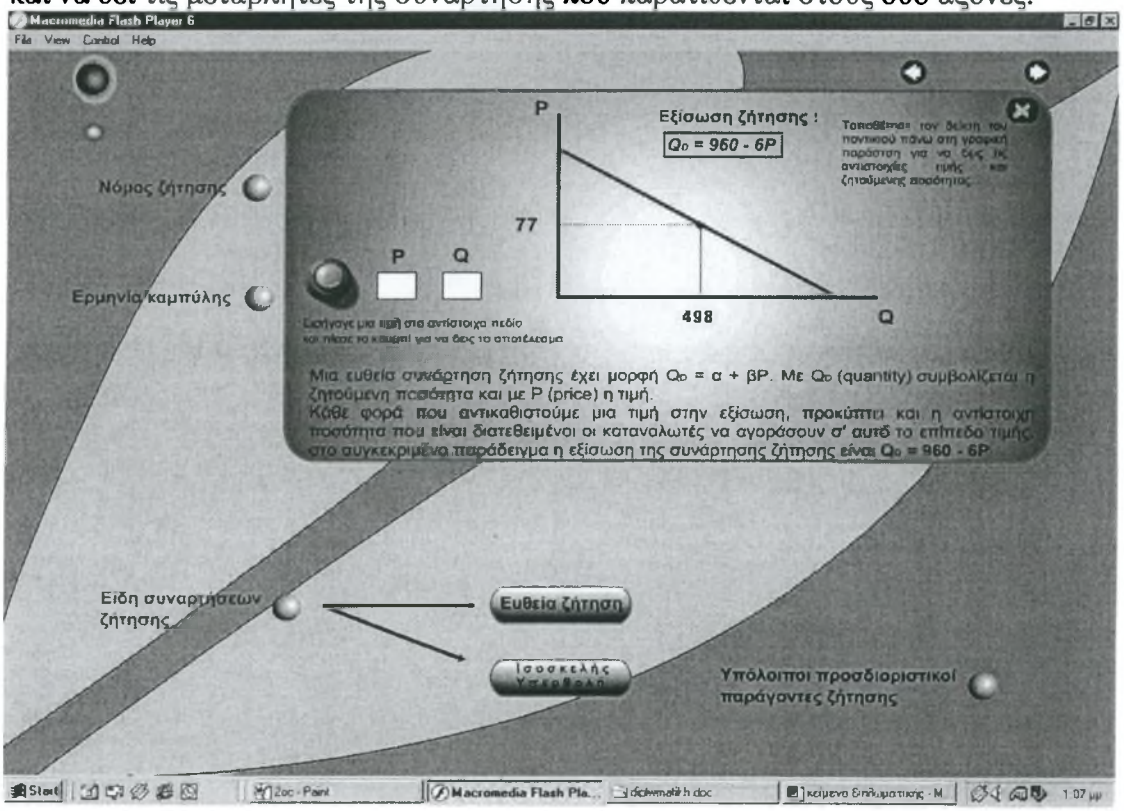
μαθητής την αντίστοιχη θεωρία. Έπειτα όπως παρατηρούμε και στην επόμενη εικόνα, εμφανίζονται δύο κουμπιά – movie clips που με το πάτημα του καθενός εμφανίζονται τα αντίστοιχα clips.



Όπως φαίνεται παρακάτω, αν ο χρήστης βάλει τον δείκτη του ποντικιού πάνω από ένα από αυτά τα καινούρια κουμπιά, τότε αυτά αλλάζουν σχήμα και μεταμορφώνονται σε μικρές γραφικές παραστάσεις, όπου η καθεμιά αντιπροσωπεύει την αντίστοιχη προσομοίωση.



Τα συγκεκριμένα κουμπιά ανοίγουν κάποια clips τα οποία αποτελούν αποτελούν προσομοιωτές της θεωρίας. Όπως βλέπουμε στην εικόνα παρακάτω, ο χρήστης έχει τη δυνατότητα να τοποθετήσει το ποντίκι πάνω στη γραφική παράσταση και να δει τις μεταβλητές της συνάρτησης που παρατίθενται στους δύο άξονες.



Καθώς θα κουνά το ποντίκι, οι γραμμές που δείχνουν τις συντεταγμένες θα μετακινούνται ακολουθώντας το ποντίκι. Επίσης θα μεταβάλλονται και οι τιμές που θα παίρνουν οι μεταβλητές ανάλογα με το ποντίκι. Το ότι οι γραμμές που δείχνουν οι συντεταγμένες, θα ακολουθούν το ποντίκι ελέγχονται από ένα αόρατο movie clip ονόματι «controlMC» το οποίο εμπεριέχει μόνο κώδικα. Αυτός ο κώδικας είναι:

```
Xmouse = getProperty("../mouse", _x);
Ymouse = getProperty("../mouse", _y);
setProperty("../jen", _y, Ymouse);
setProperty("../dva", _x, Xmouse);
```

στην πρώτη και δεύτερη γραμμή ορίζονται οι μεταβλητές Xmouse και Ymouse όπου η καθεμιά λαμβάνει σαν τιμές την τετμημένη και τεμημένη του δείκτη του ποντικιού αντίστοιχα. Στην δεύτερη γραμμή η μεταβλητή Ymouse ορίζεται σαν ιδιότητα τεταγμένης του movie clip «jen». Το movie clip «jen» είναι η γραμμή που δείχνει την τεταγμένη στην γραφική παράσταση. Η μεταβλητή Xmouse ορίζεται σαν ιδιότητα τεταγμένης του movie clip «dva». Το movie clip «dva» είναι η γραμμή που δείχνει την τετμημένη στην γραφική παράσταση. Έπειτα στα καρέ της στρώσης «actions» παρατίθεται ο κώδικας

```
stop ();
dva._visible=false;
jen._visible=false;
mouse._visible=false;
timi._visible=false;
posotita._visible=false;
lathos._visible=false;
swsto._visible=false;
invprosoxi._visible=false;

d.onRollOver = function()
{
startDrag ("mouse", true);
}
d.onRollOut= function()
{
stopDrag ();
}
d.onPress= function()
{
stopDrag ();
}
```

στο πρώτο καρέ του movie clip έχουμε την εντολή να σταματήσει η ταινία σ' αυτό το καρέ. Επίσης ορίζεται μια πλειάδα από movie clips να μην είναι αόρατα. Αυτά θα εμφανιστούν στο δεύτερο καρέ ή μόλις το ποντίκι βρεθεί πάνω στη παράσταση. Μετά από αυτά παρατίθεται κώδικας που ορίζει τρεις συναρτήσεις. Η πρώτη ορίζει ότι μόλις το ποντίκι βρεθεί εντός πεδίου του movie clip «d» τότε να θεωρηθεί το ποντίκι κυλιόμενο και να λαμβάνονται οι τιμές του, προκειμένου να μετακινούνται και οι συντεταγμένες. Η δεύτερη και τρίτη συνάρτηση ορίζουν να παύει αυτό το γεγονός

μόλις το ποντίκι εξέρχεται από το movie clip. Κάθε μιά από τις γραμμές που δείχνουν τις συντεταγμένες έχει κώδικα. Της τεταγμένης είναι:

```
onClipEvent(enterFrame)
{
        _parent.timi._y=_y-11;
        _parent.timi.text=int(_y*(-1.3)+220.6);
}
```

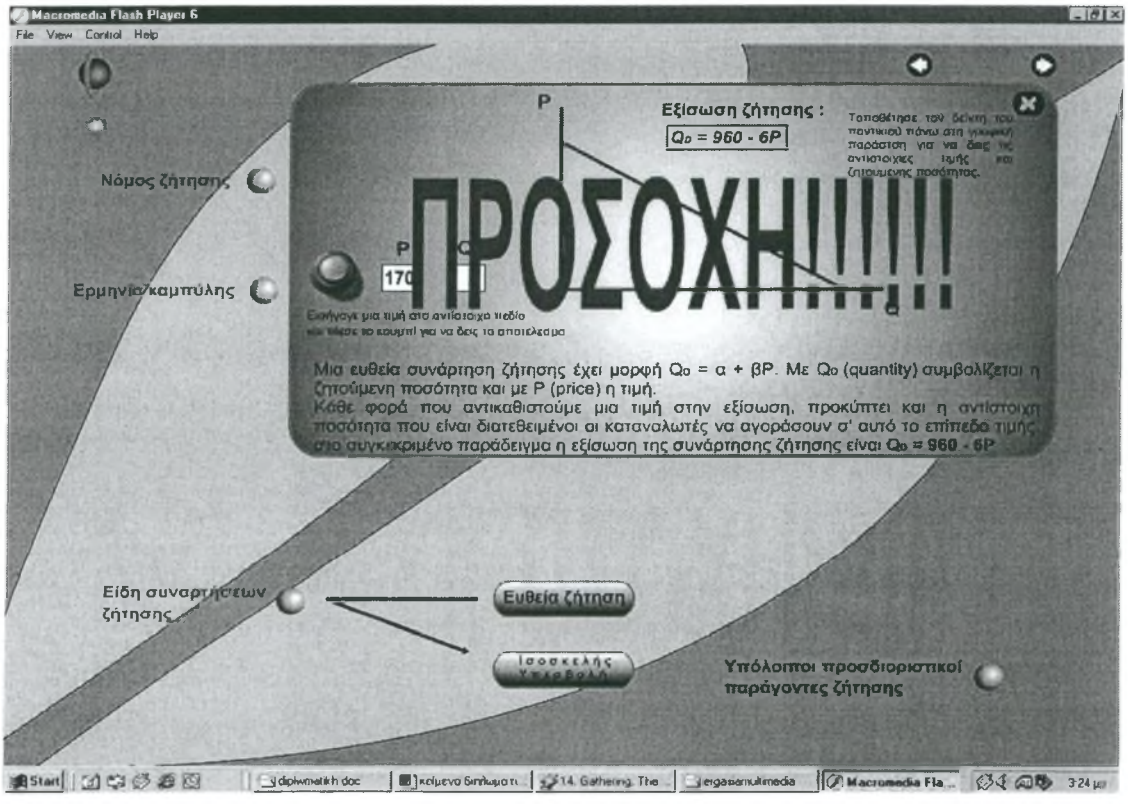
μέσα στον βρόγχο ορίζεται ότι το δυναμικό πεδίο κειμένου θα έχει τεταγμένη ίση με την τεταγμένη της γραμμής «jen», και θα εμφανίζει ως περιεχόμενο την τεταγμένη του επί το -1,3 συν 220,6. Αυτή θα είναι και η τιμή της μιας εκ των δύο μεταβλητών που υπάρχουν στο παράδειγμα. Ο κώδικας της τετμημένης είναι όμοιος:

```
onClipEvent(enterFrame)
{
        _parent.posotita._x=_x-16.5;
        _parent.posotita.text=960-6*int(_parent.jen._y*(-1.3)+220.6);

}
```

σ' αυτό το swf ο χρήστης επίσης έχει τη δυνατότητα αντί να μετακινεί το ποντίκι, να τυπώσει την τιμή που θέλει στο πεδίο κειμένου που αντιπροσωπεύει την μια μεταβλητή, και η γραφική παράσταση να προσαρμοστεί στην τιμή της συγκεκριμένης μεταβλητής. Ο κώδικας που χρησιμοποιείται φαίνεται παρακάτω.

```
on (release)
{
        int (b) ;
        if (Number(_parent.m)>=0 && Number(_parent.m)<161)
        {
        b= (Number(_parent.m)-220.6)/(-1.3);
        _parent.mouse._y=b;
        _parent.mouse._x=127+1.79*b;
        _parent.posotita2.text=960-6*Number(_parent.m);
        _parent.mouse._visible=true;
        _parent.jen._visible=true;
        _parent.dva._visible=true;
        _parent.timi._visible=true;
        _parent.posotita._visible=true;
        _parent.lathos._visible=false;
        _parent.swsto._visible=true;
        }
else    {
        _parent.swsto._visible=false;
        loadMovie("prosoxi.swf",_parent.invprosoxi);
        _parent.gotoAndPlay(2);
        }
}
```

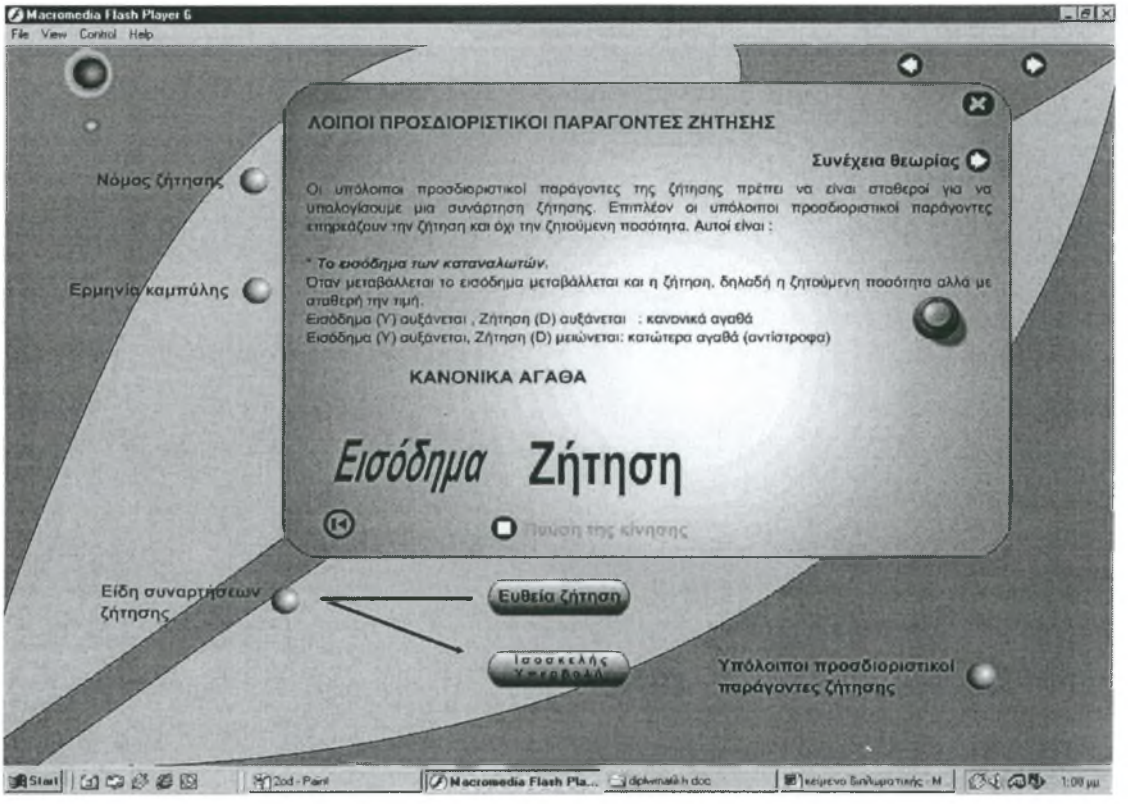
με το πάτημα του κουμπιού ορίζεται μια μεταβλητή ονόματι «b». Το πεδίο κειμένο στο οποίο ο χρήστης εισάγει τιμές, μεταβιβάζει τον αριθμό που εισάγει ο χρήστης στη μεταβλητή «m». Με το πάτημα του κουμπιού, λαμβάνεται αυτή η μεταβλητή και μετατρέπεται σε αριθμό. Αν αυτός ο αριθμός είναι μεγαλύτερος ή ίσος του μηδενός και μικρότερος του 161, τότε η μεταβλητή «b» παίρνει την τιμή της m, την αφαιρεί με το 220,6 και τη διαφορά την πολλαπλασιάζει με το -1,3. Έπειτα ορίζεται η τεταγμένη του ποντικιού να γίνει ίση με την τιμή της μεταβλητής «b», και η τετμημένη ίση με 127+1.79*b. Επίσης ορίζεται η τιμή του δυναμικού πεδίου κειμένου να γίνει ίση με 960-6*(την τιμή της μεταβλητής «m»). Έτσι και οι γραμμές ακολουθώντας τις συντεταγμένες που ορίστηκαν για το ποντίκι, λαμβάνουν τη σωστή θέση. Ακόμα, με το πάτημα του κουμπιού ορίζεται να εμφανιστούν οι γραμμές των συντεταγμένων της γραφικής παράστασης, τα πεδία κειμένου που αναφέρουν τις τιμές των μεταβλητών του παραδείγματος, και ένα επεξηγηματικό πεδίο κειμένου. Παρακάτω ορίζεται τι να γίνει αν η τιμή που εισάγει ο χρήστης δεν είναι μεταξύ του 1 και 161. Σ' αυτή την περίπτωση κρύβεται το επεξηγηματικό πεδίο κειμένου και φορτώνεται ένα εξωτερικό swf, με ένα clip που δημιουργεί εντύπωση, προκειμένου ο χρήστης να μην ξανακάνει το ίδιο λάθος. Επίσης η ταινία προχωράει στο 2^{ο} καρέ και η ταινία διανύει κάποια καρέ για να παίξει το swf, του λάθους, και έπειτα εμφανίζεται ένα πεδίο κειμένου που εξηγεί το λάθος του χρήστη. Αυτά φαίνονται στην εικόνα που παρατίθεται παρακάτω.



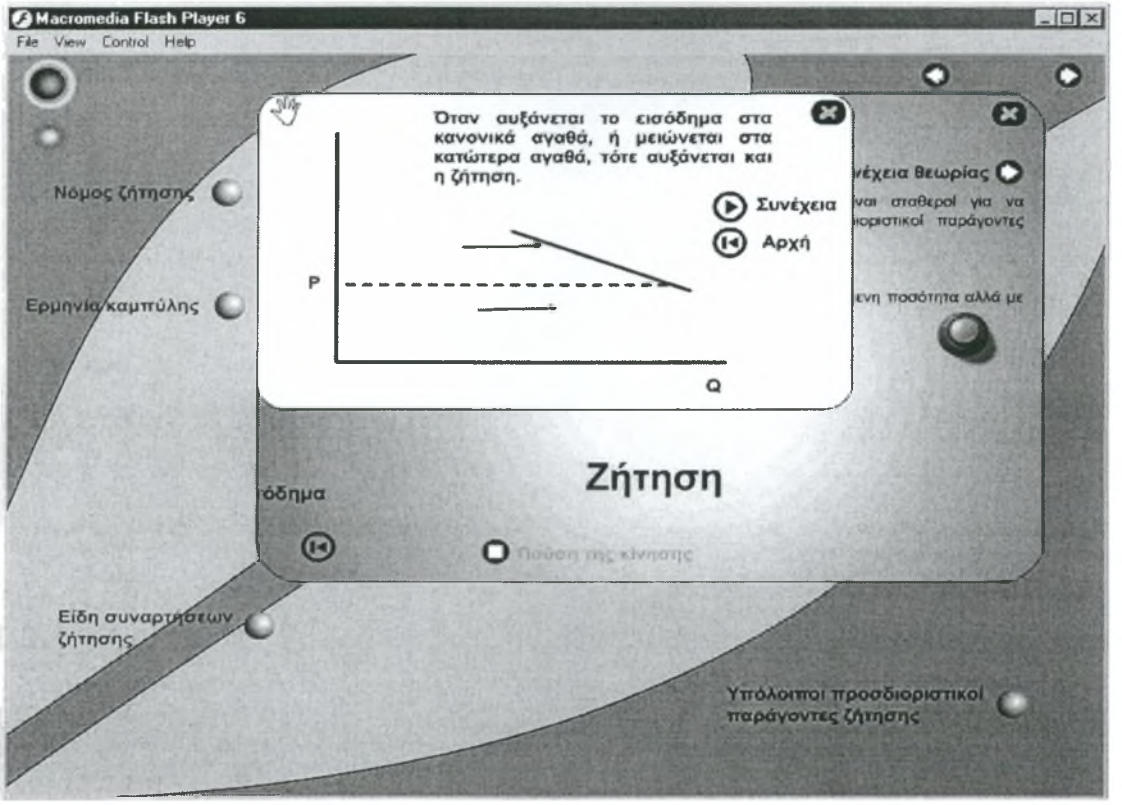
Αυτό το clip θα παίξει επίσης όταν ανοίγει το clip της ισοσκελούς υπερβολής και εκεί ο χρήστης τοποθετήσεις στο αντίστοιχο πεδίο της τιμής, την τιμή μηδέν. Τότε επειδή η τιμή είναι παρονομαστής στην συγκεκριμένη συνάρτηση, δεν μπορεί να πέρει την τιμή μηδέν, γι' αυτό και βγαίνει το αντίστοιχο μήνυμα.

Επιστρέφοντας στην κύρια ταινία και στο δεύτερο καφάλαιο, να επισημάνουμε ότι τα ίδια συμβαίνουν όταν ανοίγει το εξωτερικό swf για το άλλο από τα δύο κουμπιά που εμφανίζονται στο 2^{ο} καρέ του 2^{ου} κεφαλαίου. Όπως παρατηρούμε

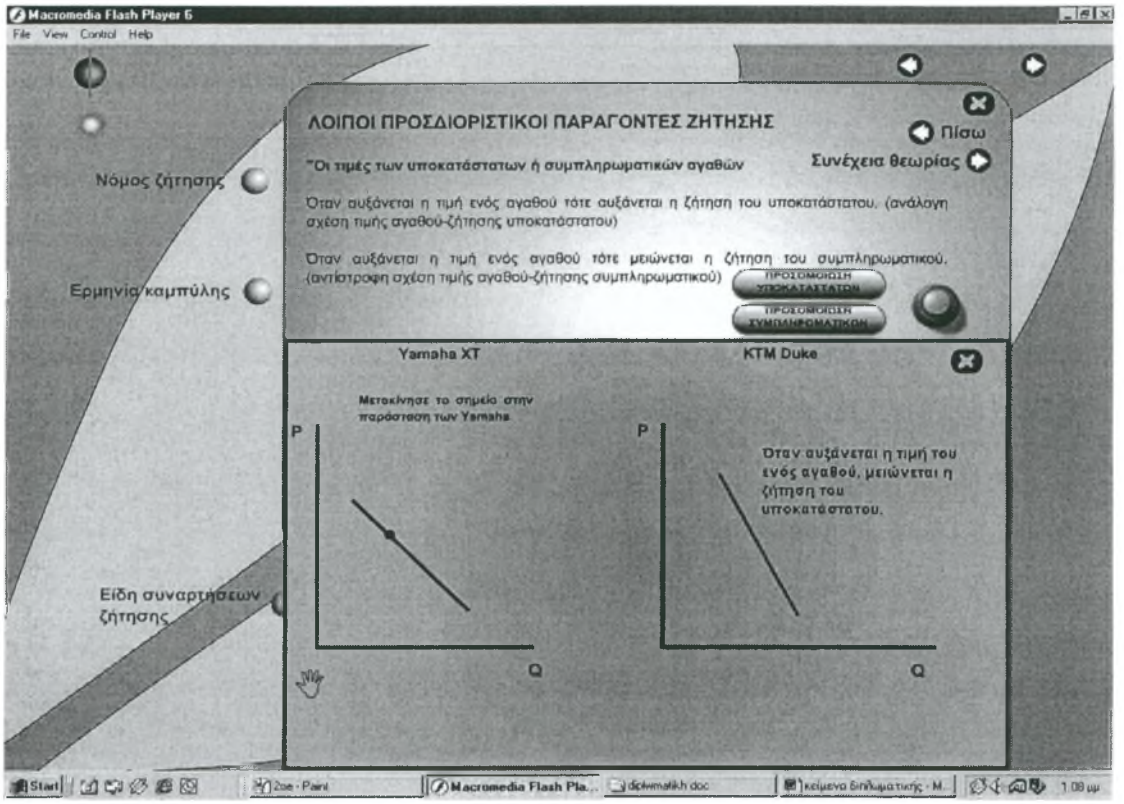
και στην εικόνα παρακάτω, πιέζοντας το κουμπί «Υπολοιποι προσδιοριστικοί παράγοντες ζήτησης» εμφανίζεται ένα πλαίσιο κειμένου που εμπεριέχει κάποια κινούμενα κείμενα που υποδηλώνουν τη σχέση που έχει ο συγκεκριμένος προσδιοριστικός παράγοντας με τη ζήτηση.



Σ' αυτό το clip υπάρχει κι ένα κουμπί που δείχνει διαγραμματικά, πως επιδρά ο συγκεκριμένος προσδιοριστικός παράγοντας στη ζήτηση.

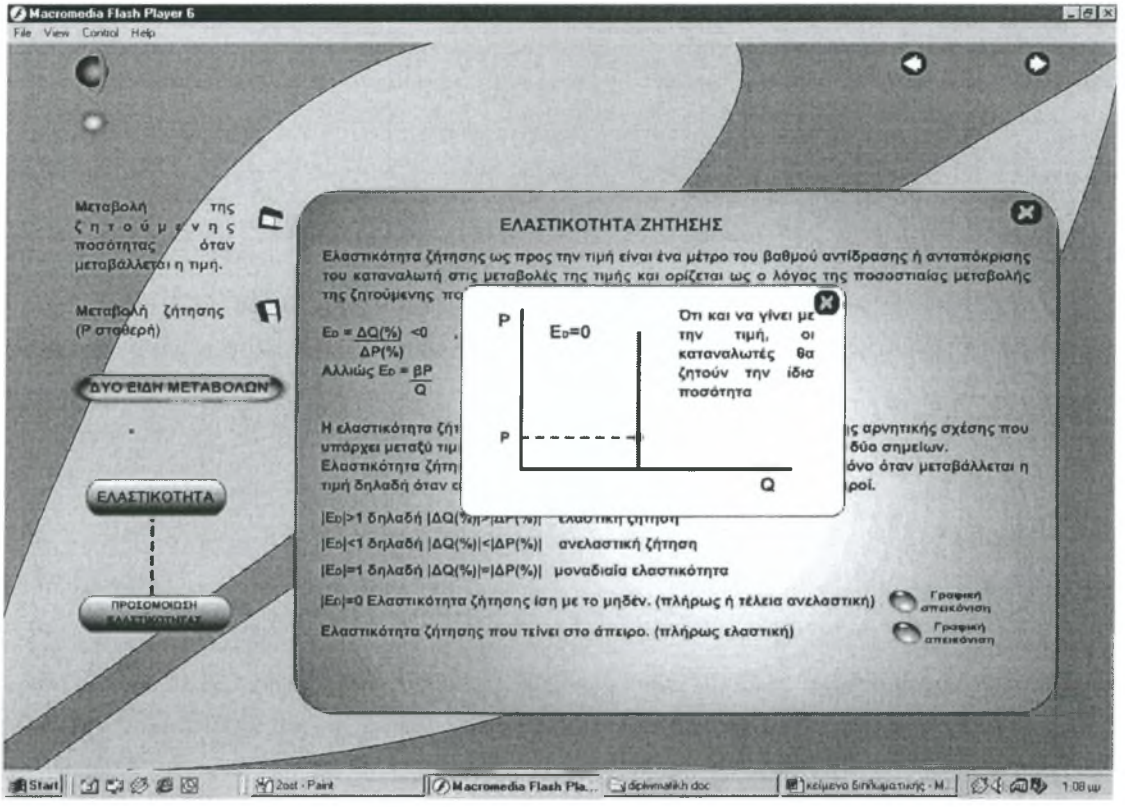


επειδή οι προσδιοριστικοί παράγοντες της ζήτησης είναι πέντε, στο νέο swf που έχει ανοίξει υπάρχει κουμπί με την δυνατότητα της συνέχισης για την παρατήρηση του επόμενου προσδιοριστικού παράγοντα ή του προηγούμενου. Ένας από αυτούς τους προσδιοριστικούς παράγοντες περιλαμβάνει δύο κουμπιά τα οποιά μόλις πατηθούν καλούν αντίστοιχα από ένα εξωτερικό swf, τα οποία εμπεριέχουν μια προσομοίωση της συγκεκριμένης θεωρίας όπως φαίνεται παρακάτω.

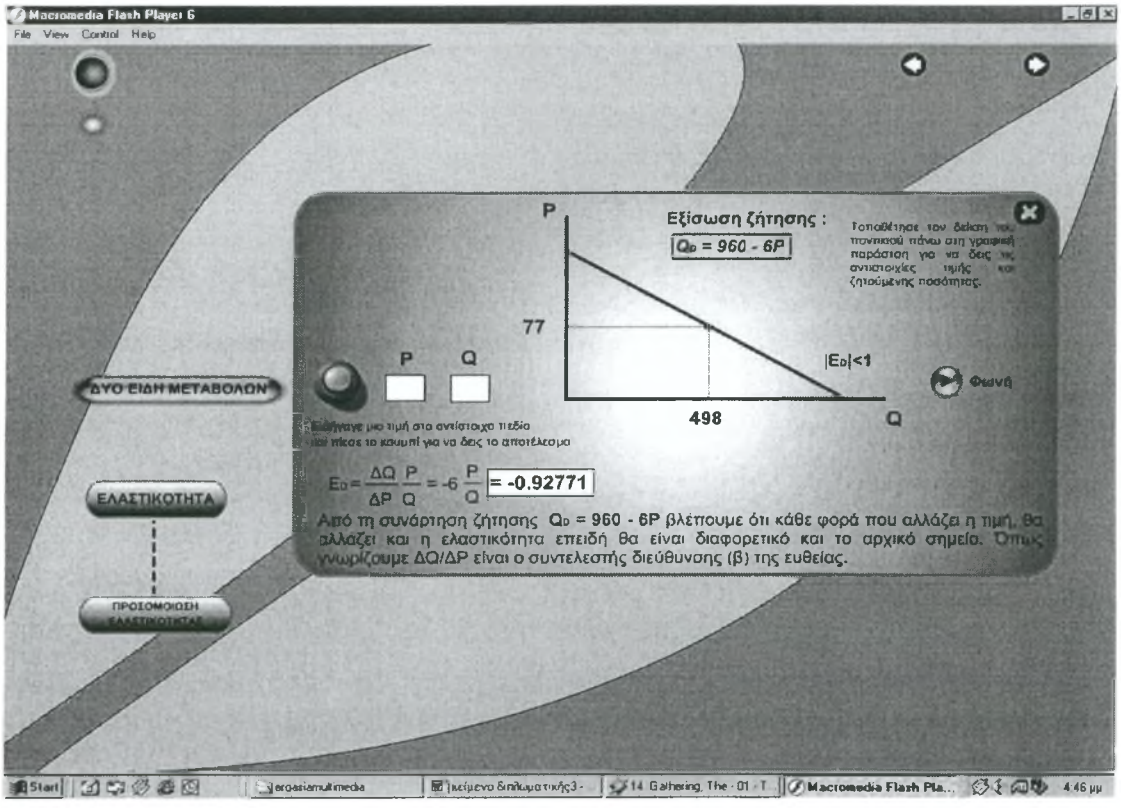


Ο χρήστης αν μετακινήσει το σημείο στην δεξιά καμπύλη, τότε αντίστοιχα θα μετακινείται και η δεξιά καμπύλη.

Στο επόμενο καρέ ένα από τα κουμπιά ανοίγει ένα εξωτερικό swf που εμπεριέχει κείμενο και κάποια κουμπιά που ανοίγουν επιπλέον κάποιες κινούμενες γραφικές παραστάσεις που επιδεικνύουν τη συγκεκριμένη θεωρία που ισχύει για την ελαστικότητα ζήτησης. Αυτά φαίνονται παρακάτω.



Επίσης ένα από τα κουμπιά φορτώνει ένα swf αρχείο διαφορετικό από τα προηγούμενα. Μοιάζει με το swf με τις συντεταγμένες αλλά έχει κάτι παραπάνω. Έχουν προστεθεί πεδία κειμένου που λαμβάνουν την τιμή που προκύπτει από έναν τύπο (το δυναμικό πεδίο κειμένου «ed») και επίσης εμφανίζονται ή εξαφανίζονται κάποια πεδία κειμένου –σύμβολα-, ανάλογα με το αν ο χρήστης βρίσκεται με το ποντίκι σε κάποιο σημείο της γραφικής παράστασης ή σε κάποιο άλλο.



Η διαφορά είναι ότι στο ίδιο το κουμπί προστέθηκε η εντολή _parent.a=1; που δίνει την τιμή 1 σε μια μεταβλητή a που έχει οριστεί, προκειμένου για κάποιους ελέγχους αν ο χρήστης κάνει λάθος, και επίσης η εντολή _parent.edsima.stop(); που ορίζει να σταματάει μια φωνητική επεξήγηση που δεν υπήρχε στο προηγούμενο swf που εξηγήθηκε, αν ο χρήστης κάνει λάθος. Μόλις ο χρήστης εισάγει το ποντίκι στην καμπύλη της γραφικής παράστασης, εκτός από τα προηγούμενα (που εξηγήθηκαν για το προηγούμενο swf), παίζει μια φορά η φωνή και με τη φυγή του χρήστη από τη γραφική παράσταση, εμφανίζεται ένα κουμπί για πιθανή επανάληψη της φωνής. Όταν ο χρήστης βρίσκεται πάνω στην καμπύλη, τότε ανάλογα με τη θέση του εμφανίζεται και το αντίστοιχο πεδίο κειμένου που εξηγεί τι συμβαίνει σ' εκείνο το σημείο που βρίσκεται ο χρήστης.

```
if(_parent.a==0)
        {
_parent.ed.text="="+-6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-
1.3)+220.6));
        }
        else
        {
        _parent.ed.text="";
        }
```

αν ο χρήστης δεν έχει κάνει λάθος, τότε η μεταβλητή a θα είναι μηδέν. Τότε όπως βλέπουμε μέσα στους βρόγχους, το νέο δυναμικό πεδίο κειμένου ονόματι «ed», θα παίρνει τιμές όπως ορίζει η σχέση -6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6 ειδάλλως αν ο χρήστης έχει κάνει λάθος, το πεδίο κειμένου γίνεται κενό μέχρι ο χρήστης να ξαναεισάγει μια σωστή τιμή.

```
if (-6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6))==-
infinity)
        {
        _parent.ed.text="= ΑΠΕΙΡΟ";
        }
```

αν ο χρήστης εισάγει μια τιμή ή το ποντίκι βρίσκεται σε κάποια θέση, όπου η σχέση -6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6 είναι ίση με το άπειρο, τότε ορίζεται το πεδίο κειμένο ed να αναγράφει τη λέξη "ΑΠΕΙΡΟ".

```
if (-6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6))==-1)
        {
                if(_parent.a==0)
                {
        _parent.edanel._visible=false;
        _parent.ed100._visible=false;
        _parent.meso._visible=true;

                }
        }
```

αν δεν υπαρχει σφάλμα από τον χρήστη, τότε αν ισχύει η σχέση -6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6))=-1 τότε κάποια κείμενα –clips να εμφανίζονται, ενώ άλλα να εξαφανίζονται. Αν ο χρήστης εισάγει λάθος τιμή στο πεδίο κειμένου που ελέγχεται απ' το κουμπί, τότε όλα τα κείμενα clips εξαφανίζονται. Όμοιες είναι οι εντολές και σε άλλες περιπτώσεις όπως να είναι -6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6))>-1 ή -6*int(_parent.jen._y*(-1.3)+220.6)/(960-6*int(_parent.jen._y*(-1.3)+220.6))<-1.

Ένα πρόβλημα που συναντήθηκε ήταν ότι μόλις άνοιγε ένα swf εξωτερικό, τα κουμπιά που υπήρχαν στην κυρίως ταινία ήταν ενεργά και μπορούσαν να δημιουργήσουν πρόβλημα. Έτσι αυτό αντιμετωπίστηκε με την εντολή kampylhtext._visible=false; Κι έτσι μόλις άνοιγε το εξωτερικό swf τα κουμπιά εξαφανιζόταν. Μόλις έκλεινε το εξωτερικό swf καθιστούσε ξανά ορατά τα συγκεκριμένα κουμπιά με την εντολή, _parent.kampylhtext._visible=false; όπου το πρόθεμα _parent ήταν απαραίτητο γιατί δινόταν μια εντολή από ένα εξωτερικό swf, που αφορούσε την κύρια ταινία.

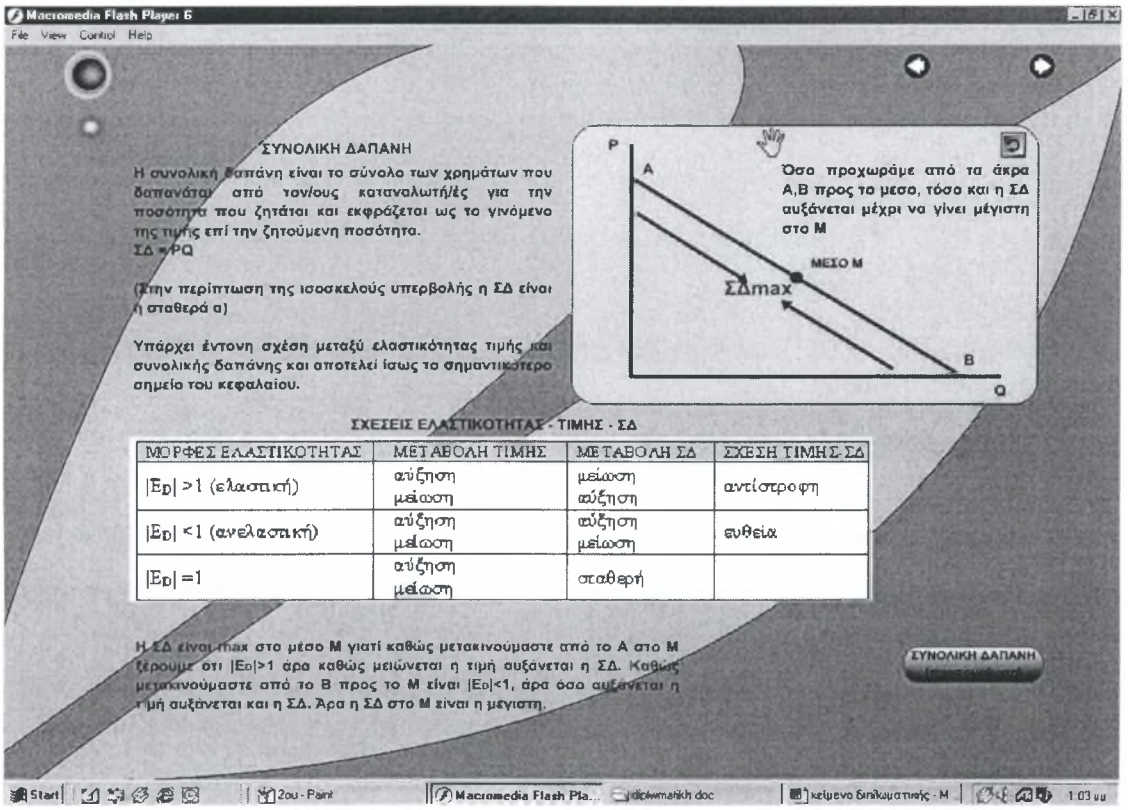
Κάποια άλλα clips διαφορετικά, αλλά μοιάζουν με του πρώτου κεφαλαίου τις προσομοιώσεις, βρίσκονται πάλι σε κάποια εξωτερικά swf, στα οποία ο χρήστης μπορεί να μετακινεί ένα σημείο πάνω σε μια γραφική παράσταση, και ταυτόχρονα να μετακινείται μια άλλη γραφική παράσταση αναλόγως. Ο κώδικας που χρησιμοποιείται γι' αυτό είναι :

```
onClipEvent (load)
{
inity=_y;
        left = _x;
        right = _x;
        top = _y-66;
        bottom = _y+66;
}
```

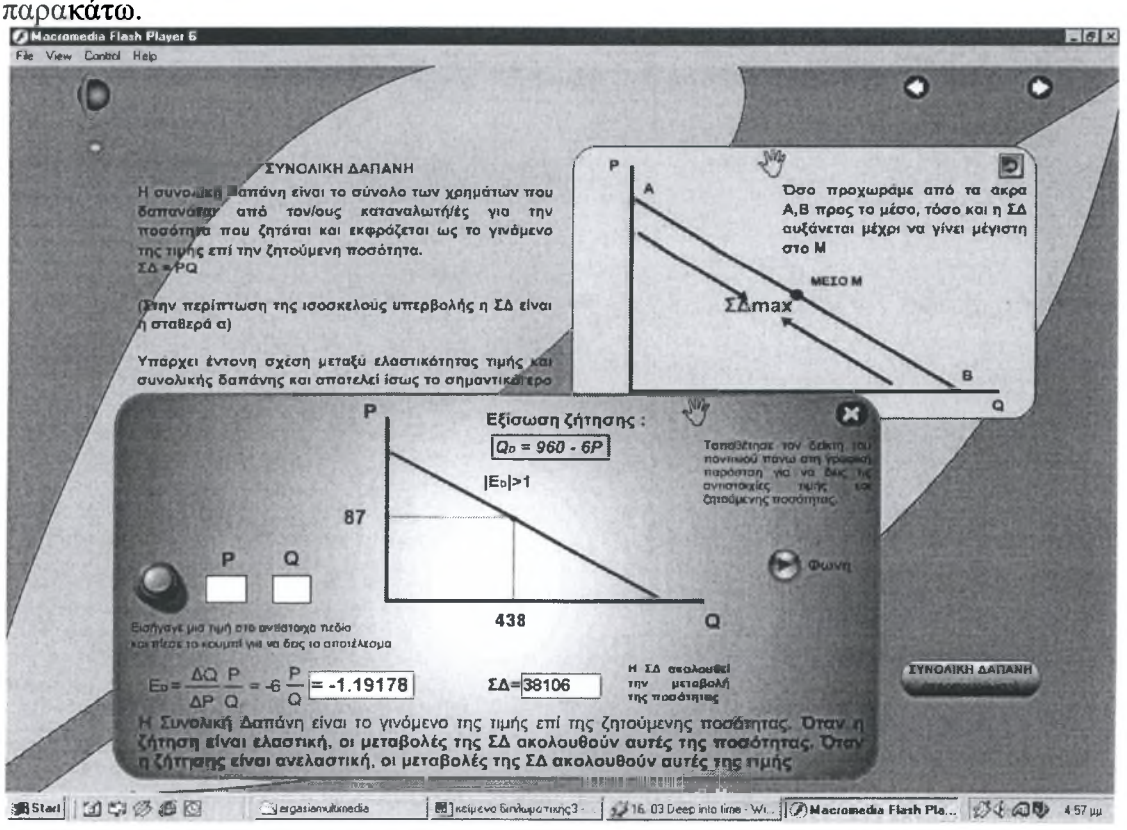
```
onClipEvent(enterFrame)
{
_parent._parent.lineclip.line._x=_y;
_parent._parent.lineclip.line._xscale=_y*0.5+100;
_parent._parent.lineclip.line._yscale=_y*(-0.25)+100;
        if(_y>1)
        {
                _parent._parent.lineclip.exig2.text="";
                _parent._parent.lineclip.exig.text="Όταν μειώνεται η τιμή ενός
αγαθού, αυξάνεται η ζήτηση του υποκατάστατου.";
        }
        if(_y<1&&_y>-1)
        {
                _parent._parent.lineclip.exig.text="";
                _parent._parent.lineclip.exig2.text="";
        }
        if(_y<-1)
        {
                _parent._parent.lineclip.exig2.text="Όταν αυξάνεται η τιμή ενός
αγαθού, μειώνεται η ζήτηση του υποκατάστατου.";
                _parent._parent.lineclip.exig.text="";
        }
}
```

πάλι πρόκειται περί ενός κυλιόμενου κουμπιού, το οποίο μπορεί να μετακινείται κατά 66 pixels πάνω ή κάτω. Έπειτα μέσα στον βρόγχο του onClipEvent(enterFrame) ορίζεται το clip line που εμπεριέχεται μέσα στο clip lineclip κι αυτό με τη σειρά του βρίσκεται στην κυρίως ταινία, δηλαδή έξω από το clip του κυλιόμενου κουμπιού (γι' αυτό και η δήλωση _parent._parent επειδή το κυλιόμενο κουμπί βρίσκεται δύο επίπεδα μέσα στο movie clip στο οποίο ανήκει.), να έχει ως τετμημένη, την τεταγμένη του κουμπιού. Επίσης ορίζεται, το ίδιο clip, να μεγενθύνεται οριζοντίως όσο μεγαλύτερες τιμές παίρνει η τεταγμένη του κουμπιού και το αντίθετο. Επίσης ορίζεται το ίδιο κλιπ να μεγανθύνεται καθέτως, ανάλογα με το αν αυξάνεται ή μειώνεται η τιμή της τεταγμένης του κουμπιού. Η τεταγμένη _y του κουμπιού παίρνει τιμές από -66 ως +66. Μέσα στον προηγούμενο βρόγχο, ορίζεται ότι όταν η τιμή της τεταγμένης του κουμπιού είναι μεγαλύτερη της μονάδας, να εμφανίζεται ένα συγκεκριμένο κείμενο. Όταν η τιμή της _y είναι μικρότερη της μονάδας διαφορετικό κείμενο, και όταν παίρνει τιμές μεταξύ του -1 και του 1 τότε να εμφανίζεται διαφορετικό κείμενο.

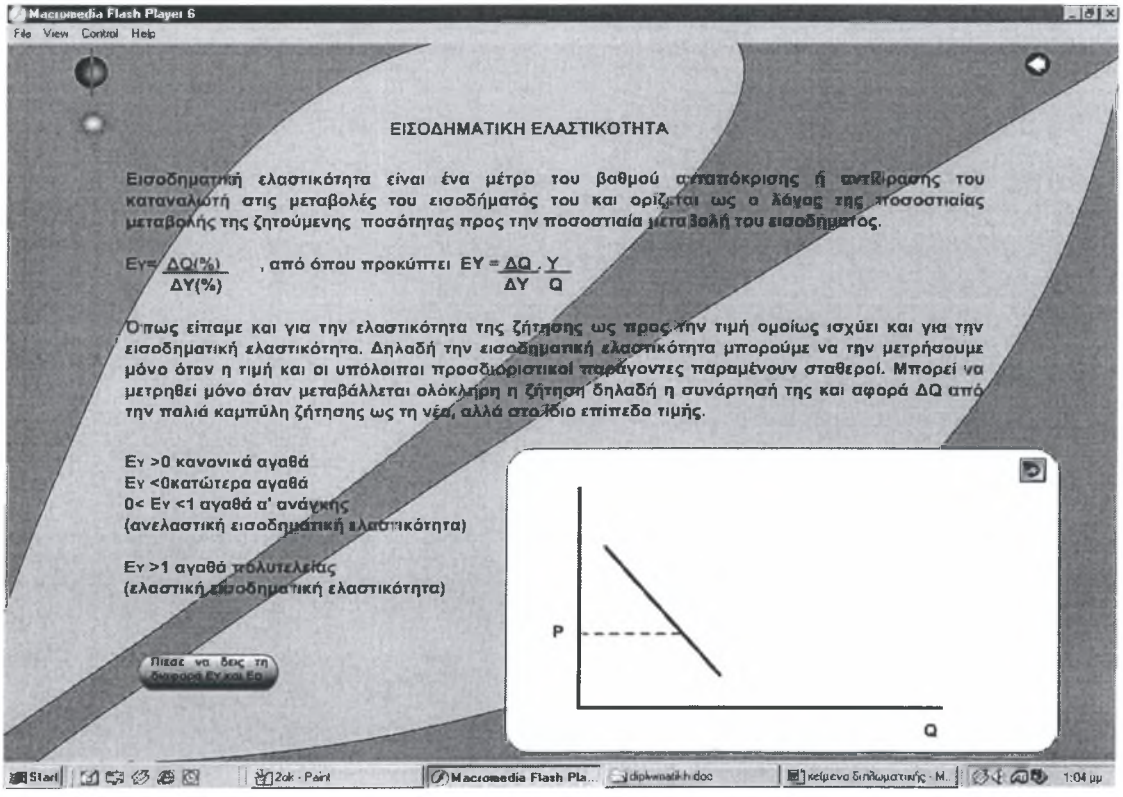
Στο προτελευταίο καρέ του κεφαλαίου παρευρίσκονται κάποια κλιπ που εμπεριέχουν animation και κάποια πεδία κειμένου, όπως επίσης και κουμπιά που καλούν εξωτερικά swf, για την επεξήγηση της θεωρίας. Υπάρχει ένα κλιπ που εμπεριέχει animation και δείχνει τι συμβαίνει με κάποια μεγέθη της ζήτησης, και πως αυτά σχετίζονται μεταξύ τους όπως φαίνεται πρακάτω.



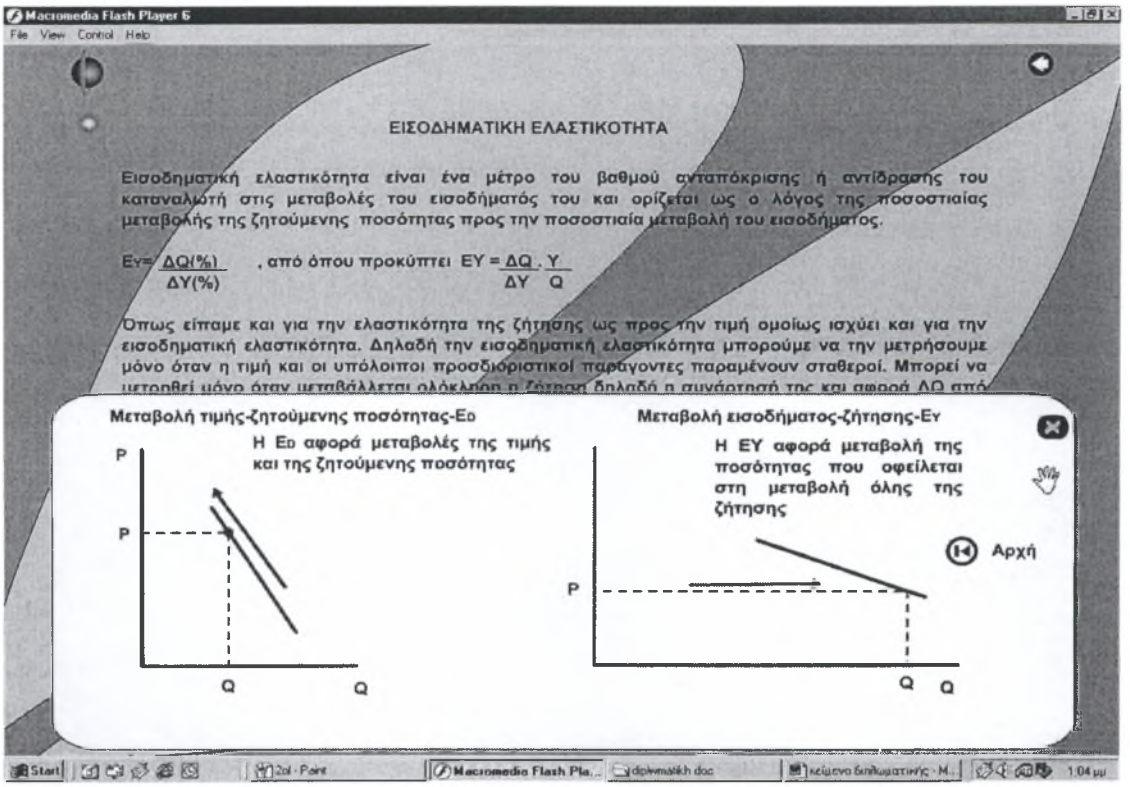
όπως βλέπουμε υπάρχει ένα κουμπί με το όνομα «συνολική δαπάνη (δαπάνη)». Πατώντας αυτό το κουμπί, εμφανίζεται πάλι μια προσομοίωση μιας θεωρίας η οποία εμπεριέχει ό,τι και η προηγούμενη προσομοίωση και συν τοις άλλοις δείχνει πως υπολογίζεται η συνολική δαπάνη και ποιά η σχέση της με την ελαστικότητα. Συν αυτοίς, υπάρχει και μια φωνητική εξήγηση πάνω στο ίδιο θέμα. Αυτά φαίνονται παρακάτω.



Στο τελευταίο καρέ του κεφαλαίου, εκτός από τα πεδία κειμένου, υπάρχουν και κουμπιά που ενεργοποιούν κλιπ. Τα συγκεκριμένα κλιπ έχουν τη δυνατότητα να συρθούν από τον χρήστη, προκειμένου αν θελήσει ο ίδιος να τα αντιπαραβάλει με τη θεωρία.

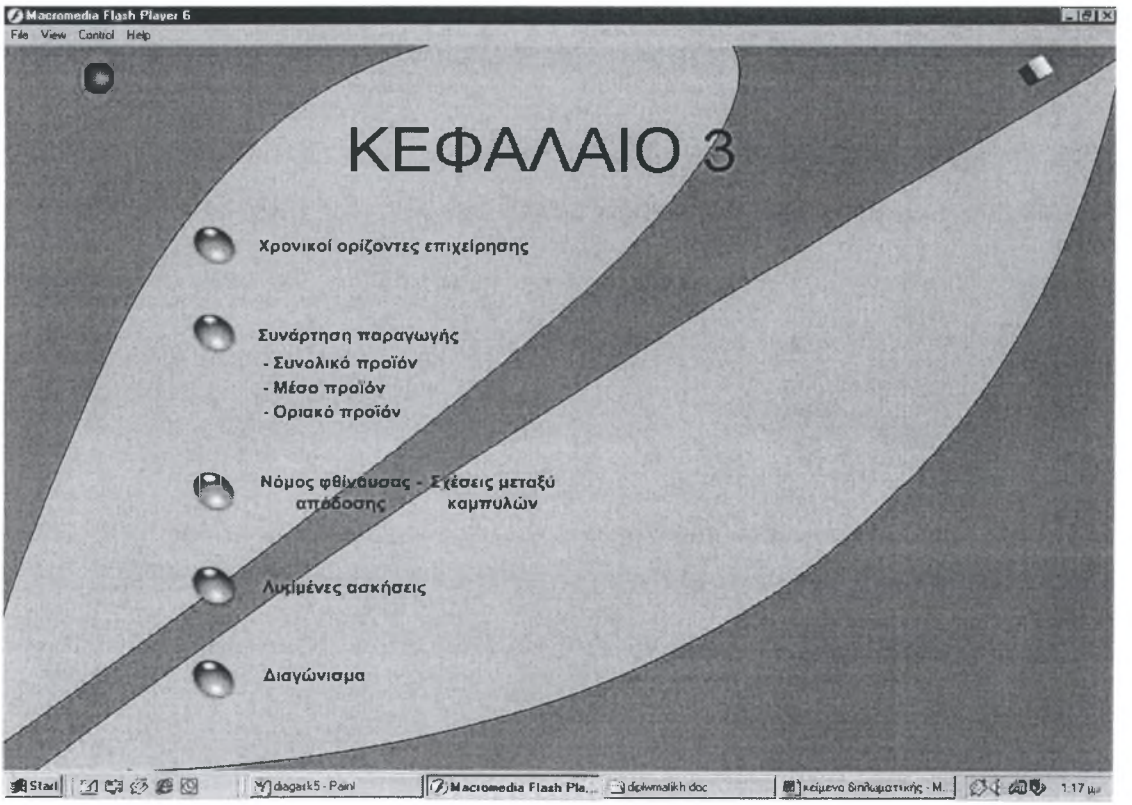


Όπως βλέπουμε κι εδώ υπάρχει ένα κλιπάκι που δείχνει τι ακριβώς αφορά η συγκεκριμένη θεωρία, σύμφωνα με ένα διάγραμμα. Υπάρχει κι ένα κουμπί που μόλις πιεστεί, προβάλλεται ένα εξωτερικό swf, το οποίο εμπεριέχει κίνηση και φωνητική επεξήγηση κάποιου σημείου που έχει ιδιαίτερη σημασία να το ξεκαθαρίσει στο μυαλό του ο μαθητής.

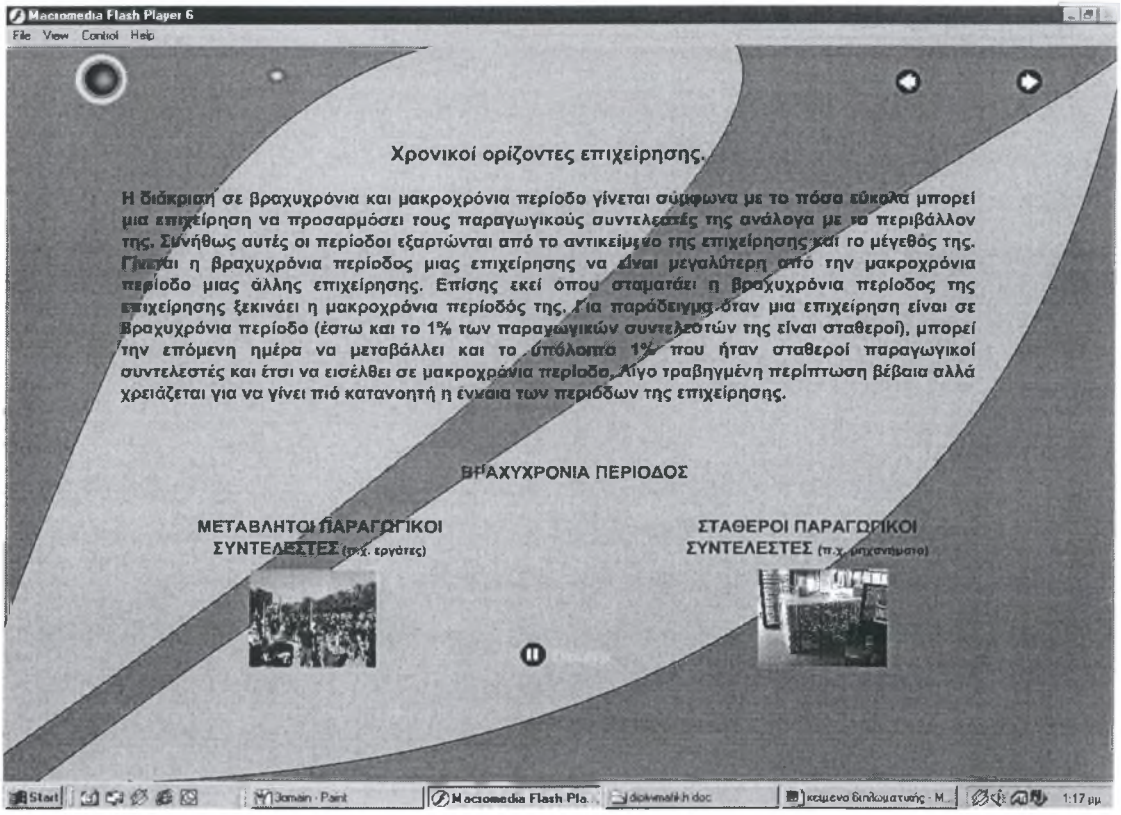


3ο κεφάλαιο:

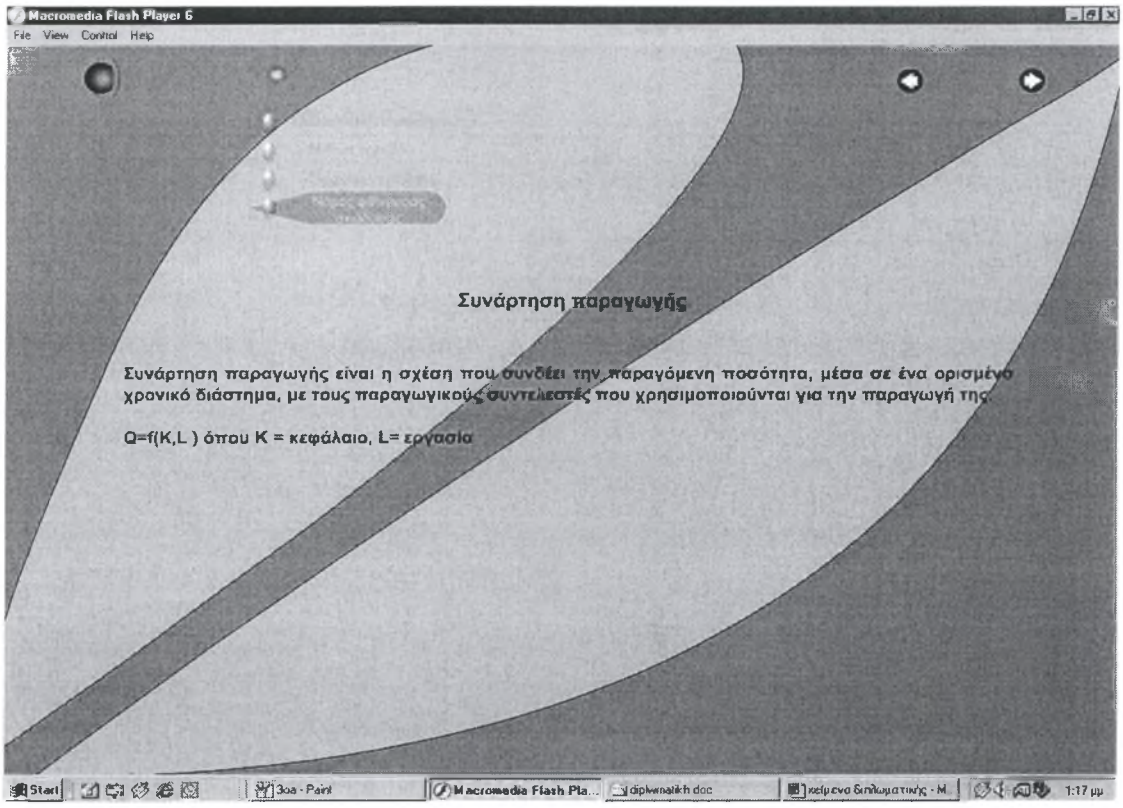
Στην αρχή όπως και στα υπόλοιπα κεφάλαια, υπάρχει πρώτα το περιβάλλον εισαγωγής στο κεφάλαιο με τα αντίστοιχα κουμπιά που παραπέμπουν στην αντίστοιχη θεωρία.



Στο πρώτο καρέ του κεφαλαίου παίζει αυτόματα ένα κλιπ, που περιλαμβάνει κάποιες εικόνες που κινούνται, και μια αφηγηματική επεξήγηση. Ο χρήστης έχει τη δυνατότητα, πατώντας ένα κουμπί, να «παγώσει» την κίνηση των εικόνων.

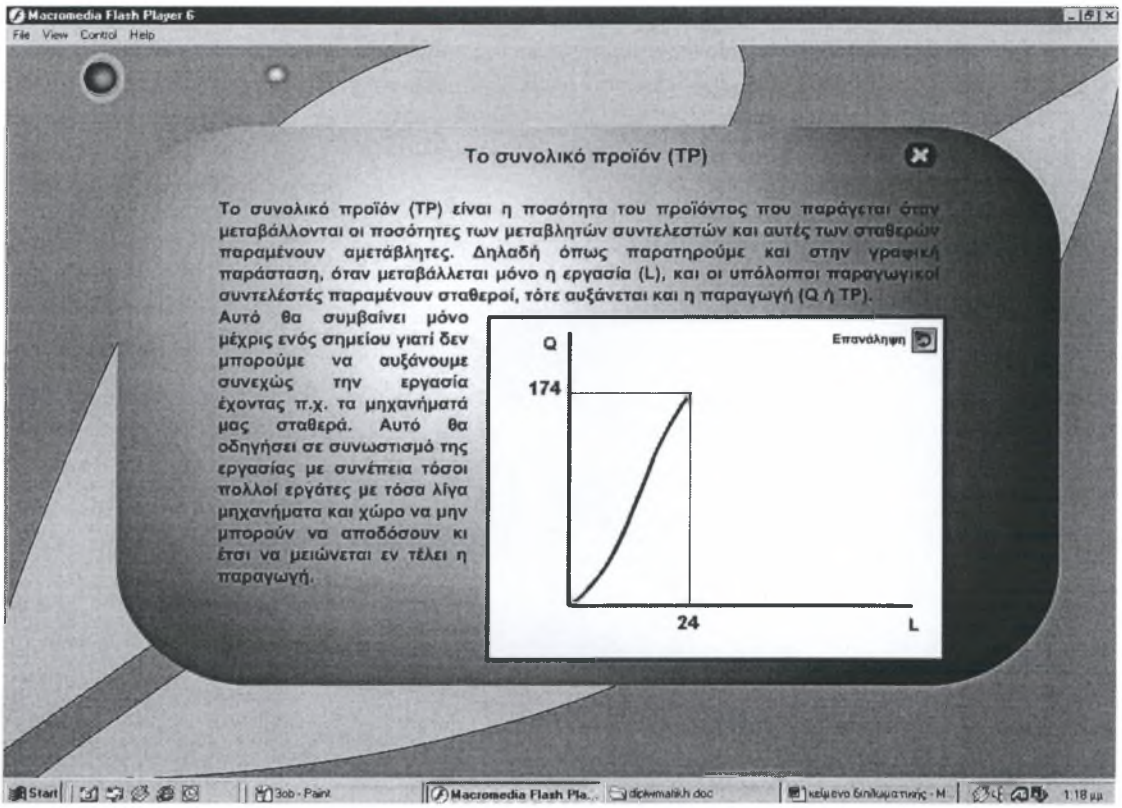


Στο δεύτερο καρέ, υπάρχει ένα κλιπ από ιπτάμενα κουμπιά, που μόλις ο χρήστης φέρει τον δείκτη του ποντικιού πάνω από αυτά, τότε αυτά σταματούν να ίπτανται και λαμβάνουν μια διαφορετική διάταξη. Δίπλα τους δε, εμφανίζονται τα ονόματα των θεματικών ενοτήτων, οι οποίες είναι διαθέσιμες για επιλογή από τον χρήστη όπως φαίνεται παρακάτω.

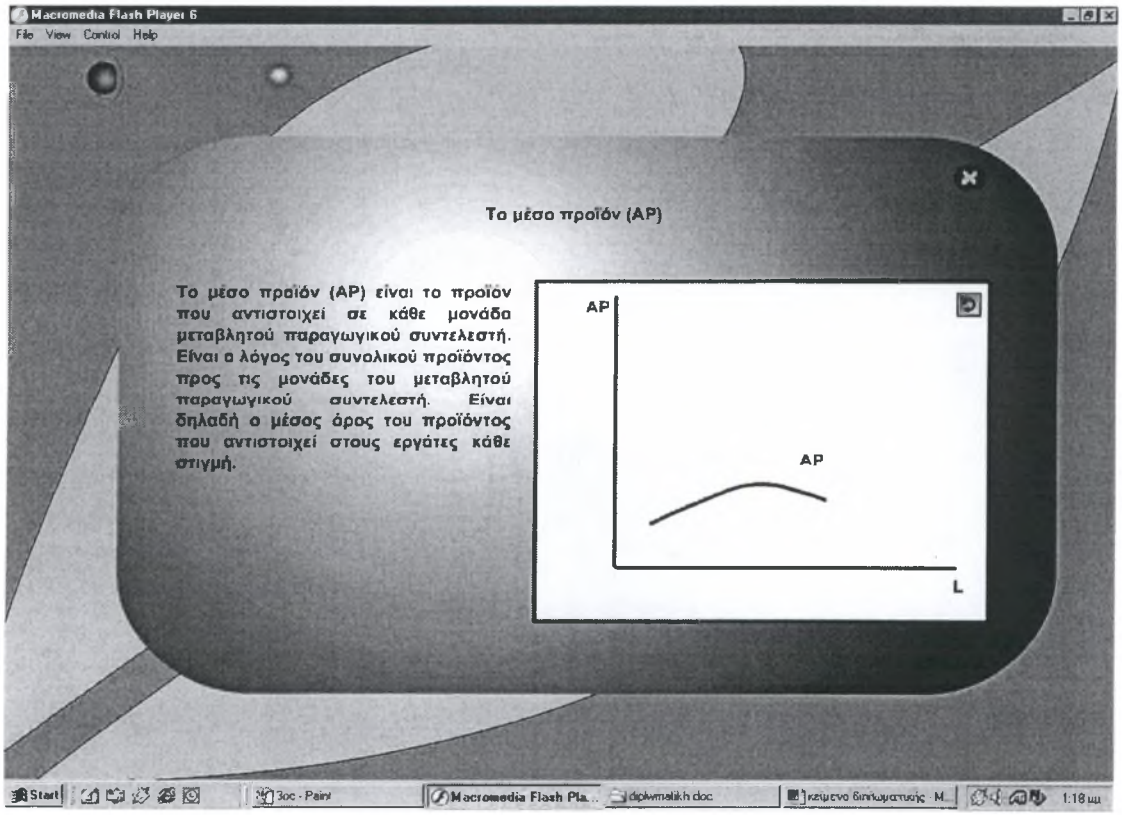


Μόλις πατηθεί κάποιο από αυτά τα κουμπιά, καλείται το αντίστοιχο swf. Αυτά τα swf περιέχουν animation, κείμενο και ήχο. Εξηγούν διάφορες φάσεις κάποιας οικονομικής καμπύλης και σε ένα κλιπ προσφέρεται η δυνατότητα στον χρήστη με την πίεση ενός κουμπιού, να διαλέξει ποιά φάση της καμπύλης θέλει να δει και να ακούσει. Όλα αυτά γίνονται με εντολές gotoAndPlay(); και με εντολές φόρτωσης και αποφόρτωσης ήχου.

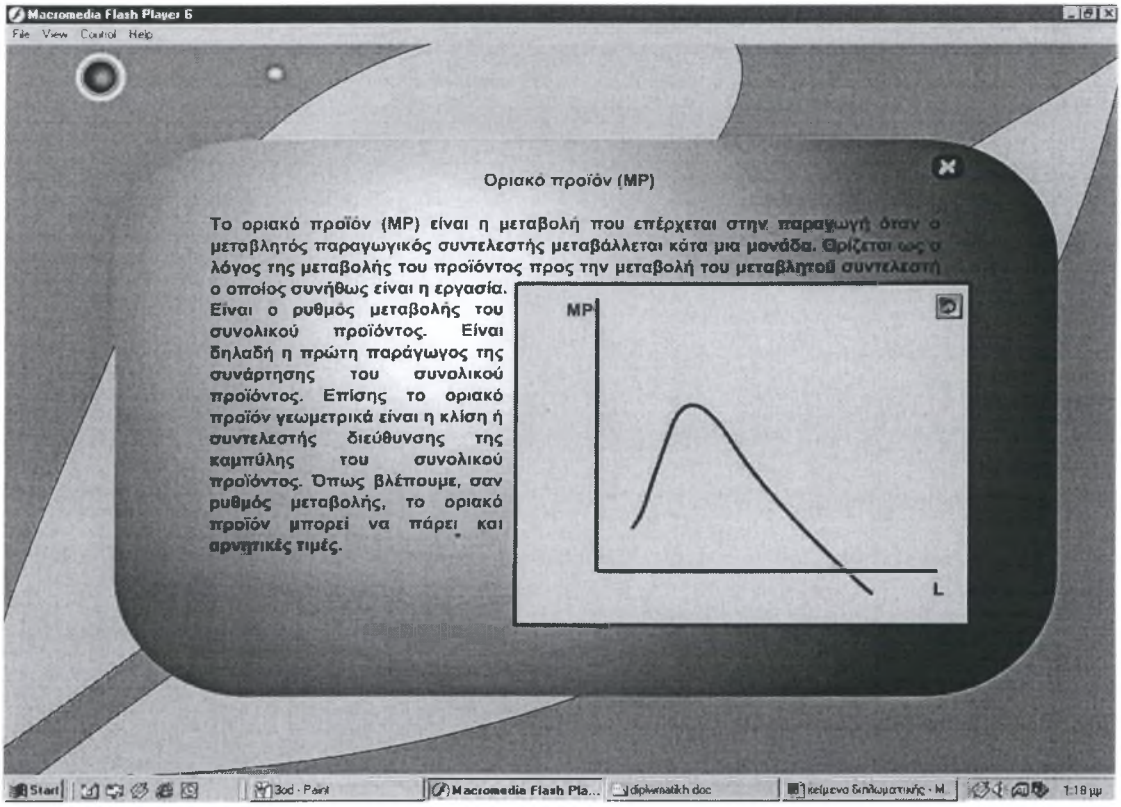
Τα τρία πρώτα κουμπία της προηγούμενης εικόνας ανοίγουν κάποια swf, που περιέχουν κείμενο, και φωνή, και επεξηγούν κάποιους ορισμούς και μέσω κινηματικών παραστάσεων δείχνουν την πορεία της γραφικής παράστασης του αντίστοιχου μεγέθους. Παρακάτω βλέπεουμε ένα υπόδειγμα τι συμβαίνει αν πατηθεί ένα από τα τρία πρώτα κουμπιά. Αν πατηθεί το κουμπί «Συνολικό προϊόν» τότε εμφανίζεται το swf:



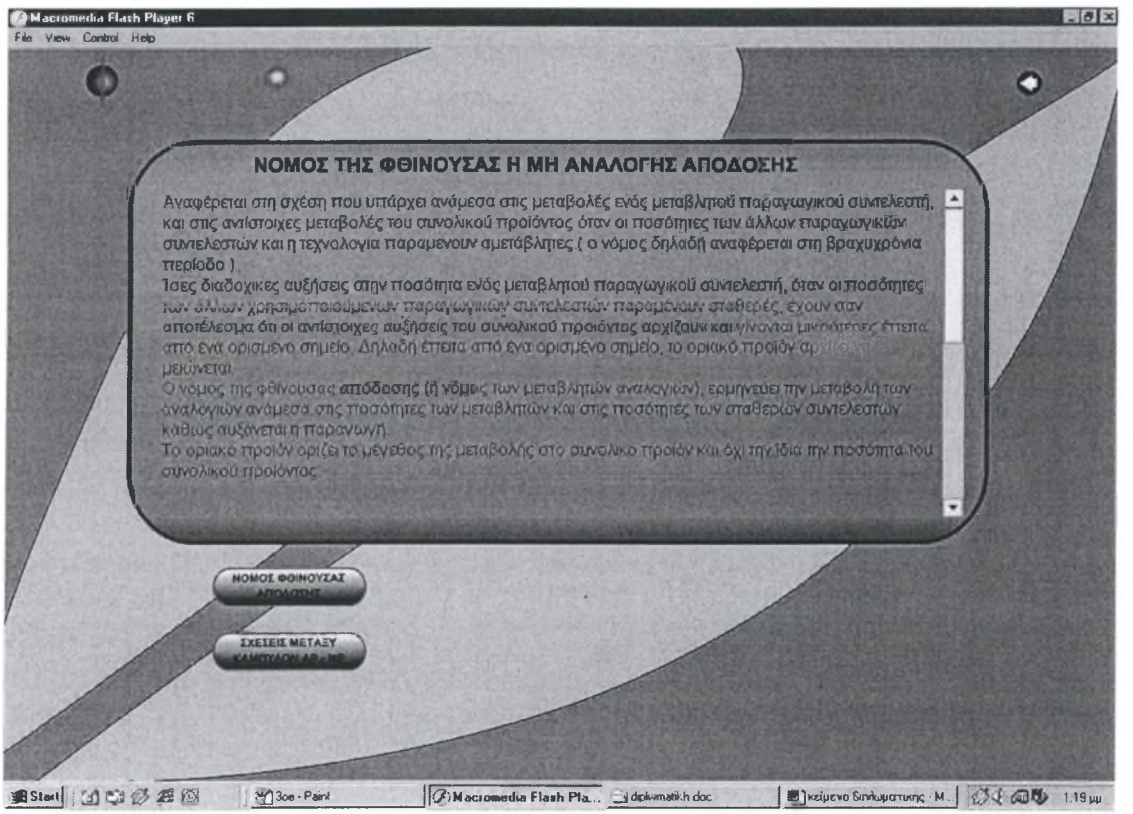
Αν πατηθεί το κουμπί «Μέσο προϊόν» εμφανίζεται το swf:



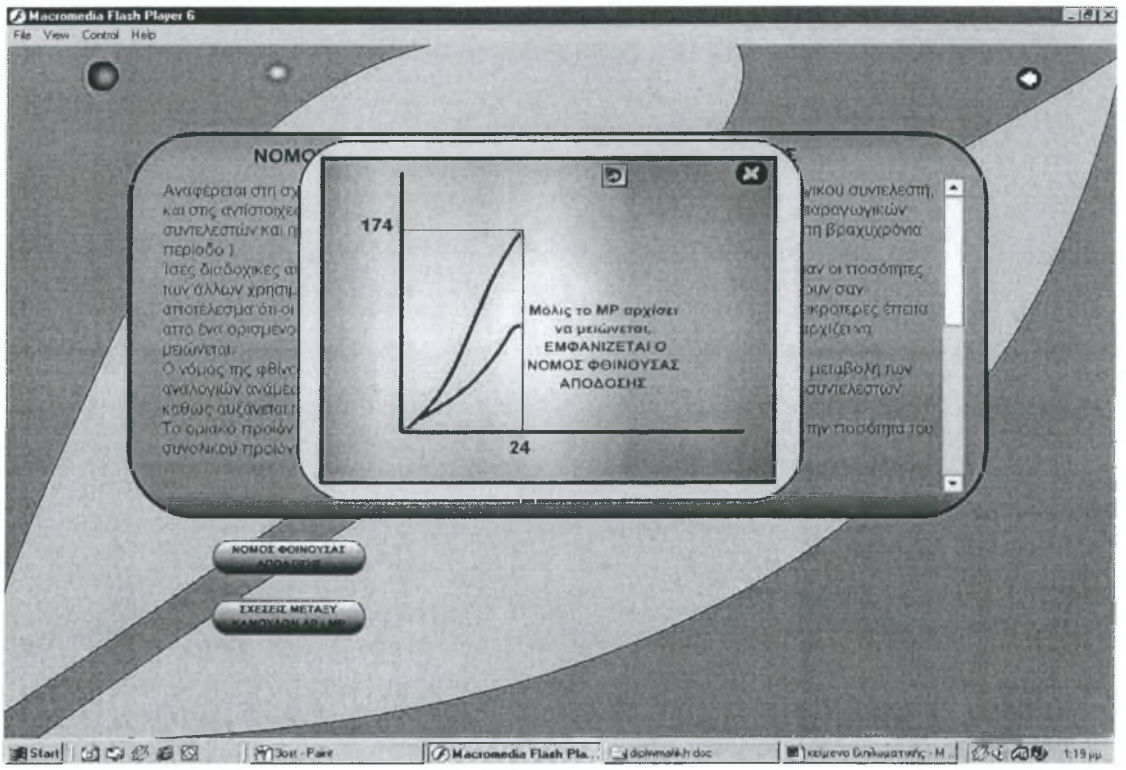
και αν πατηθεί το κουμπί «Οριακό προϊόν»,



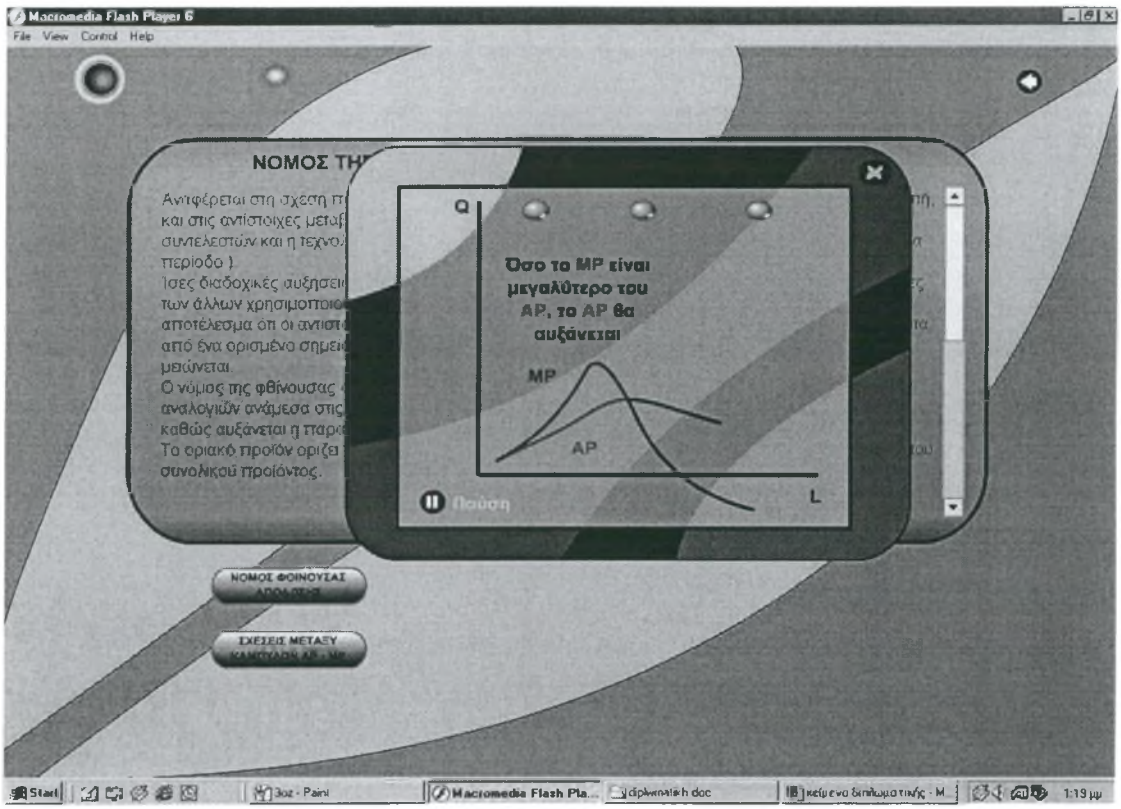
Τέλος αν πατηθεί το τέταρτο κουμπί, μεταβαίνουμε σε ένα άλλο καρέ του τρίτου κεφαλαίου, το οποίο περιέχει ένα κυλιόμενο πεδίο κειμένου και δύο κουμπιά που δείχνουν επιπλέον επεξηγήσεις.



Αν πατηθεί το κουμπί «Νόμος φθίνουσας απόδοσης» τότε εμφανίζεται ένα εξωτερικό swf που περιέχει κίνηση κείμενο, και φωνή, προκειμένου για την καλύτερη κατανόηση της θεωρίας που εμπεριεχόταν στο κυλιόμενο κείμενο της προηγούμενης εικόνας.

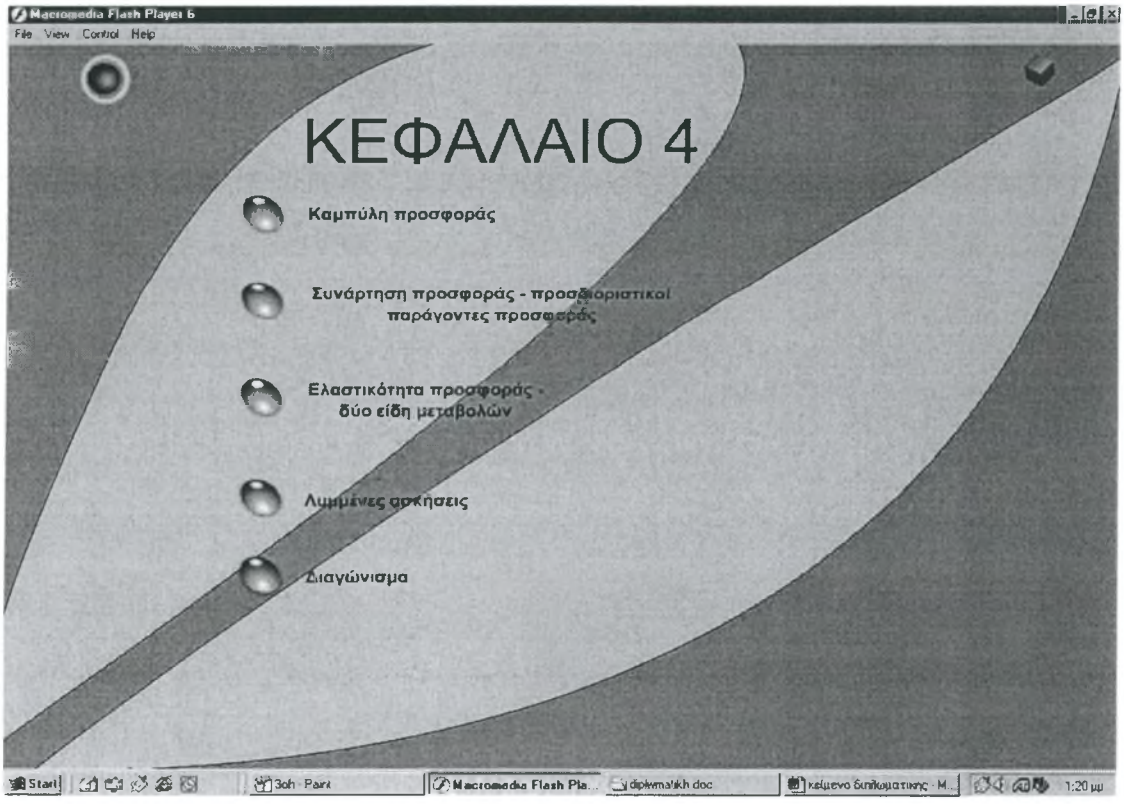


Τέλος αν πατηθεί το κουμπί «Σχέσεις καμπυλών AP-MP» τότε εμφανίζεται το τελευταίο swf του κεφαλαίου το οποίο περιέχει κι αυτό κινηση κείμενο και ήχο. Σ' αυτό υπάρχουν και τρία κουμπία που δίνουν τη δυνατότητα στο χρήστη, πατώντας τα, να πηγαίνει στο αντίστοιχο μέρος της θεωρίας του συγκεκριμένου swf.

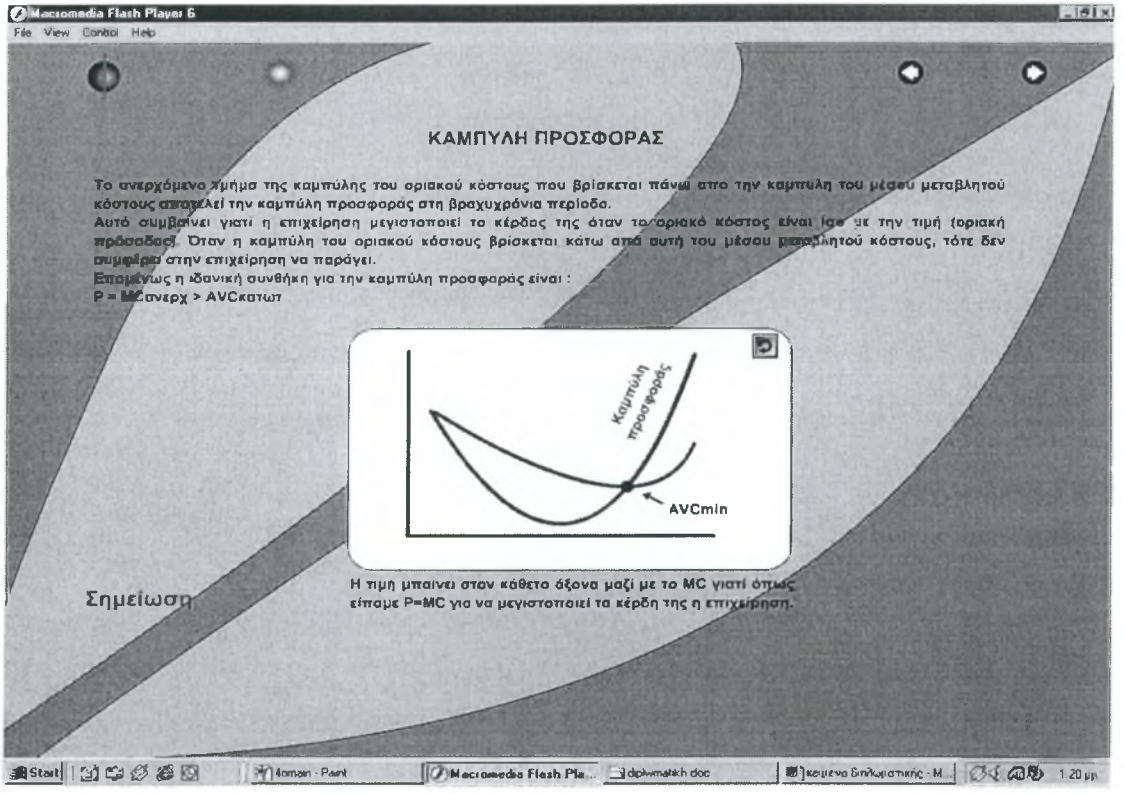


4° κεφάλαιο:

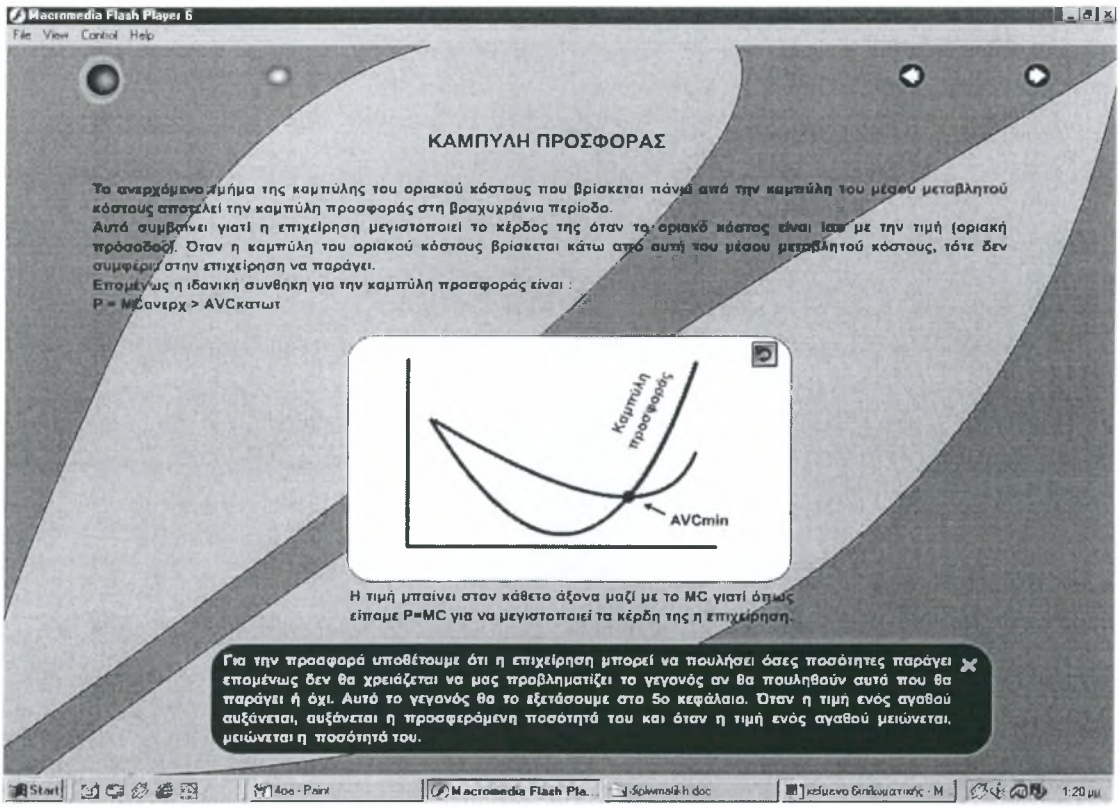
πρωτίστως ο χρήστης έρχεται σε επαφή με το εισαγωγικό περιβάλλον του κεφαλαίου όπου έχει στη διάθεσή του τα κουμπιά για να διαλέξει τί θέλει να δει.



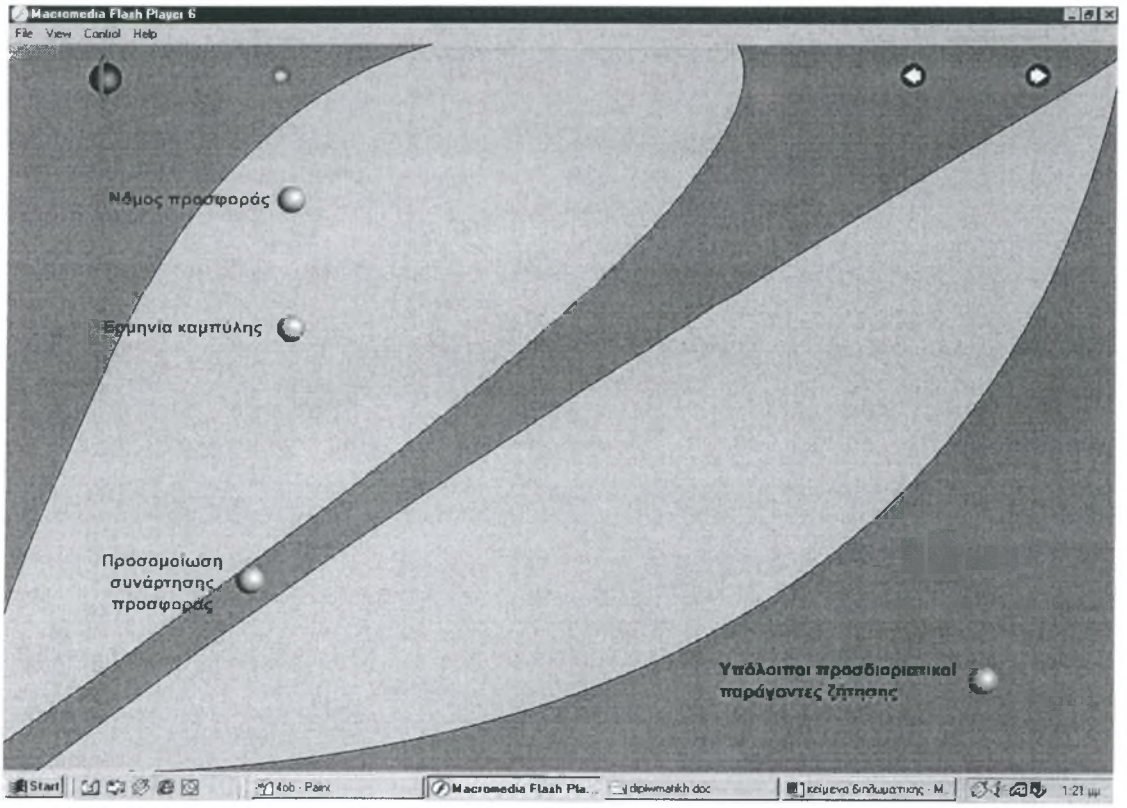
Στο πρώτο καρέ του τετάρτου κεφαλαίου, ο χρήστης έρχεται σε επαφή με τη θεωρία, βλέποντας ένα κλιπ με φωνητική επεξήγηση και κείμενο.



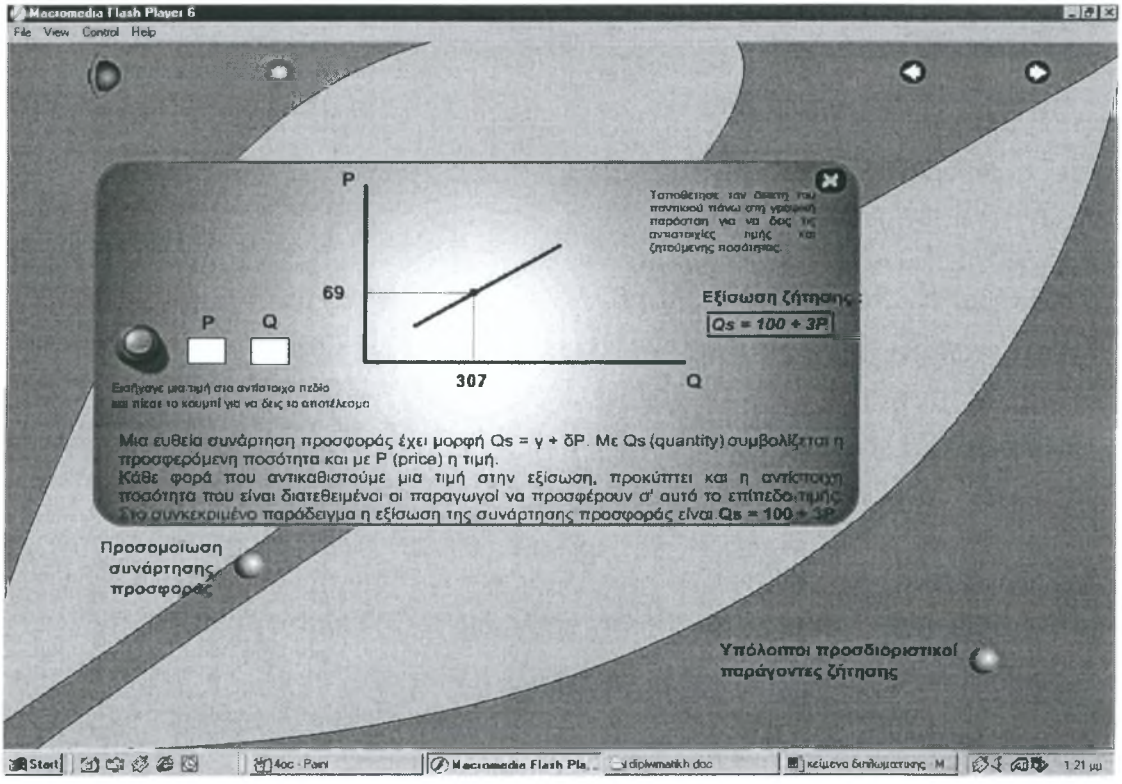
Με το πάτημα ενός κουμπιού, μπορεί να δει και να ξανακούσει τη θεωρία. Επίσης λίγο αργότερα, προβάλλεται ένα κλιπάκι που περιέχει κείμενο, που μετακινείται προκειμένου να τραβήξει την προσοχή του χρήστη, γιατί με το πάτημα αυτού, προκύπτει μια αρκετά ουσιώδης υποσημείωση όπως φαίνεται παρακάτω.



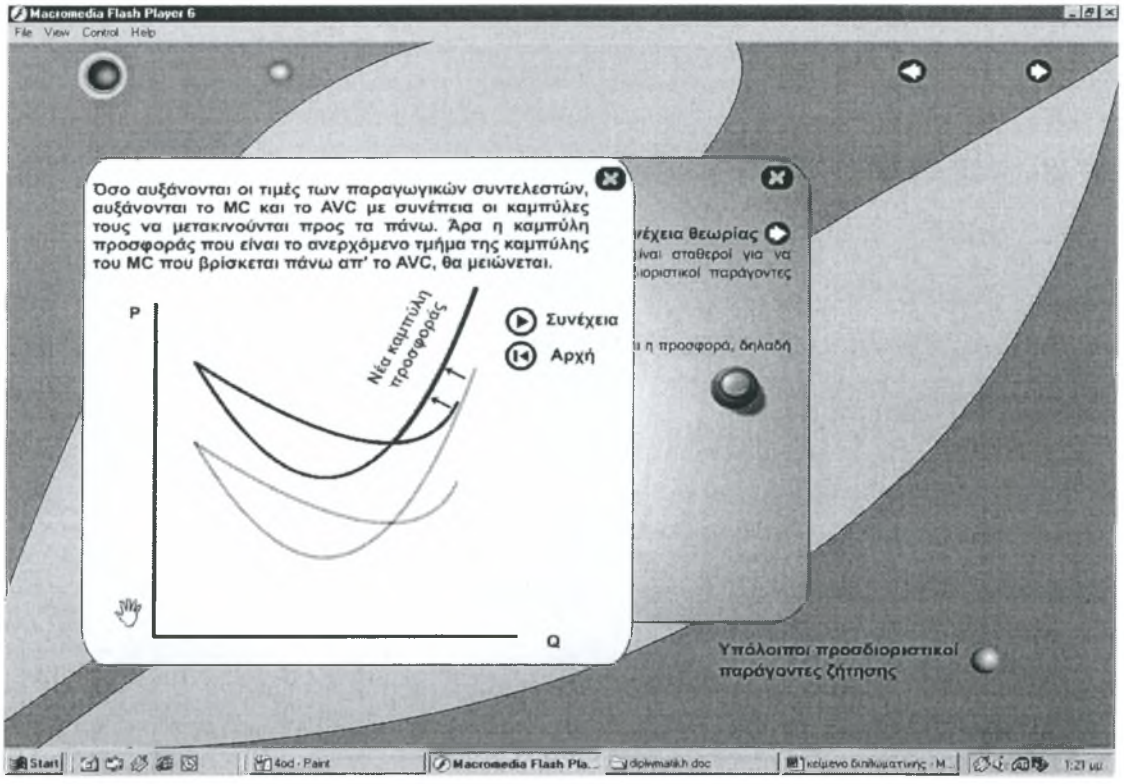
Έπειτα, το δεύτερο και το τρίτο καρέ δομούνται σαν τα αντίστοιχα του δευτέρου κεφαλαίου, μιας και η θεωρία τους, μοιάζει. Κουμπιά που καλούν εξωτερικά swf, που περιέχουν κείμενο ήχο και animation, καθώς και ένα swf που προσομοιώνει τη θεωρία, όπως γινόταν και με τον αντίστοιχο προσομοιωτή του δευτέρου κεφαλαίου. Παρακάτω παρατίθεται εικόνα από το δεύτερο καρέ.



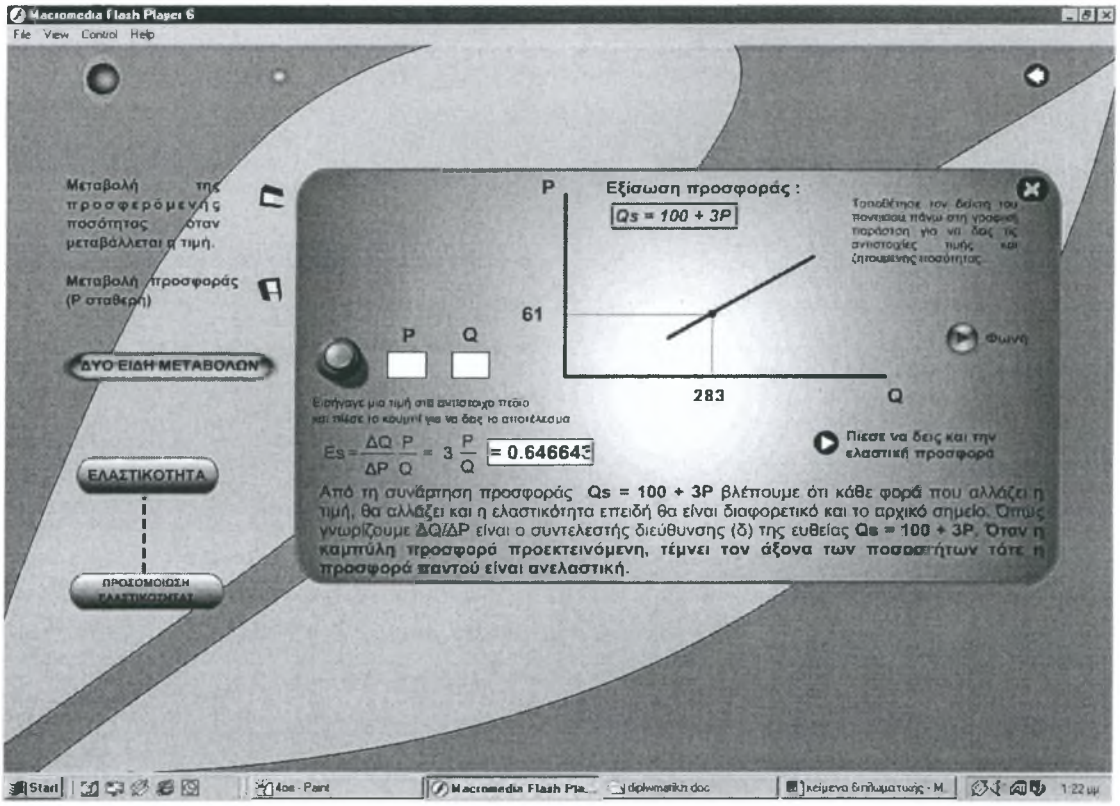
Επειδή το τέταρτο κεφάλαιο μοιάζει με το δεύτερο, λήφθηκε μέριμνα έτσι ώστε να μοιάζει και το περιβάλλον του ώστε ο χρήστης να το συνδυάζει με το δεύτερο κεφάλαιο για να το θυμάται καλύτερα. Αν πιεστεί το κουμπί «Προσομοίωση συνάρτησης προσφοράς» τότε εμφανίζεται το παρακάτω clip το οποίο μοιάζει με την προσομοίωση του δευτέρου κεφαλαίου. Ο χρήστης πάλι τοποθετεί τον δείκτη του ποντικιού πάνω από την γραφική παράσταση και προκύπτει η αντίστοιχη τιμή και ποσότητα σύμφωνα με την υπάρχουσα συνάρτηση. Επίσης κι εδώ υπάρχει το αντίστοιχο πλαίσιο εισαγωγής κειμένου, αν θέλει ο χρήστης να εισάγει κάποια τιμή κι έτσι να εμφανιστεί η αντίστοιχη ποσότητα στο διπλανό πλαίσιο κειμένου καθώς και να εμφανιστούν οι δύο τιμές στην γραφική παράσταση ως συντεταγμένες. Όλα αυτά φαίνονται παρακάτω.



Αντίστοιχα αν πιεστεί το κουμπί «Υπόλοιποι προσδιοριστικοί παράγοντες προσφοράς», ανοίγουν, όμοια με την ζήτηση, κάποια swf, που δείχνουν την αντίστοιχη θεωρία. Αυτά χρησιμοποιούν κείμενο, γραφικά, κίνηση και φωνή. Ένα παράδειγμα βλέπουμε παρακάτω.

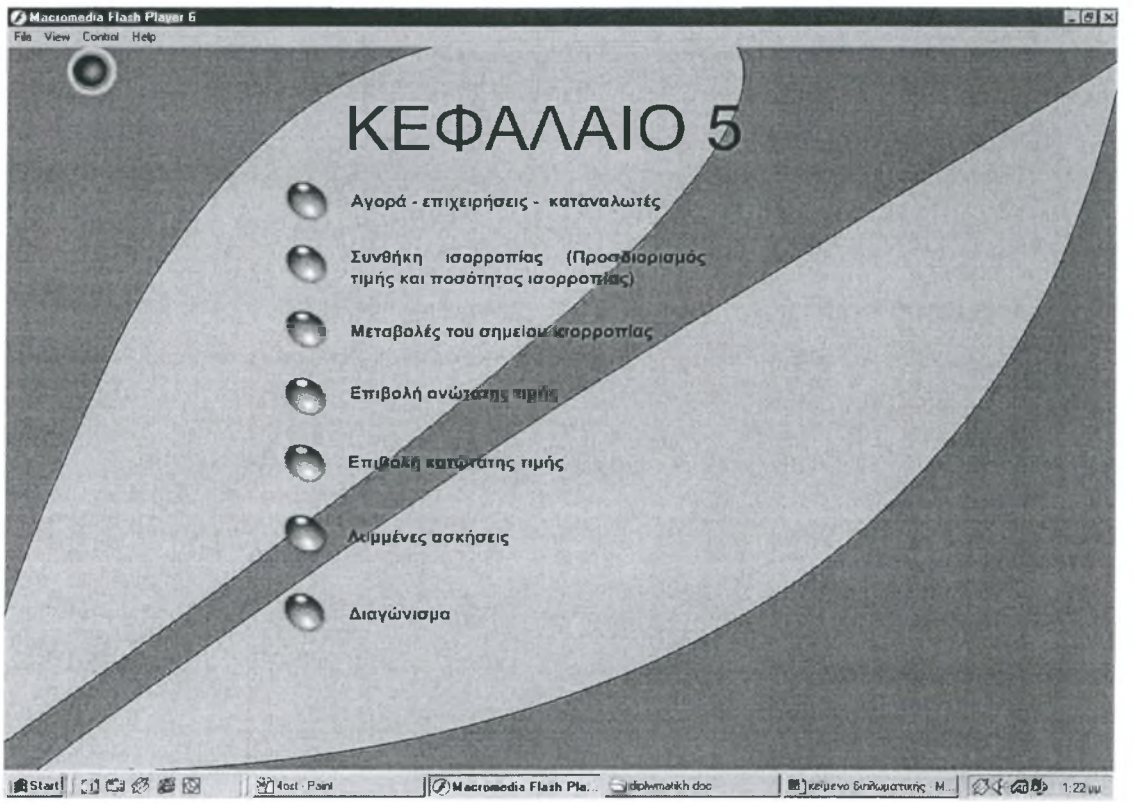


Στο τελευταίο καρέ βλέπουμε πάλι διάφορα κουμπιά όπου με το πάτημα του καθενός προκύπτουν κάποια κλιπ που περιέχουν κείμενο. Με το πάτημα του κουμπιού «Προσομοίωση ελαστικότητας» εμφανίζεται πάλι το swf με την προσομοίωση όπου αυτή τη φορά εμπεριέχονται κάποια επιπλέον στοιχεία όπως κάποια που αφορούν την ελαστικότητα. Ο χρήστης εχει τη δυνατότητα, πιέζοντας κάποιο κουμπί να δεί κι άλλες προσομοιώσεις τέτοιου είδους που αφορούν καμπύλες προσφοράς με διαφορετική ελαστικότητα. Ένα κουμπί υπάρχει, που εξυπηρετεί την περίπτωση που ο χρήστης θέλει να ξανακούσει την φωνητική επεξήγηση.

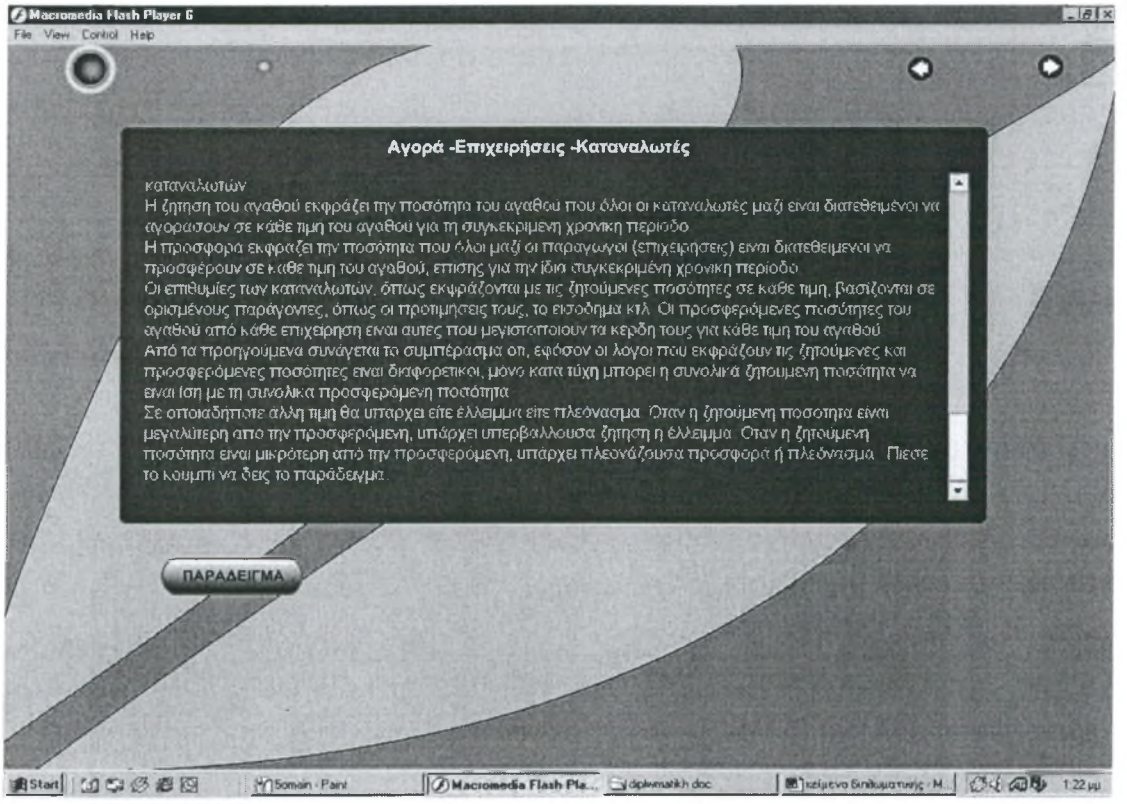


5° κεφάλαιο:

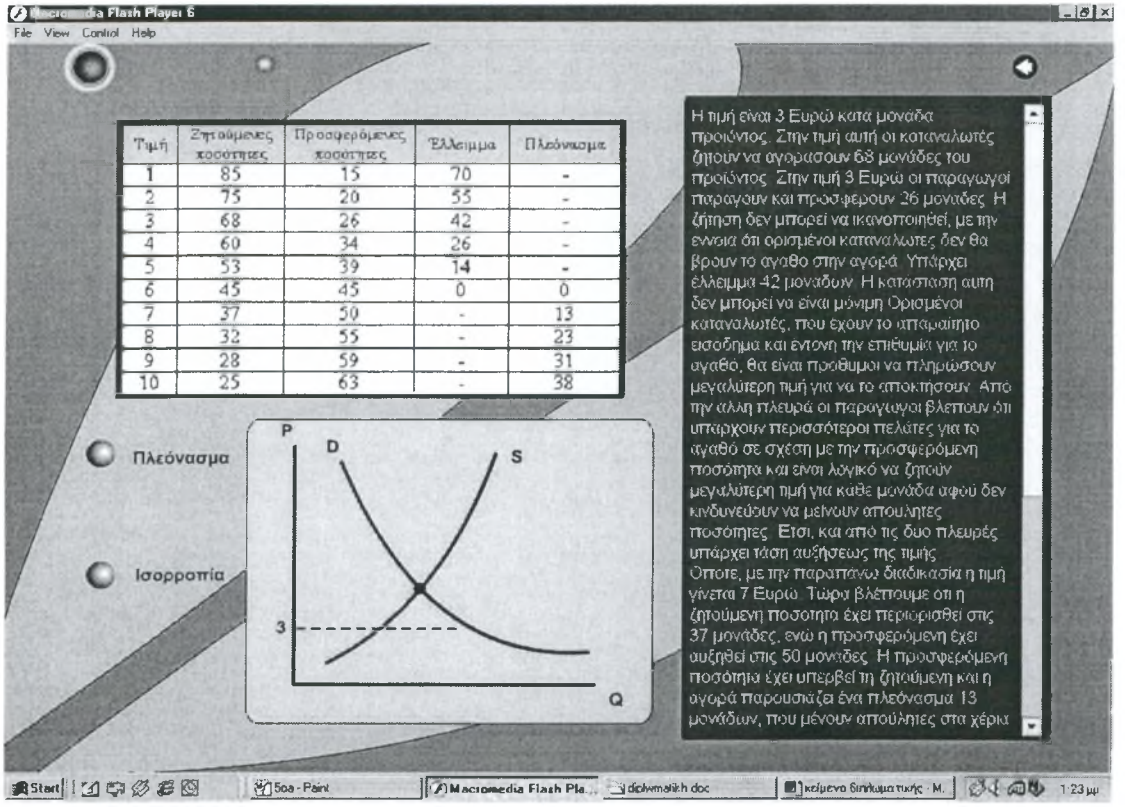
Το πρώτο καρέ έχει να κάνει με το δεύτερο επίπεδο δηλαδή το σημείο που ο χρήστης επιλέγει ποιά θεματική ενότητα θέλει να δει.



Αποτελείται από ένδεκα καρέ. Το πρώτο καρέ, περιλαμβάνει ένα κυλιόμενο πεδίο κειμένου, όπου μόλις η μπάρα κύλισης φτάσει στο χαμηλότερο σημείο της, εμφανίζεται ένα κουμπί που παραπέμπει τον χρήστη σε ένα παράδειγμα.

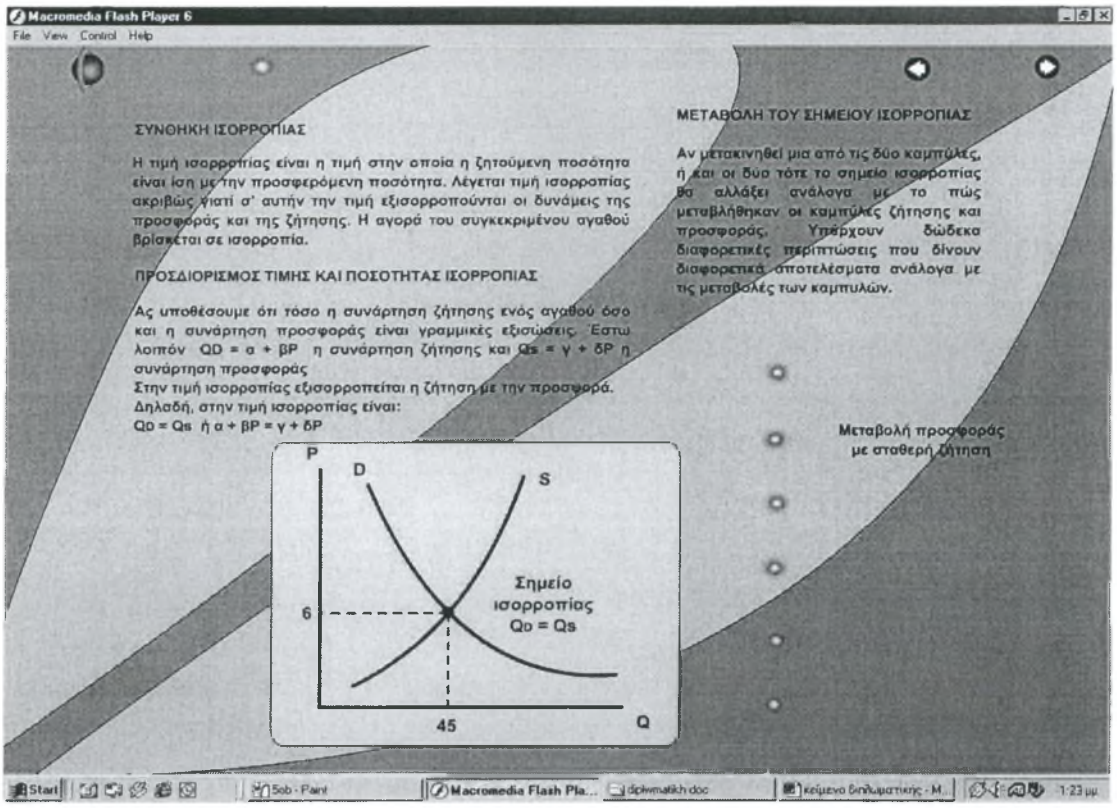


Μόλις πατηθεί το κουμπί, εμφανίζεται το παράδειγμα. Το παράδειγμα, που βρίσκεται στο δεύτερο καρέ, περιέχει κείμενο, animation και ήχο. Με το πάτημα κουμπιών, ο χρήστης βλέπει, διαβάζει και ακούει διάφορα σημεία της θεωρίας, που δεικνύονται μέσω του παραδείγματος όπως φαίνονται παρακάτω.



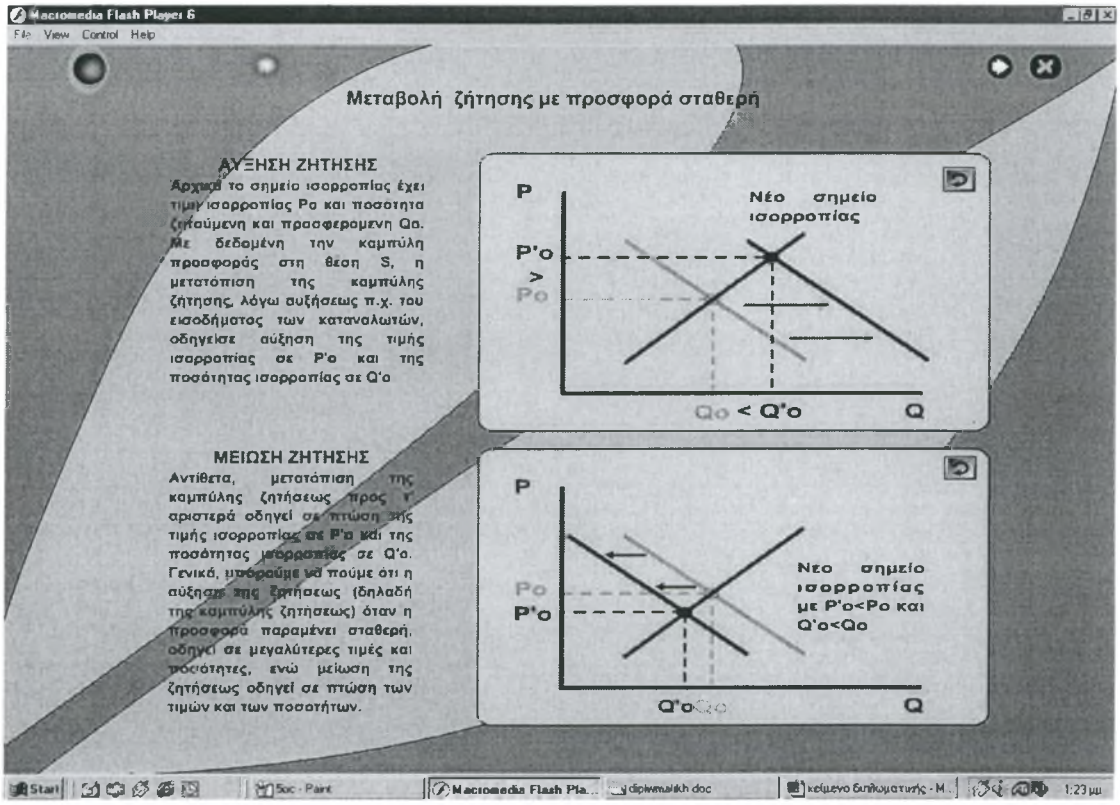
Ο χρήστης βλέπει έναν πίνακα που αφορά το παράδειγμα, ένα κυλιόμενο πλαίσιο κειμένου, και τρία κουμπιά δίπλα από μια γραφική παράσταση. Κάθε φορά που πατιέται κι ένα κουμπί, συμβαίνει και το αντίστοιχο γεγονός στη γραφική παράσταση, και ακούγεται ο ανάλογος ήχος, τα οποία αποτελούν και την εξήγηση του φαινομένου που απάιτησε να δει ο χρήστης.

Το τρίτο καρέ περιέχει κείμενο και έξι κουμπιά, που το καθένα οδηγεί και σε μια διαφορετική περίπτωση που εξετάζει η θεωρία.

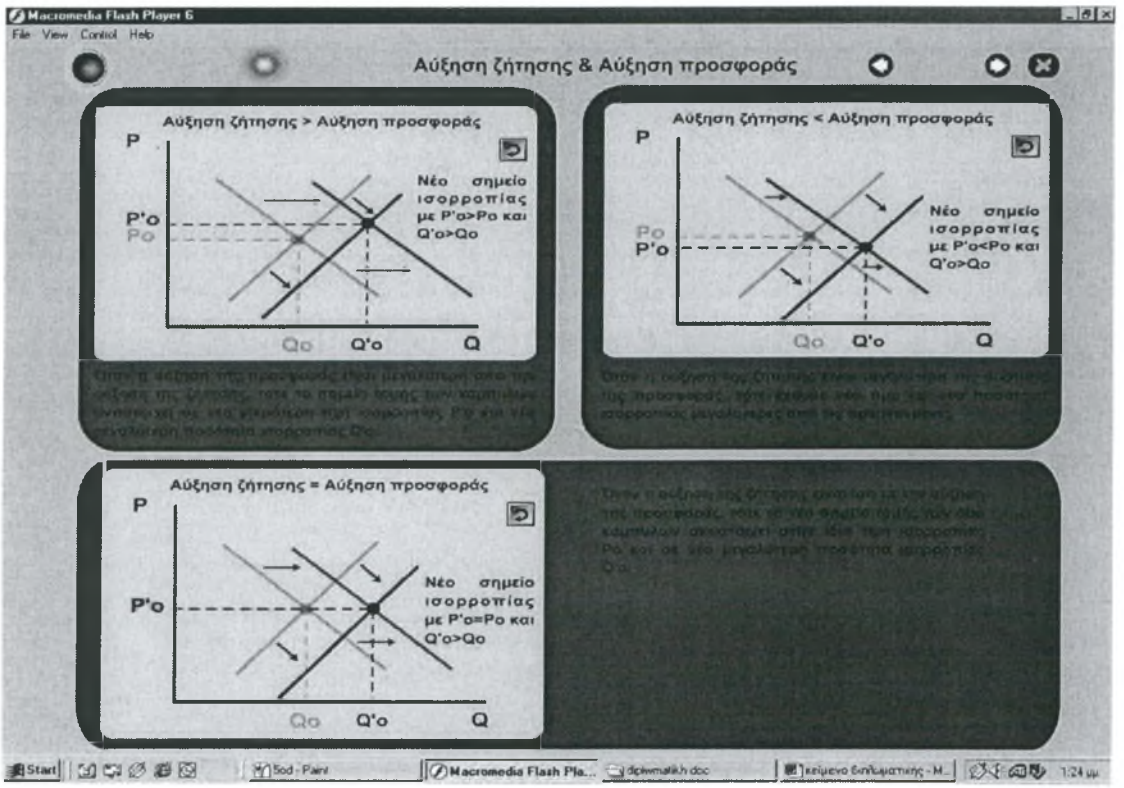


Σ'αυτό το καρέ υπάρχει κείμενο και έξι κουμπία που μόλις ο χρήστης τοποθετήσει τον δείκτη του ποντικιού πανω σε ένα από αυτά, εμφανίζεται δίπλα ο τίτλος της θεματικής ενότητας που αντιστοιχεί το κάθε κουμπί. Με το πάτημα του εκάστοτε κουμπιού, ο χρήστης μεταβαίνει στο καρέ με την αντίστοιχη επεξήγηση.

Αυτές οι έξι περιπτώσεις που εξετάζει η θεωρία, βρίσκονται σε διαφορετικό καρέ η καθεμία. Σε κάθε καρέ φορτώνονται εξωτερικά swf, όπου γίνεται χρήση κλιπ που περιέχουν κείμενο, ήχο και animation, κι ο χρήστης έχει τη δυνατότητα, αν θέλει, πιέζοντας ένα κουμπί, να ξαναδεί και να ξανακούσει αυτά που τον ενδιαφέρουν.

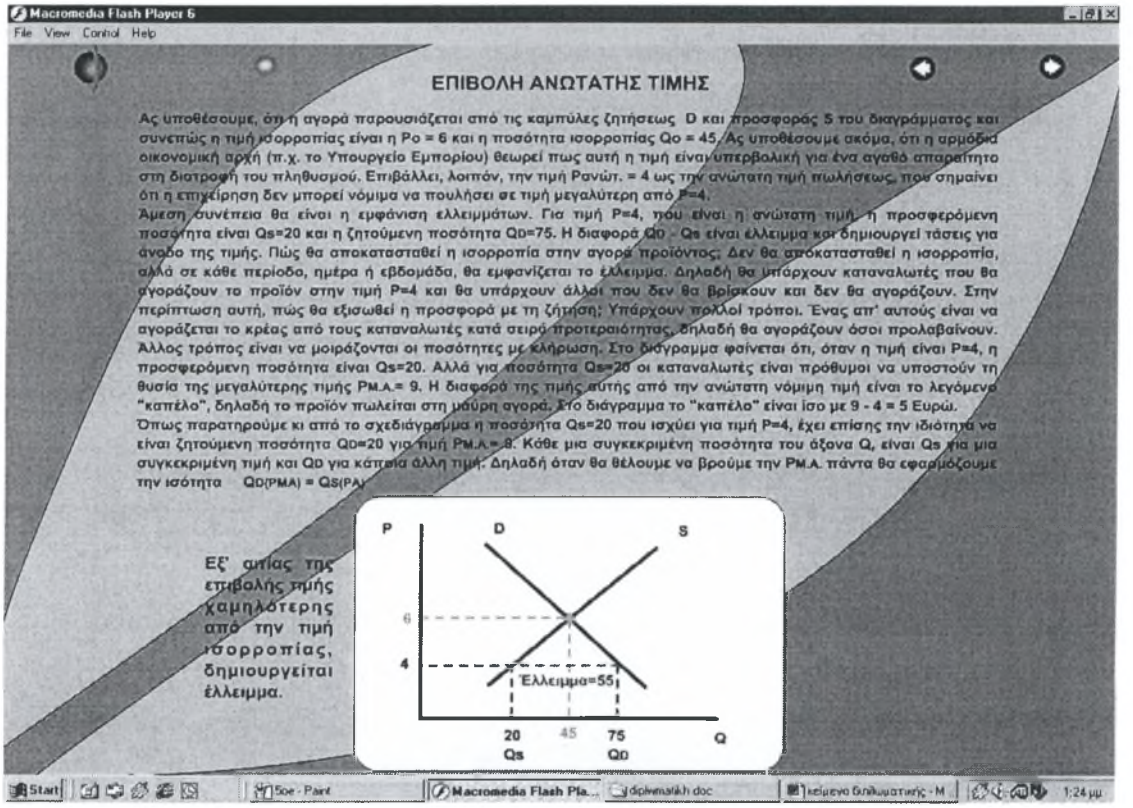


Αυτά τα έξι καρέ, επειδή εξετάζουν έξι διαφορετικές περιπτώσεις της ίδιας θεωρίας, είναι δομημένα με όμοιο τρόπο, έτσι ώστε να καταλαβαίνει ο μαθητής ότι πρόκειται για μέρη της ίδιας θεωρίας, κι έτσι να μην δημιουργείται σύγχιση.

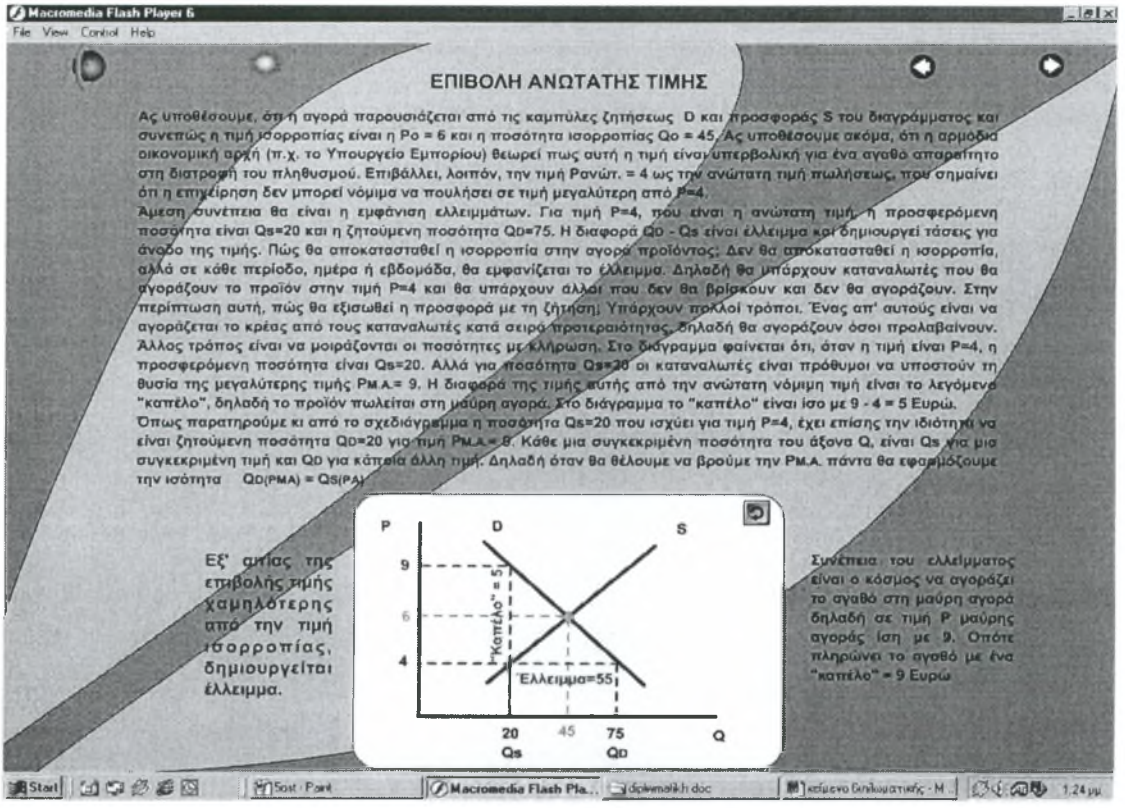


Στις γραφικές παραστάσεις που παρατίθενται, υπάρχει κίνηση και επίσης έχει φροντιστεί η αρχική τοποθεσία των καμπυλών να φαίνεται σκιασμένη έτσι ώστε ο χρήστης να καταλαβαίνει πως μετακινήθηκε η καμπύλη και να συγκρίνει την παλιά θέση της με την νέα.

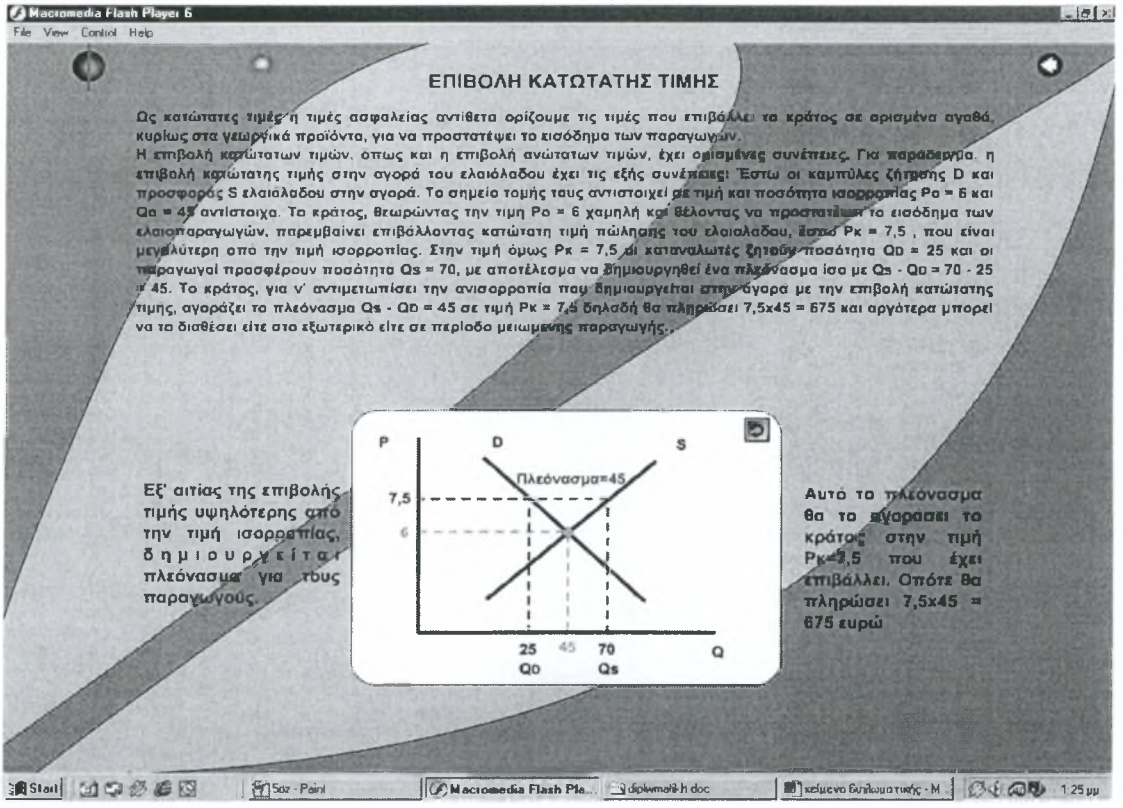
Στο προτελευταίο καρέ υπάρχει κείμενο, και ένα κλιπ που αναμένει τη διάδραση του χρήστη για να παίξει. Μόλις ο χρήστης πατήσει το κουμπί, παίζει το κλιπ το οποίο εκτός απ' το animation, εμφανίζει κείμενο, και ξεκινά και την ανάλογη αφηγηματική επεξήγηση. Η διάρκεια του κλιπ είναι χωρισμένη σε τρία μέρη. Για να συνεχίσει ο χρήστης από το ένα μέρος στο επόμενο, έχει απλά να πιέσει το κουμπί. Με το πρώτο πάτημα του κουμπιού αρχίζει να κινείται η γραφική παράσταση και να εξηγείται η θεωρία.



Έπειτα με το δεύτερο πάτημα συναχίζεται η επεξήγηση και το animation, δηλαδή προχωρά η επεξήγηση της θεωρίας.

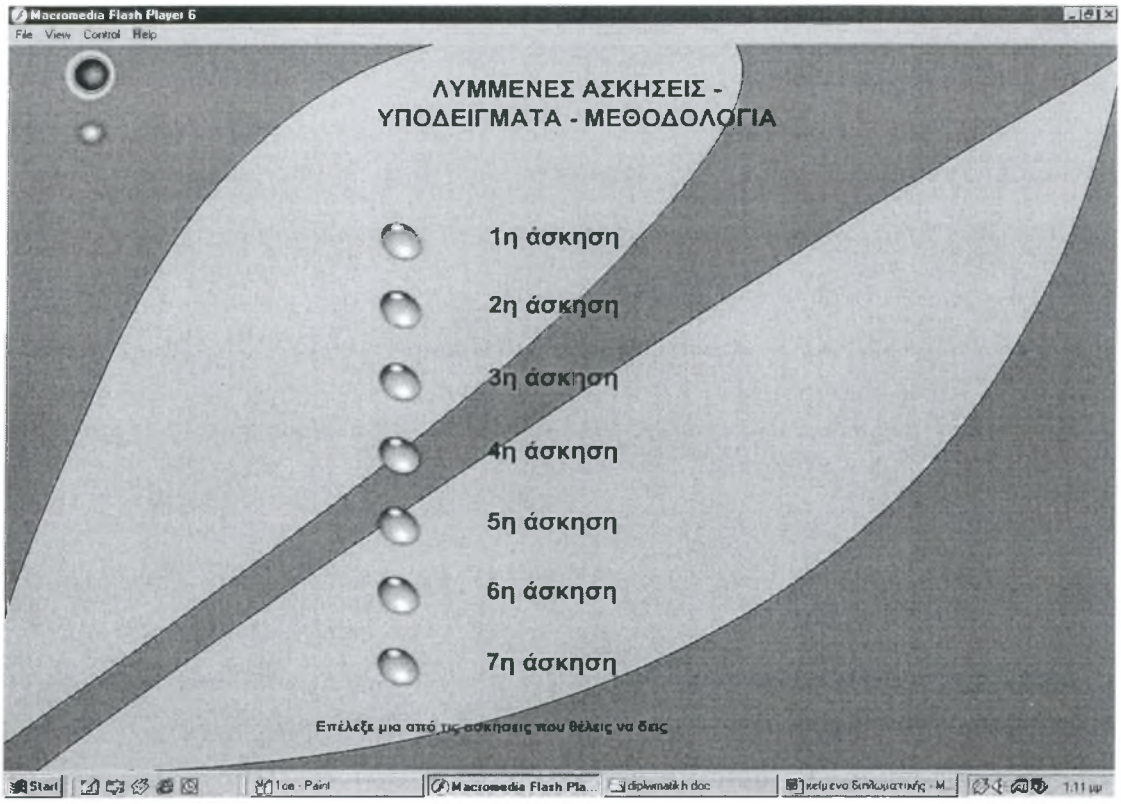


Στο τελευταίο καρέ παρατηρούμε ότι είναι όμοιο με το προηγούμενο λόγω του ότι προκειται για όμοια θεωρία που λειτουργεί με αντίθετο τρόπο. Εδώ η θεωρία για να παιχτεί απότη γραφική παράσταση και από τη φωνή, αναμένονται δύο ανταποκρίσεις του χρήστη. Μία ο χρήστης πατώντας το κουμπί θα ακούσει το πρώτο μισό της συγκεκριμένης θεωρίας, και ξαναπατώντας, θα ακούσει και θα δει την υπόλοιπη.

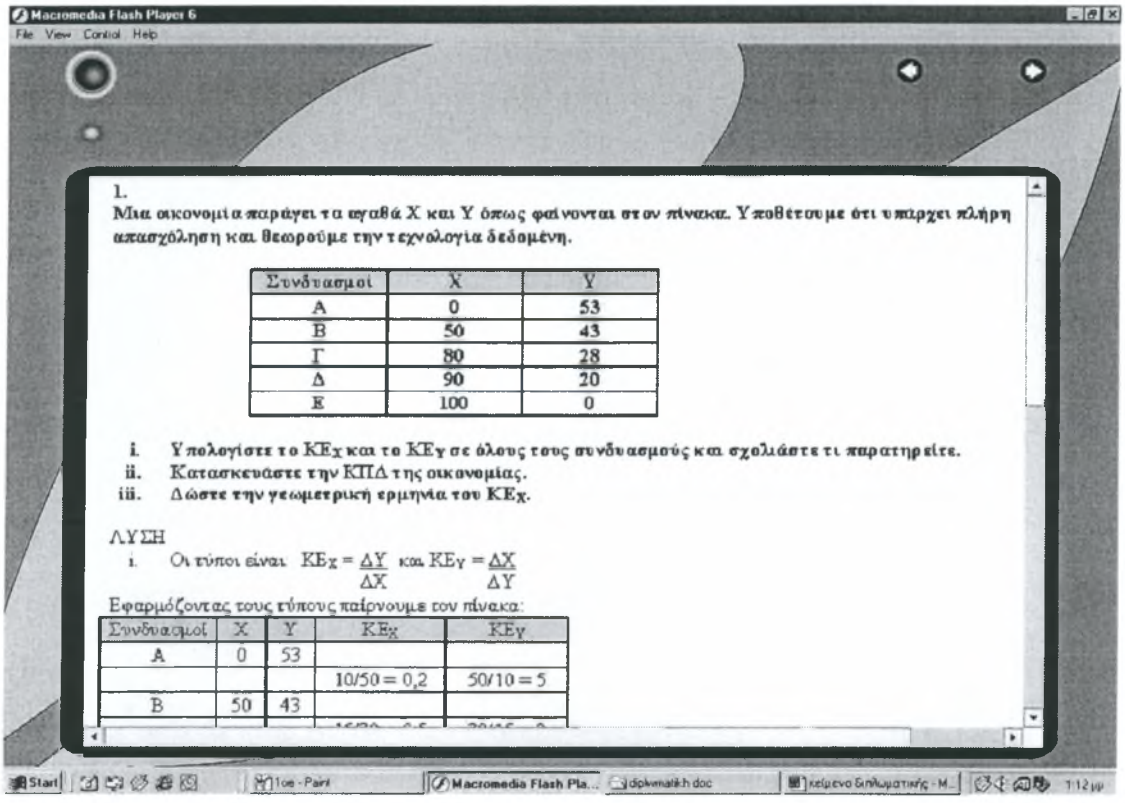


ΛΥΜΜΕΝΕΣ ΑΣΚΗΣΕΙΣ

Στο τέλος κάθε κεφαλαίου υπάρχουν οι λυμμένες ασκήσεις. Από το δεύτερο επίπεδο ο χρήστης μπορεί μόλις νιώσει έτοιμος, να μελετήσει τις λυμμένες ασκήσεις. Μόλις επιλεχθεί το κουμπί του δευτέρου επιπέδου «Λυμμένες ασκήσεις» ο χρήστης παραπέμπεται σ' ένα περιβάλλον όπως φαίνεται στην επόμενη εικόνα, όπου παρατίθενται επτά κουμπιά όπου το καθένα αντιστοιχεί και σε μια λυμμένη άσκηση.



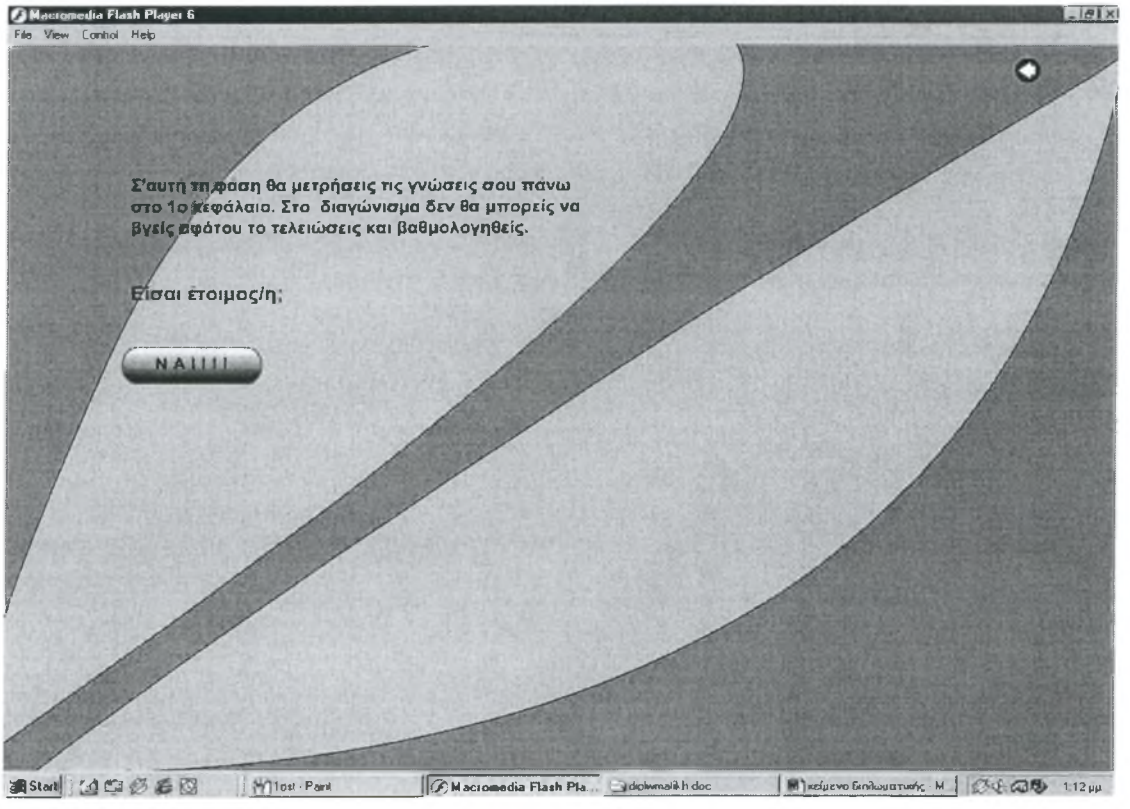
Σ΄ αυτό το επίπεδο ο χρήστης διαλέγει ποιά ασκηση θέλει να δει. Μόλις διαλέξει, εισέρχεται σ' ένα περιβάλλον όπου του παρουσιάζεται η αντίστοιχη άσκηση σε μορφή jpeg, μέσα σ' ένα πλαίσιο που διαθέτει μπάρες κύλισης όπως φαίνεται στην επόμενη εικόνα.



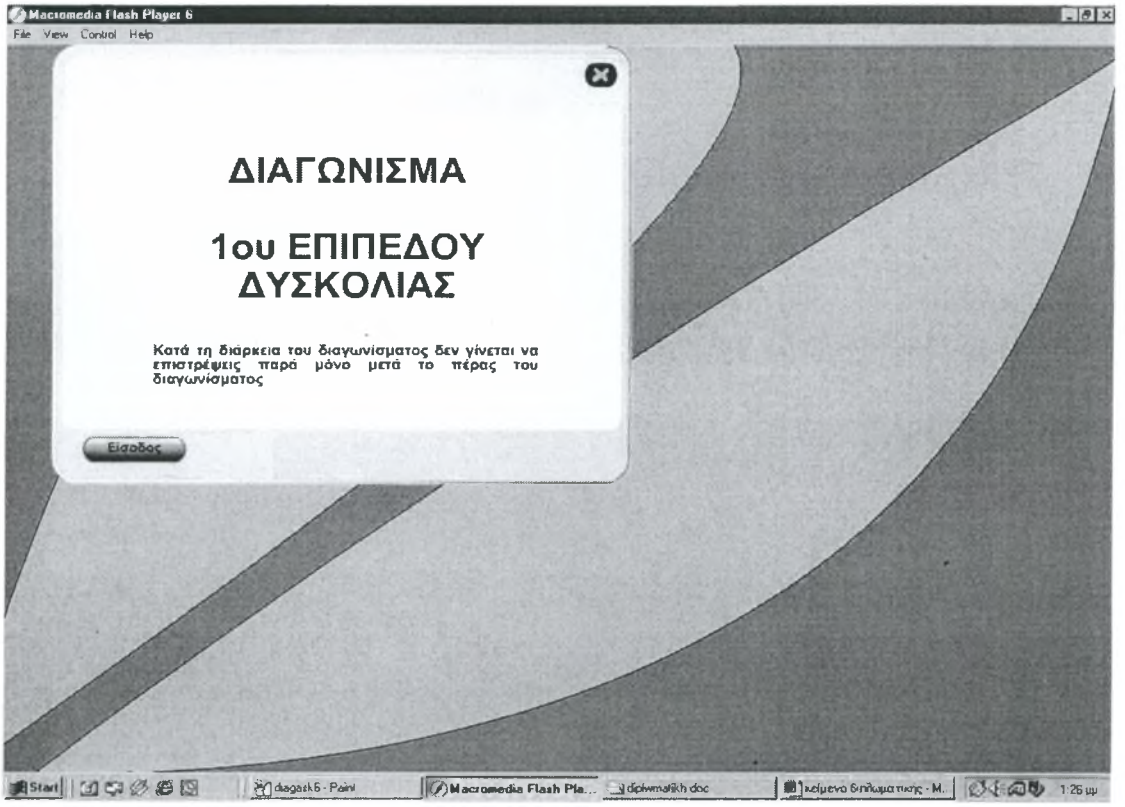
Όπως παρατηρούμε, υπάρχει κουμπί προκειμένου αν ο χρήστης επιθυμεί να δει τις ασκήσεις με τη σειρά μια μια και να μην χρειάζεται να βγαίνει συνεχώς για να διαλέξει την επόμενη.

ΔΙΑΓΩΝΙΣΜΑΤΑ

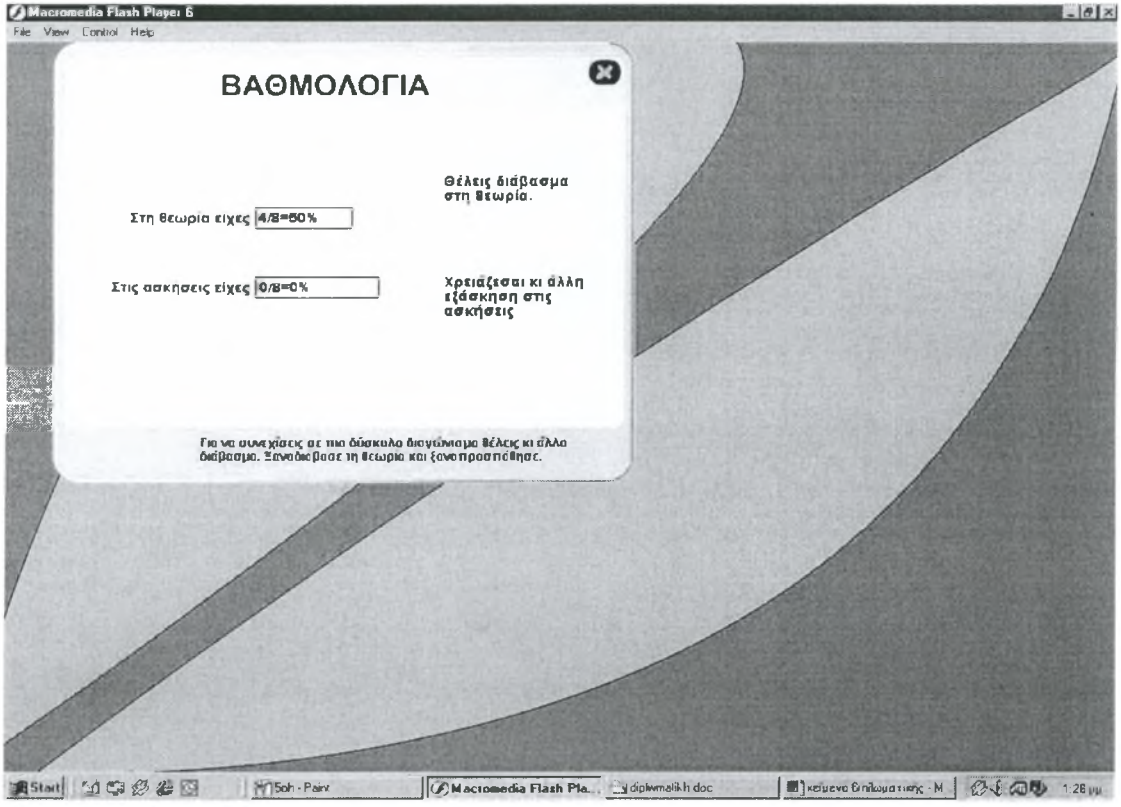
Στην εφαρμογή υπάρχουν συνολικά δεκαπέντε διαγωνίσματα, ένα στο τέλος του κάθε κεφαλαίου και άλλα δέκα στον τομέα των διαγωνισμάτων. Τη στιγμή που ο χρήστης θα μπει στο διαγώνισμα, δεν μπορεί να φύγει παρά μόνο αν κλείσει την εφαρμογή. Σκοπός αυτού, είναι να μην μπορεί ο χρήστης να επιστρέφει στη θεωρία και να ξαναεπιστρέφει πίσω στα διαγωνίσματα προετοιμασμένος μόνο για τα συγκεκριμένα ερωτήματα. Γι' αυτόν τον λόγο όπως φαίνεται και στην επόμενη εικόνα, πριν ο χρήστης μπει στο διαγώνισμα του γίνεται η υπόδειξη ότι δεν θα μπορεί να επιστρέψει.



Στον ξεχωριστό τομέα των διαγωνισμάτων, τα δέκα διαγωνίσματα χωρίζονται σε τρία επίπεδα. Στο πρώτο επίπεδο είναι τα πιο εύκολα διαγωνίσματα, πέντε στον αριθμό. Μόλις ο χρήστης θελήσει να δοκιμαστεί, εισάγεται σε ένα περιβάλλον όπου ερωτάται αν είναι έτοιμος να προχωρήσει στο πρώτο διαγώνισμα. Αν πιεστεί το κουμπί με τη λέξη «Ναι», τότε επιλέγεται τυχαία ένα από τα πέντε διαγωνίσματα του πρώτου επιπέδου. Παρακάτω στην εικόνα βλέπουμε τον τίτλο που φέρουν τα διαγωνίσματα πριν ξεκινήσουν.



Τα διαγωνίσματα του δευτέρου επιπέδου μπορεί να τα δει ο χρήστης, μόνο και μόνο αν, μπορέσει να φέρει ικανοποιητική βαθμολογία στο υπάρχον επίπεδο. Αν φέρει την ικανοποιητική βαθμολογία, τότε μπορεί ο χρήστης να συνεχίσει σε διαγώνισμα του δευτέρου επιπέδου. Αντιθέτως αν δεν μπορέσει τότε εισπράτει το ανάλογο μήνυμα και πρέπει να ξαναπροσπαθήσει. Παράδειγμα παρατίθεται παρακάτω.



Στο δεύτερο επίπεδο υπάρχουν τέσσερα διαγωνίσματα, τα οποία πάλι ανοίγουν στην τύχη. Αν ο χρήστης μπορέσει να φέρει σε πέρας και το πιό δύσκολο διαγώνισμα, τότε ανοίγει το τελικό διαγώνισμα του τρίτου επιπέδου, το πιό δύσκολο όλων. Τι γίνεται όμως αν ο χρήστης δεν μπορέσει να περάσει το διαγώνισμα του πρώτου επιπέδου; Όλοι σκεφτόμαστε ότι θα επιστρέψει στη θεωρία, θα ξαναδιαβάσει, θεωρία και λυμμένες ασκήσεις, και θα ξαναεπανέλθει να δοκιμαστεί. Ασφαλώς και εφόσον επιλέγονται τυχαία είναι πιθανό να μην ξαναπέσει ο χρήστης στο ίδιο διαγώνισμα με πρίν. Παρ' όλα αυτά όμως παραμένουν 20% πιθανότητες ο χρήστης να ξαναβγεί στο ίδιο διαγώνισμα με πριν που απέτυχε. Γι' αυτό το λόγο έπρεπε να εξαλειφθούν κι αυτές οι πιθανότητες. Ο χρήστης αν αποτύχει από το διαγώνισμα, και μετά επιστρέψει, δεν γίνεται να ξαναβρεθεί στο ίδιο διαγώνισμα. Έτσι ώστε να μην θυμάται τις ερωτήσεις και διαβάσει μόνο για τις συγκεκριμένες. Όλα αυτά όμως χρειαζόταν αρκετές γραμμές κώδικα στο κουμπί που στέλνει τον χρήστη στο αντίστοιχο διαγώνισμα.

```
int(met);
int(gr);
gr=0;
int(deytero);
deytero=0;
int(trito);
trito=0;
int(tetarto);
tetarto=0;
int(pempto);
pempto=0;
int(pr);
pr=0;
```

```
met=random(5);
```

Ο κώδικας αυτός έχει οριστεί στην αρχή της εφαρμογής, σε σημείο που ο χρήστης δεν μπορεί να επιστρέψει, έτσι ώστε να ξαναμηδενιστούν οι μεταβλητές. Κάθε μια απ' αυτές τις μεταβλητές αντιστοιχούν σε κάθε ένα από τα διαγωνίσματα και αποτελούν διακόπτες, έτσι ώστε αν το αντίστοιχο διαγώνισμα ανοίξει, η αντίστοιχη μεταβλητή να λάβει την τιμή 1. Έτσι αν ξαναπάει να ανοίξει το ίδιο διαγώνισμα, και αν ο διακόπτης – μεταβλητή έχει την τιμή 1 , τότε δεν θα ανοίγει το ίδιο, αλλά ένα άλλο. Στο τέλος, η μεταβλητή met λαμβάνει μια τιμή από το 0 ως το 4.

```
on (Press) {
        readybutt._visible=false;
        prologue._visible=false;
        pisw._visible=false;
int(tr);
tr=0;
int(te);
te=0;
int(pe);
pe=0;
int(de);
de=0;
int(pro);
pro=0;
```

κατ' αρχήν ορίζεται ότι αν πατηθεί το κουμπί να εξαφανιστεί το κουμπί και ότι άλλο υπάρχει στη σελίδα, έτσι ώστε όταν ανοίξει το διαγώνισμα να μην υπάρχει τίποτα άλλο από πίσω. Αυτά ξαναεμφανίζουνται μόλις ο χρήστης τελειώσει το διαγώνισμά του. Επίσης ορίζεται μια πλειάδα μεταβλητών όπου καθεμιά αντιστοιχεί σε ένα διαγώνισμα, για να γίνουν εφικτοί οι έλεγχοι και να επιτευχθεί αυτό που εξηγήθηκε νωρίτερα.

```
if (_root.met==0)
{pro=1;}
if (_root.met==1)
{de=1;}
if (_root.met==2)
{tr=1;}
if (_root.met==3)
{te=1;}
if (_root.met==4)
{pe=1;}
```

έπειτα περιλαμβάνονται πέντε βρόγχοι που αφορούν, την από πριν ορισμένη μεταβλητή met, και ανάλογα με την τιμή που θα πάρει η met , θα δωθεί και η τιμή 1 στην αντίστοιχη μεταβλητή, που όπως είπαμε, καθεμία αντιστοιχεί σε κάθε ένα διαγώνισμα. Έτσι, για πρώτη φορά θα ανοίξει ένα διαγώνισμα. Για τις επόμενες φορές όμως, χρειάζονται επιπλέον έλεγχοι.

```
if (pro==1)
```

```
{
_root.met=random(4);

        if (_root.met==0)
        {
                if(_root.deytero==0)
                {
                loadMovie("2odiag.swf",invdiag);
                _root.deytero=1;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                        loadMovie("4odiag.swf",invdiag);
                _root.tetarto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.pempto=0;
                }
        }
        if (_root.met==1)
        {
                if(_root.trito==0)
                {
                loadMovie("3odiag.swf",invdiag);
                _root.trito=1;
                _root.deytero=0;
                _root.gr=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                        loadMovie("1odiag.swf",invdiag);
                _root.gr=1;
                _root.deytero=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }
        if (_root.met==2)
        {
                if(_root.tetarto==0)
                {
                loadMovie("4odiag.swf",invdiag);
                _root.tetarto=1;
                _root.deytero=0;
```

```
            _root.gr=0;
            _root.trito=0;
            _root.pempto=0;
            }
            else{
                    loadMovie("1odiag.swf",invdiag);
            _root.gr=1;
            _root.deytero=0;
            _root.trito=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
        }
        if(_root.met==3)
        {
            if(_root.pempto==0)
            {
            loadMovie("5odiagb.swf",invdiag);
            _root.pempto=1;
            _root.deytero=0;
            _root.gr=0;
            _root.trito=0;
            _root.tetarto=0;
            }
            else{
                    loadMovie("4odiag.swf",invdiag);
            _root.tetarto=1;
            _root.deytero=0;
            _root.gr=0;
            _root.trito=0;
            _root.pempto=0;
            }
        }

}
```

αν η πρωτοορισθείσα μεταβλητή met, λάβει την τιμή 0, τότε ορίζεται ότι η μεταβλητή pro να λάβει την τιμή 1, να ξαναεπιλεχθεί στην τύχη ένα νούμερο μεταξύ του 0 και του 3 και να αποδωθεί στη μεταβλητή met. (Τώρα θα επιλεχθεί ένας αριθμός μεταξύ τεσσάρων ακριβώς, γιατί αφήνεται απ' έξω ο αριθμός που αντιστοιχούσε στο διαγώνισμα που ήδη έχει ανοίξει, που σ' αυτή την περίπτωση θα είναι το πρώτο.) Έπειτα βλέπουμε τέσσερις βρόγχους ανάλογα με την τιμή που έχει επιλεχθεί. Αν η τιμή της met είναι 0, τότε γίνεται έλεγχος μήπως ήταν εκείνη που άνοιξε πριν. Αν δεν ήταν αυτή που άνοιξε μόλις πριν, τότε θα ανοίξει το δεύτερο διαγώνισμα και θα δοθεί στον αντίστοιχο διακόπτη «deytero», η τιμή 1. Μαζί μ' αυτά φροντίζεται να μηδενιστούν οι υπόλοιποι διακόπτες, σε περίπτωση που κάποιος απ' αυτούς είναι 1. Άν όμως ο διακόπτης deytero είχε την τιμή 1, τότε να ανοίξει ένα άλλο διαγώνισμα και να λάβει ο αντίστοιχος διακόπτης, εκείνου του διαγωνίσματος, την τιμή 1 και ο διακόπτης του υπάρχοντος διαγωνίσματος να λάβει την τιμή 0.

Οι ίδιες ενέργειες εκτελούνται, απλά ανοίγουν άλλα διαγωνίσματα και δίνονται τιμές σε διαφορετικούς διακόπτες, στην περίπτωση που η πρωτοορισθείσα met -στον δεύτερο ορισμό αριθμού- δεν λάβει την τιμή 0, αλλά λάβει την τιμή 1, 2 ή 3. Αντίστοιχα σε κάθε περίπτωση, εκτελούνται όμοιες ενέργειες και με την ίδια λογική.

```
if (de==1)
{
_root.met=random(4);

        if (_root.met==0)
        {
                if(_root.gr==0)
                {
                loadMovie("1odiag.swf",invdiag);
                _root.gr=1;
                _root.deytero=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                loadMovie("3odiag.swf",invdiag);
                _root.trito=1;
                _root.deytero=0;
                _root.gr=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }
        if (_root.met==1)
        {
                if(_root.trito==0)
                {
                loadMovie("3odiag.swf",invdiag);
                _root.trito=1;
                _root.deytero=0;
                _root.gr=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                loadMovie("2odiag.swf",invdiag);
                _root.deytero=1;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
```

```
        }
        if (_root.met==2)
        {
                if(_root.tetarto==0)
                {
                loadMovie("4odiag.swf",invdiag);
                _root.tetarto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.pempto=0;
                }
                else{
                loadMovie("2odiag.swf",invdiag);
                _root.deytero=1;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }
        if (_root.met==3)
        {
                if(_root.pempto==0)
                {
                loadMovie("5odiagb.swf",invdiag);
                _root.pempto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                }
                else{
                loadMovie("1odiag.swf",invdiag);
                _root.gr=1;
                _root.deytero=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }

}
```

Η ίδια λογική ακολουθείται, όταν με την πρώτη τυχαία επιλογή του αριθμού που θα πάρει η met, είναι το 1. Τότε ακουλουθούνται οι ίδιες ενέργειες, και γίνονται οι ίδιοι έλεγχοι και η λογική με τους διακόπτες είναι η ίδια. Η μόνη διαφορά είναι ότι φορτώνονται διαφορετικά διαγωνίσματα, και είναι άλλοι οι διακόπτες που μηδενίζονται ή παίρνουν την τιμή 1.

```
if (tr==1)
{
_root.met=random(4);

        if (_root.met==0)
        {
                if(_root.gr==0)
                {
                loadMovie("1odiag.swf",invdiag);
                _root.gr=1;
                _root.deytero=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                loadMovie("2odiag.swf",invdiag);
                _root.deytero=1;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }
        if (_root.met==1)
        {
                if(_root.deytero==0)
                {
                loadMovie("2odiag.swf",invdiag);
                _root.deytero=1;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                loadMovie("1odiag.swf",invdiag);
                _root.gr=1;
                _root.deytero=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }

        }
        if (_root.met==2)
        {
                if(_root.tetarto==0)
                {
                loadMovie("4odiag.swf",invdiag);
```

```
                _root.tetarto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.pempto=0;
                }
                else{
                loadMovie("5odiagb.swf",invdiag);
                _root.pempto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                }
        }
        if (_root.met==3)
        {
                if(_root.pempto==0)
                {
                loadMovie("5odiagb.swf",invdiag);
                _root.pempto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                }
                else{
                loadMovie("2odiag.swf",invdiag);
                _root.deytero=1;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }

}
```

πιό πάνω παρατίθεται και ο κώδικας, αν η met πάρει με την πρώτη, τιμή 2. Πάλι επίσης εκτελούνται ενέργειες ίδιας λογικής.

```
if (te==1)
{
_root.met=random(4);


        if (_root.met==0)
        {
                if(_root.gr==0)
                {
```

```
            loadMovie("1odiag.swf",invdiag);
            _root.gr=1;
            _root.deytero=0;
            _root.trito=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
            else{
            loadMovie("4odiag.swf",invdiag);
            _root.tetarto=1;
            _root.deytero=0;
            _root.gr=0;
            _root.trito=0;
            _root.pempto=0;
            }
}
if (_root.met==1)
{
            if(_root.deytero==0)
            {
            loadMovie("2odiag.swf",invdiag);
            _root.deytero=1;
            _root.gr=0;
            _root.trito=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
            else{
            loadMovie("3odiag.swf",invdiag);
            _root.trito=1;
            _root.deytero=0;
            _root.gr=0;
            _root.tetarto=0;
            _root.pempto=0;
            }

}
if (_root.met==2)
{
            if(_root.trito==0)
            {
            loadMovie("3odiag.swf",invdiag);
            _root.trito=1;
            _root.deytero=0;
            _root.gr=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
            else{
            loadMovie("5odiagb.swf",invdiag);
```

```
                _root.pempto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                }
        }
        if (_root.met==3)
        {
                if(_root.pempto==0)
                {
                loadMovie("5odiagb.swf",invdiag);
                _root.pempto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.tetarto=0;
                }
                else{
                loadMovie("3odiag.swf",invdiag);
                _root.trito=1;
                _root.deytero=0;
                _root.gr=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }
}
```

ίδια και η λογική όταν η met πάρει πρώτη φορά την τιμή 3.

```
if (pe==1)
{
_root.met=random(4);

        if (_root.met==0)
        {
                if(_root.gr==0)
                {
                loadMovie("1odiag.swf",invdiag);
                _root.gr=1;
                _root.deytero=0;
                _root.trito=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
                else{
                loadMovie("5odiagb.swf",invdiag);
                _root.pempto=1;
                _root.deytero=0;
```

```
            _root.gr=0;
            _root.trito=0;
            _root.tetarto=0;
            }
    }
    if (_root.met==1)
    {
            if(_root.deytero==0)
            {
            loadMovie("2odiag.swf",invdiag);
            _root.deytero=1;
            _root.gr=0;
            _root.trito=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
            else{
            loadMovie("5odiagb.swf",invdiag);
            _root.pempto=1;
            _root.deytero=0;
            _root.gr=0;
            _root.trito=0;
            _root.tetarto=0;
            }
    }
    if (_root.met==2)
    {
            if(_root.trito==0)
            {
            loadMovie("3odiag.swf",invdiag);
            _root.trito=1;
            _root.deytero=0;
            _root.gr=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
            else{
            loadMovie("1odiag.swf",invdiag);
            _root.gr=1;
            _root.deytero=0;
            _root.trito=0;
            _root.tetarto=0;
            _root.pempto=0;
            }
    }
    if (_root.met==3)
    {
            if(_root.tetarto==0)
            {
            loadMovie("4odiag.swf",invdiag);
```

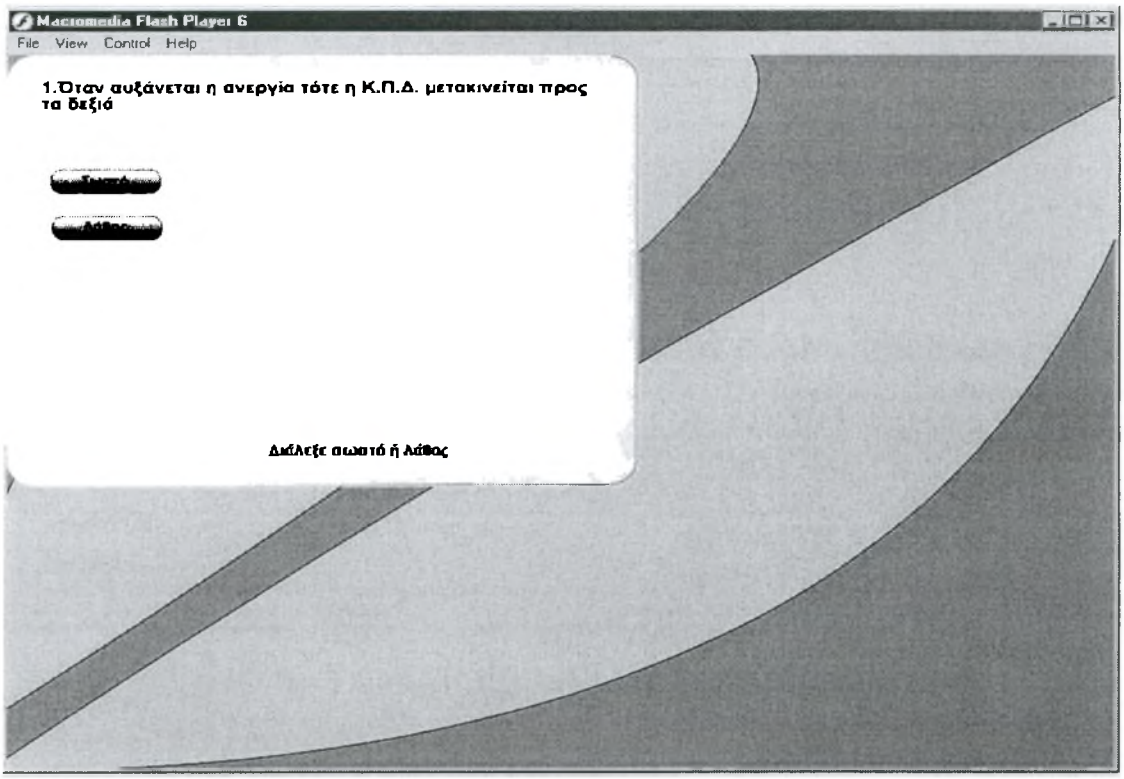
```
                _root.tetarto=1;
                _root.deytero=0;
                _root.gr=0;
                _root.trito=0;
                _root.pempto=0;
                }
                else{
                loadMovie("3odiag.swf",invdiag);
                _root.trito=1;
                _root.deytero=0;
                _root.gr=0;
                _root.tetarto=0;
                _root.pempto=0;
                }
        }

}
}
```

και τέλος δείχνεται και η περίπτωση που η met θα λάβει πρώτη φορά την τιμή 4.

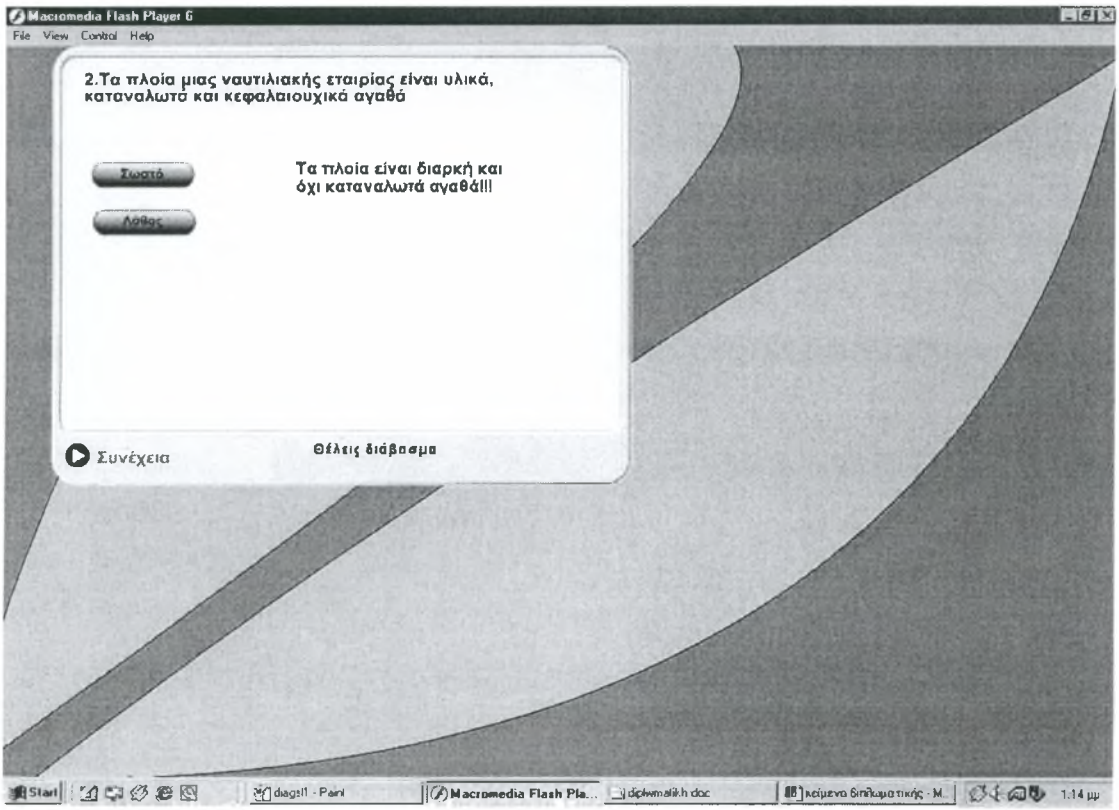
Τώρα να δούμε πως λειτουργεί ένα διαγώνισμα. Για τα διαγωνίσματα δεν προτιμήθηκαν τα έτοιμα υποδείγματα του flash, αλλά δημιουργήθηκαν εξ'αρχής ολότελα διαφορετικά, γιατί κρίθηκε ότι αυτά που περιείχαν τα έτοιμα υποδείγματα του flash, δεν επαρκούσαν για να καλύψουν τις υπάρχουσες ανάγκες. Κατ'αρχήν, το διαγώνισμα είναι χωρισμένο σε τέσσερα τμήματα, έτσι ώστε να μοιάζει όσο το δυνατόν περισσότερο τις πανελλήνιες εξετάσεις, για την εισαγωγή στην ανώτερη και ανώτατη εκπαίδευση.

Ερωτήσεις θεωρίας

Το πρώτο τμήμα περιλαμβάνει ερωτήσεις σωστού – λάθους, όπου ο μαθητής έχει τη δυνατότητα να επιλέξει το κουμπί που αυτός θεωρεί κατάλληλο και οι οποίες έχουν την παρακάτω μορφή.



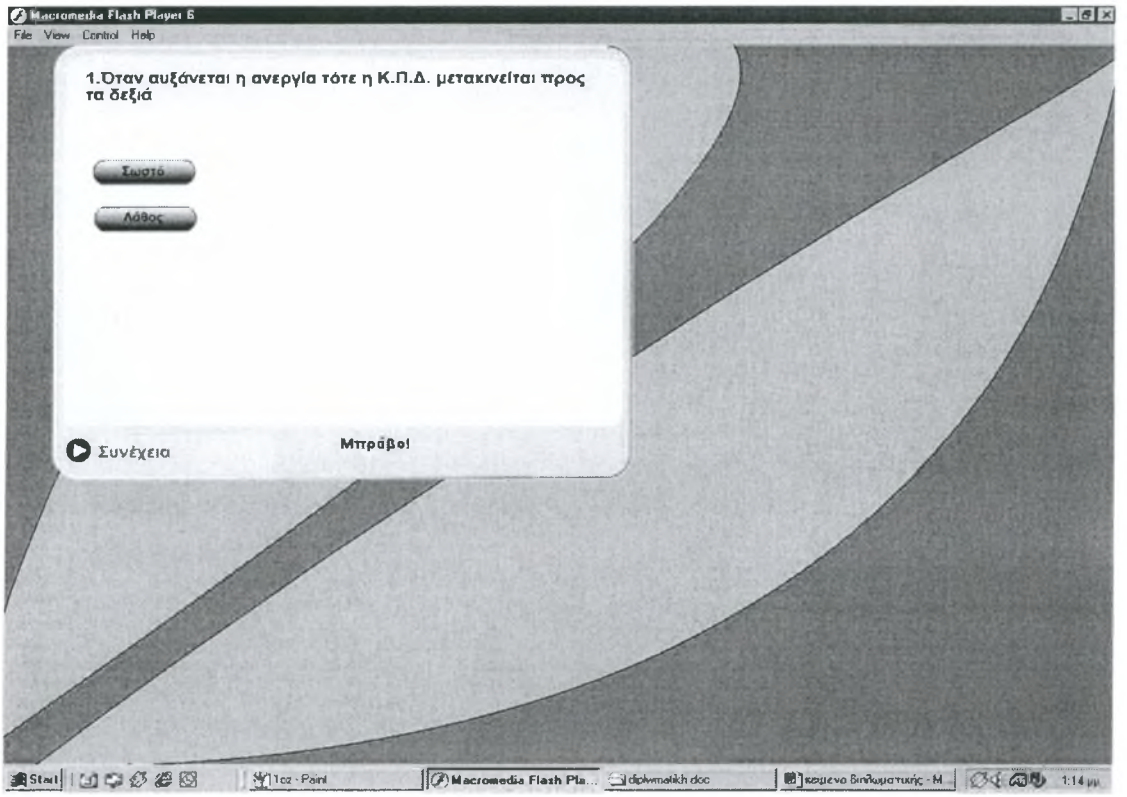
Όποιο κουμπί και να πιέσει βγαίνουν αντίστοιχα μηνύματα, ανάλογα με το αν η απάντησή του ήταν εύστοχη ή όχι. Σε περίπτωση που ο χρήστης δεν δώσει εύστοχη απάντηση, εμφανίζεται ένα πλαίσιο κειμένου που εξηγεί το λάθος του χρήστη.



on (release) {

```
        feedback.text="Θέλεις διάβασμα!";
        swsto.text="Το σωστό είναι μεταξύ σταθερών και μεταβητών παραγωγικών
συντελεστών.";
nextFrame();
}
```

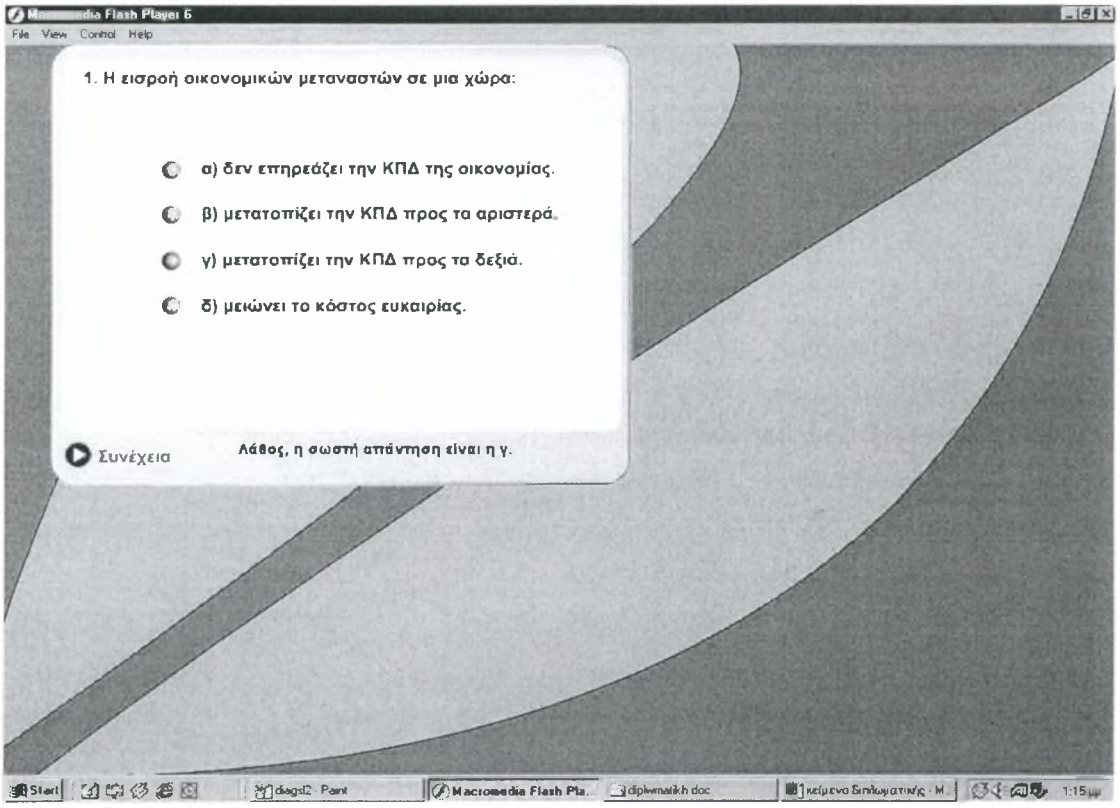
με τον παραπάνω κώδικα βλέπουμε πώς γίνεται ο έλεγχος της εύστοχης απάντησης και τα μηνύματα που προκύπτουν σε περίπτωση που ο χρήστης δεν απαντήσει εύστοχα.



Στην αρχή των ερωτήσεων θεωρίας, έχει οριστεί μια μεταβλητή «w», προκειμένου να μετράει την βαθμολογία όταν ο χρήστης απαντά εύστοχα.

```
on (release) {
        feedback.text="Μπράβο!";
        gotoAndPlay(_currentframe+2);
        w=w+1;
}
```

Μετά από τέσσερις ερωτήσεις σωστού – λάθους, ακολουθούν τέσσερις ερωτήσεις πολλαπλής επιλογής. Ο χρήστης επιλέγει μια από τις τέσσερις απαντήσεις και σε περίπτωση λάθους υποδεικνύεται η σωστή απάντηση και προκύπτουν και τα αντίστοιχα μηνύματα.




```
on (release) {
        feedback.text="Η σωστή απάντηση είναι η γ.";
        nextFrame();
}
```

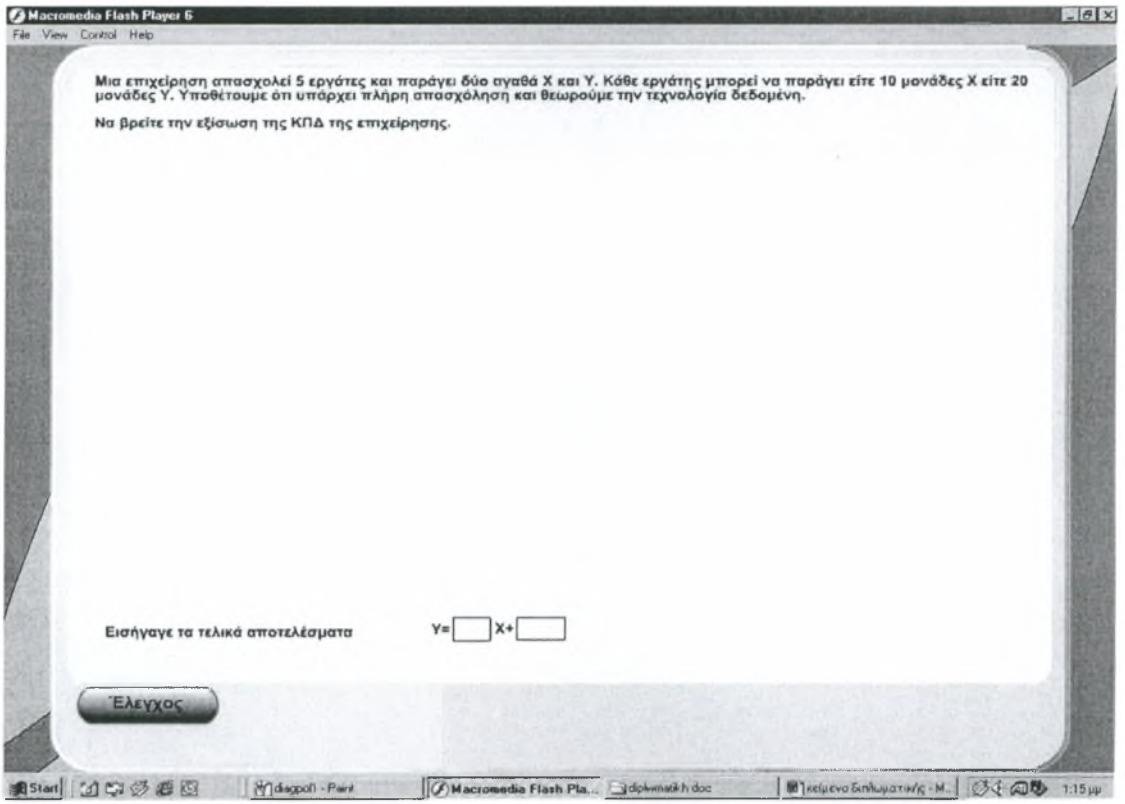
ενώ σε περίπτωση που ο χρήστης απαντήσει σωστά εκτελείται ο κώδικας:

```
on (release) {
        feedback.text=" Μπράβο!";
        w=w+1;
        gotoAndPlay(_currentframe+2);
}
```

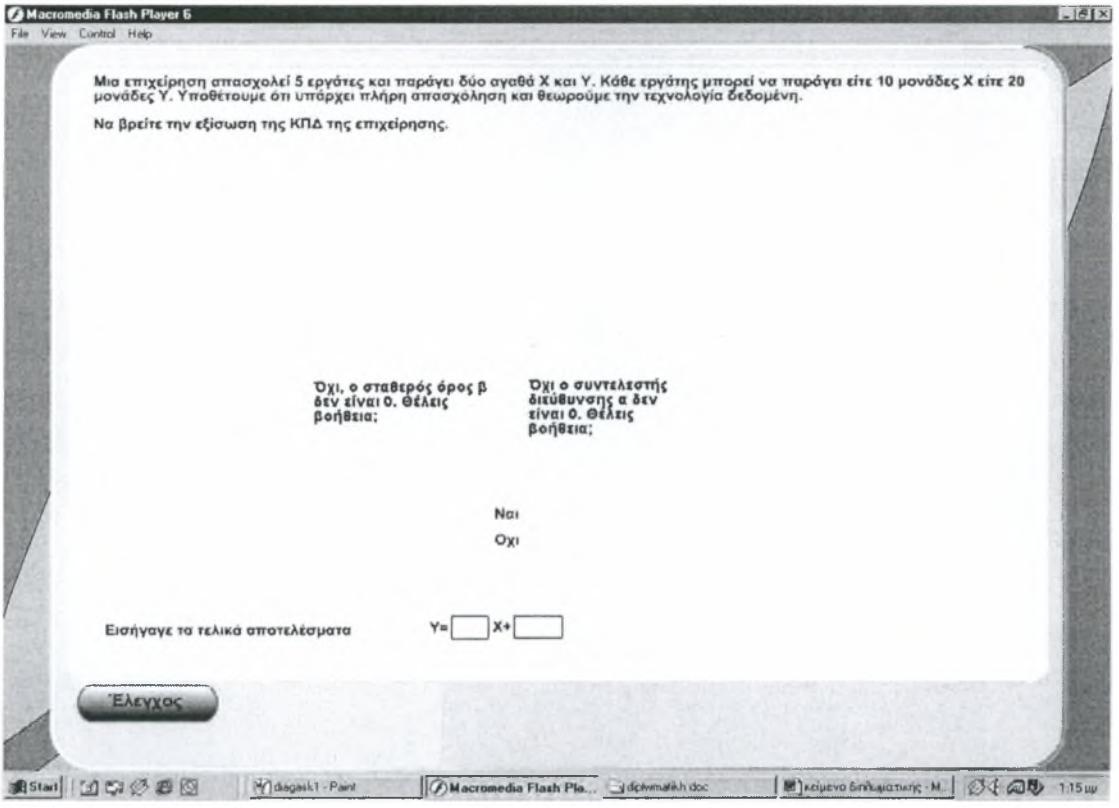
όπου η μεταβλητή w αυξάνεται κατά ένα.

Ασκήσεις

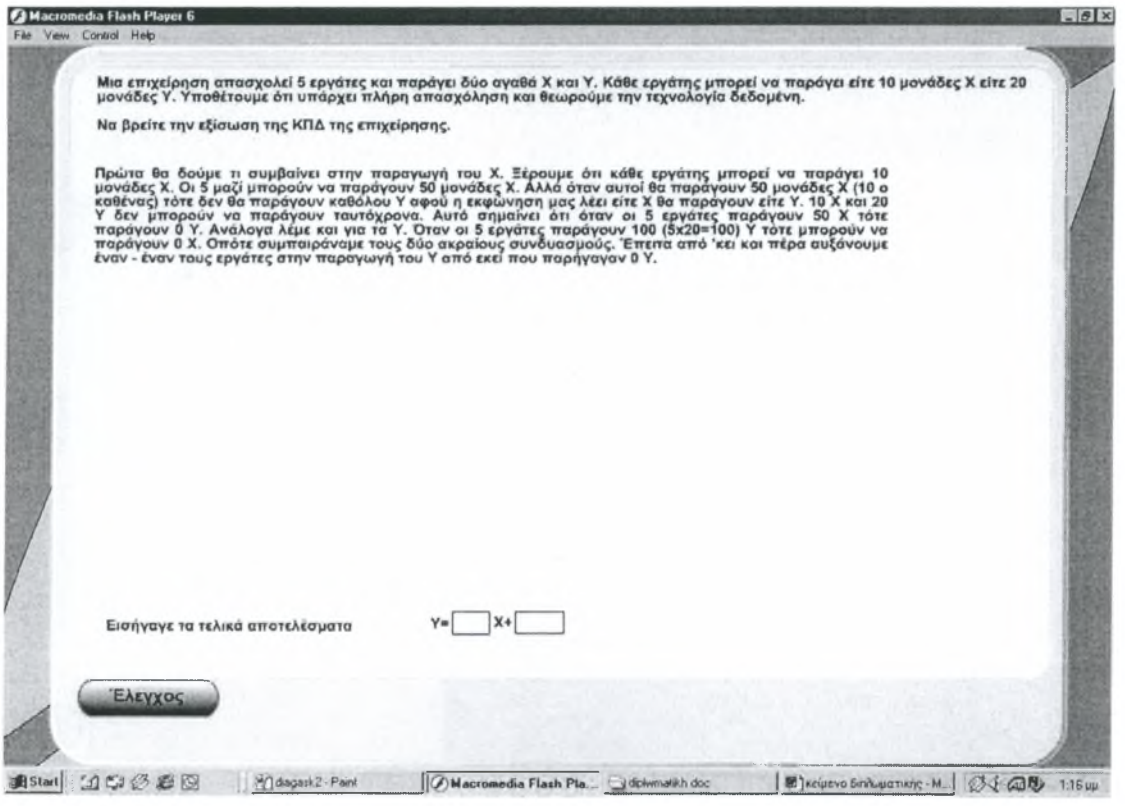
Μόλις τελειώσουν οι ερωτήσεις θεωρίας ξεκινούν οι ασκήσεις. Ορίζεται μια νέα μεταβλητή για την μέτρηση της βαθμολογίας και δίνονται από ένα έως έξι πεδία εισαγωγής κειμένου (input texts) ανάλογα με την άσκηση, ώστε ο χρήστης να εισάγει την απάντησή του, είτε αυτή είναι αριθμός είτε λέξη.



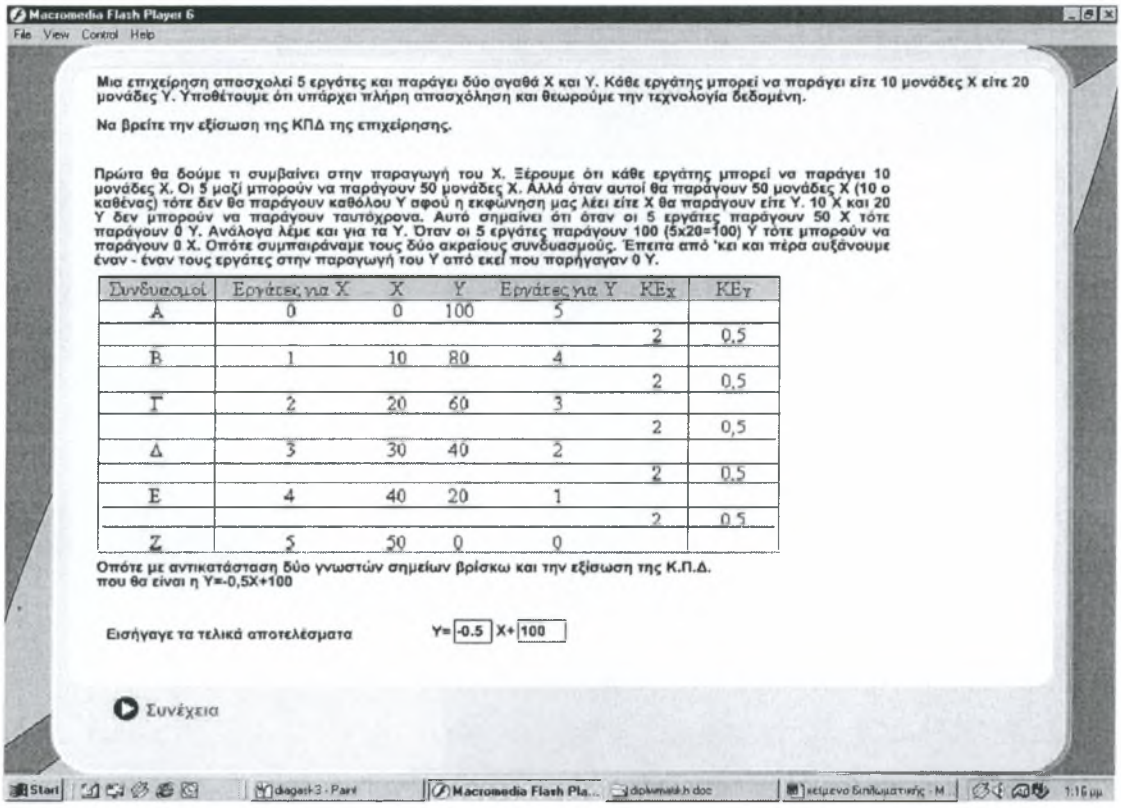
Οι ασκήσεις δεν αρκούνται απλά στο να ελέγξουν μια φορά την τελική απάντηση του χρήστη και να βαθμολογηθεί. Αντιθέτως, αν ο χρήστης δεν εισάγει ή εισάγει λάθος αποτέλεσμα, προκύπτει ένα μήνυμα που του κάνει γνωστό ότι κάνει λάθος και του προσφέρει την ευκαιρία αν θέλει να έχει κάποια βοήθεια. Ταυτόχρονα εμφανίζονται και δύο κουμπιά, για να επιλέξει ο χρήστης να δεχθεί η όχι την προτεινόμενη βοήθεια. Αυτό φαίνεται στην παρακάτω εικόνα:



Αν ο χρήστης δεν θέλει βοήθεια, τότε τα πεδία κειμένου και τα κουμπιά εξαφανίζονται και ο χρήστης έχει μπροστά του την ίδια οθόνη με πριν. Αν επιλέξει ότι θέλει βοήθεια, τότε εμφανίζεται ένα μικρό μέρος της λύσης, έτσι επιλεγμένο, ώστε να μπορέσει ο χρήστης να βοηθηθεί όπως φαίνεται παρακάτω.



Αν εισάγει τώρα ο χρήστης τις σωστές απαντήσεις, τότε θα λάβει μικρότερη βαθμολογία από τους βαθμούς που θα κέρδιζε, αν απαντούσε σωστά χωρίς βοήθεια. Αν ο χρήστης απαντήσει σωστά, εμφανίζεται η λύση της άσκησης, προκειμένου ο χρήστης να δει τη λύση, σε περίπτωση που διαφέρει με τον τρόπο που ο χρήστης έχει λύσει την άσκηση. Αυτό γιατί όπως γνωρίζουμε μια άσκηση λύνεται με πολλούς τρόπους και στόχος είναι, ακόμα και όταν ο μαθητής λύσει μια άσκηση, να δεί και κάποιον άλλο τρόπο αν αυτός είναι πιό εύκολος. Στην περίπτωση που ο χρήστης απαντήσει και τις δύο φορές λανθασμένα, εμφανίζεται η λύση της άσκησης, για να δει ο χρήστης που έκανε λάθος και η εφαρμογή προχωρά στην επόμενη άσκηση όπως φαίνεται ακολούθως



Συνδυασμοί	Εργάτες για Χ	Χ	Υ	Εργάτες για Υ	KE_X	KE_Y
Α	0	0	100	5		
					2	0,5
Β	1	10	80	4		
					2	0,5
Γ	2	20	60	3		
					2	0,5
Δ	3	30	40	2		
					2	0,5
Ε	4	40	20	1		
					2	0,5
Ζ	5	50	0	0		

Στην αρχή στο πρώτο καρέ, που αφορά τη συγκεκριμένη άσκηση, δηλώνονται οι μεταβλητές που θα χρησιμεύσουν ως διακόπτες για τους ελέγχους των απαντήσεων.

```
int (i);
i=0;
int(g);
g=0;
int(h);
h=0;
int(m);
m=0;
int(f);
f=0;
int(u);
u=0;
```

```
int(v);
v=0;
int(o);
o=0;
int(r);
r=0;
int(t);
t=0;
int(x);
x=0;
int(at);
at=0;
int(ax);
ax=0;
nai._visible=false;
oxi._visible=false;
lysha._visible=false;
lyshb._visible=false;
graph._visible=false;
```

επίσης η λύση και τα κουμπιά επιλογής βοήθειας προς το παρόν είναι κρυμμένα, περιμένοντας την διάδραση του χρήστη για να εμφανιστούν.
Παρακάτω παρατίθεται ο κώδικας του κουμπιού ελέγχου:

```
on (release) {
        int(synt);
        synt=(Number(z));
        int (stath) ;
        stath=(Number(k));
        int (p) ;
        p=(Number(q));
        int (qo) ;
        qo=(Number(qistonos));
        int (se) ;
        se=(Number(dse));
        int (dy) ;
        dy=(Number(dyy));
```

στην αρχή ορίζονται οι μεταβλητές που θα παίζουν το ρόλο της απάντησης και σ' αυτές θα γίνεται ο έλεγχος, αν είναι σωτές ή όχι. Αυτές οι μεταβλητές λαμβάνουν την τιμή που θα ορίσει ο χρήστης στο αντίστοιχο πεδίο εισαγωγής κειμένου. Επειδή αυτό που θα εισάγει ο χρήστης στο πεδίο εισαγωγής κειμένου, αναγνωρίζεται σαν string, είναι απαραίτητο να προσθέσουμε την εντολή Number(z), έτσι ώστε το string να μετατρέπεται σε αριθμό. Η μεταβλητή z είναι η μεταβλητή του αντίστοιχου πεδίου κειμένου και λαμβάνει σαν τιμή, το κείμενο σε μορφή string, που θα εισάγει ο χρήστης. Η τιμή αυτής της μεταβλητής z μετατρέπεται σε αριθμό και αποδίδεται στην μεταβλητή synt.

```
if (p==180&&stath==-3&&synt==80&&qo==2&&se==1500&&dy==27.7)
        {
```

```
            lysha._visible=true;
            lyshb._visible=true;
            swsto.text="Μπράβο!!";
            nextFrame();
        }
```

στον παραπάνω κώδικα ελέγχεται η περίπτωση που ο χρήστης, με την πρώτη, εισάγει όλες τις απαντήσεις σωστά.(η συγκεκριμένη άσκηση έχει έξι απαντήσεις). Επομένως αν οι απαντήσεις είναι σωστές, τότε εμφανίζεται ολόκληρη η λύση, η οποία είναι χωρισμένη στην αρχική βοήθεια (lysha) και στην υπόλοιπη λύση (lyshb), και η εφαρμογή στέλνει τον χρήστη στη σελίδα της τελικής βαθμολογίας. Σε περίπτωση που ο χρήστης έχει κάποιες από τις απαντήσεις σωστές, τότε πρέπει να εξεταστεί κάθε απάντηση ξεχωριστά, αν θα δοθεί σωστά από το χρήστη κι αν ναι, τότε πρέπει να εξετασθεί αν αυτή δόθηκε εύστοχη μετά τη χρήση της βοήθειας ή πριν.

```
        if (p==180)
        {
            if(i==0)
            {
                if(m==0)
                {
                n=n+0.5
                }
            m=1;
            }
            if (i==1)
            {
                if (g!==1)
                {
                    if(m!==1)
                    {
                    n=n+0.2;
                    }
                    g=1;
                }
                else if(m==1)
                {
                n=n;
                }
            }
        feed1.text = "Όντως ο σταθερός όρος της συνάρτησης ζήτησης είναι 180";
            }
        else if (p!==180)
        {
            if (i==1)
            {
            feed1.text =" Όχι ο σταθερός όρος της συνάρτησης ζήτησης δεν είναι
180"+p+"Θέλεις τη λύση;";
            nai._visible=true;
            oxi._visible=true;
```

```
            }
            else
            {
            feed1.text =" Όχι ο σταθερός όρος της συνάρτησης ζήτησης δεν είναι
180"+p+"Θέλεις βοήθεια;";
            nai._visible=true;
            oxi._visible=true;
                    }
        }
```

Αν η απάντηση του χρήστη είναι σωστή τότε αμέσως ελέγχεται αν ο χρήστης χρησιμοποίησε τη βοήθεια ή όχι. Αν ο χρήστης έχει χρησιμοποιήσει τη βοήθεια, τότε δίνεται στη μεταβλητή i η τιμή 1. Οπότε αν ο χρήστης απάντησε σωστά και δεν έχει λάβει βοήθεια, τότε γίνεται ένας τρίτος έλεγχος, μήπως και του έχουν δοθεί ήδη οι βαθμοί της σωστής απάντησης. Αυτό γίνεται γιατί, αν ο χρήστης έδινε μια σωστή απάντηση και όχι όλες σωστές, τότε κάθε φορά που θα πατούσε το κουμπί ελέγχου, η βαθμολογία του θα αυξανόταν. Οπότε, χρειάστηκε ένας διακόπτης, στην προκειμένη περίπτωση η μεταβλητή m, για να λαμβάνει την τιμή 1 μόλις αποδοθούν στο χρήστη οι βαθμοί της σωστής, άνευ βοήθειας, απάντησης. Άν τελικά ο χρήστης απάντησε σωστά με την πρώτη, γίνεται ο έλεγχος μήπως του έχουν ήδη αποδοθεί οι βαθμοί if(m==0), και αν δεν του έχουν αποδοθεί, τότε του αποδόδονται οι ανάλογοι βαθμοί και ο διακόπτης ελέγχου m λαμβάνει την τιμή 1, έτσι ώστε αν ξαναπατηθεί το κουμπί ελέγχου, να μην ξαναποδοθούν στο χρήστη άλλοι βαθμοί. Απ' την άλλη, αν ο χρήστης έχει χρησιμοποιήσει την βοήθεια τότε θα λάβει τους λιγότερους βαθμούς, που αντιστοιχούν στη σωστή απάντηση, αλλά με χρήση της βοήθειας. Πρώτα όμως θα εξεταστεί, αν ήδη του έχουν δοθεί αυτοί οι βαθμοί με τη χρήση ενός διακόπτη πάλι. Αυτός ο διακόπτης είναι η μεταβλητή g και έχει ακριβώς την ίδια λειτουργία με τη μεταβλητή m στην προηγούμενη περίπτωση. Όταν ο χρήστης δηλαδή έχοντας λάβει τη βοήθεια, εισάγει τη σωστή τιμή στο πεδίο εισαγωγής κειμένου, με το πάτημα του κουμπιού ελέγχου, πρώτα ελέγχεται αν ήδη έχει πάρει τους βαθμούς που του αντιστοιχούν if(g==0), και αν δεν τους έχει πάρει, τότε τους λαμβάνει, και στον διακόπτη g τίθεται η τιμή 1, έτσι ώστε αν ξαναπατηθεί το κουμπί ελέγχου, να μην ξαναδοθούν βαθμοί στον χρήστη. Επίσης στην περίπτωση που ο χρήστης εισάγει τη σωστή απάντηση στο συγκεκριμένο ερώτημα, άσχετα αν χρησιμοποιήσε βοήθεια ή όχι, εμφανίζεται ένα μήνυμα ότι η συγκεκριμένη απάντηση ήταν σωστή.

Σε περίπτωση τώρα, που ο χρήστης δώσει λάθος απάντηση, τότε προκύπτει ένα μήνυμα που τον ενημερώνει ότι έκανε λάθος και αν δεν έχει λάβει βοήθεια, του τίθεται η ερώτηση αν θέλει βοήθεια. Αν έχει λάβει βοήθεια, τότε του τίθεται το ερώτημα αν επιθυμεί τη λύση. Και στις δύο περιπτώσεις είτε έλαβε την βοήθεια είτε όχι, εμφανίζονται τα κουμπιά αποδοχής της βοήθειας ή της λύσης, ανάλογα με την περίπτωση.

```
if (stath==-3)
{
        if (i==0)
        {
                if(f==0)
                {
                n=n+0.5;
                }
```

```
                f=1;
                }
                if (i==1)
                {
                        if (h!==1)
                                {
                                if(f!==1)
                                {
                                n=n+0.2;
                                }
                                h=1;
                                }
                        else if(f==1)
                        {
                        n=n;
                        }
                }
        feed2.text="Όντως ο συντελεστής διεύθυνσης της συνάρτησης ζήτησης είναι -
3";

                }
        else if (stath!==-3)
        {
                if (i==1)
                {
                feed2.text =" Όχι ο συντελεστής διεύθυνσης της συνάρτησης ζήτησης
δεν είναι "+stath+". Θέλεις τη λύση;";
                nai._visible=true;
                oxi._visible=true;
                }
                else
                {
                feed2.text =" Όχι ο συντελεστής διεύθυνσης της συνάρτησης ζήτησης
δεν είναι "+stath+". Θέλεις βοήθεια;";
                nai._visible=true;
                oxi._visible=true;
                }
        }
```

η ίδια λογική ισχύει και για το δεύτερο πεδίο εισαγωγής κειμένου, που αφορά την δεύτερη απάντηση, με τη διαφορά ότι εδώ οι διακόπτες είναι ο f αντί του m και ο h αντί του g. Επίσης διαφέρουν και τα μηνύματα που ανταποκρίνονται στον χρήστη, γιατί πρόκειται για διαφορετικό ερώτημα της άσκησης.

```
if (synt==80)
        {
                if (i==0)
                {
                        if(u==0)
                        {
```

```
                n=n+0.5;
                }
        u=1;
        }
        if (i==1)
        {
                if (v!==1)
                        {
                        if(u!==1)
                        {
                        n=n+0.2;
                        }
                        v=1;
                        }
                else if(u==1)
                {
                n=n;
                }
        }
feed3.text="Όντως ο σταθερός όρος της συνάρτησης προσφοράς είναι 80";

        }
else if (synt!==80)
{
        if (i==1)
        {
        feed3.text =" Όχι ο σταθερός όρος της συνάρτησης προσφοράς δεν
είναι synt+". Θέλεις τη λύση;";
        nai._visible=true;
        oxi._visible=true;
        }
        else
        {
        feed3.text =" Όχι ο σταθερός όρος της συνάρτησης προσφοράς δεν
είναι "+synt+". Θέλεις βοήθεια;";
        nai._visible=true;
        oxi._visible=true;
        }
}
```

στο τρίτο πεδίο εισαγωγής κειμένου που αφορά την τρίτη απάντηση, επικρατεί η ίδια φιλοσοφία πάλι μόνο που τη θέση του m την έλαβε ο u και τη θέση του g ο v.

```
if (qo==2)
        {
                if (i==0)
                {
                        if(o==0)
                        {
                        n=n+0.5;
```

```
                }
            o=1;
            }
            if (i==1)
            {
                if (r!==1)
                    {
                    if(o!==1)
                    {
                    n=n+0.2;
                    }
                    r=1;
                    }
                else if(o==1)
                {
                n=n;
                }
            }
        feed4.text=" Όντως ο συντελεστής διεύθυνσης της συνάρτησης προσφοράς
είναι 2";
            }
        else if (qo!==2)
        {
            if (i==1)
            {
            feed4.text =" Όχι ο συντελεστής διεύθυνσης της συνάρτησης
προσφοράς δεν είναι "+qo+". Θέλεις τη λύση;";
            nai._visible=true;
            oxi._visible=true;
            }
            else
            {
            feed4.text =" Όχι ο συντελεστής διεύθυνσης της συνάρτησης
προσφοράς δεν είναι "+qo+". Θέλεις βοήθεια;";
            nai._visible=true;
            oxi._visible=true;
            }
        }
```

Εδώ τη θέση του διακόπτη m την έλαβε ο «ο» και τη θέση του g ο «r».

```
if (se==1500)
        {
            if (i==0)
            {
                if(t==0)
                {
                n=n+0.5;
                }
            t=1;
```

```
                }
                if (i==1)
                {
                        if (x!==1)
                                {
                                if(t!==1)
                                {
                                n=n+0.2;
                                }
                                x=1;
                                }
                        else if(t==1)
                        {
                        n=n;
                        }
                }
        feed5.text="Η επιβάρυνση του κράτους όντως είναι 1500";

                }
        else if (se!==1500)
        {
                if (i==1)
                {
                feed5.text ="Όχι η επιβάρυνση του κράτους δεν είναι "+se+". Θέλεις
τη λύση;";
                nai._visible=true;
                oxi._visible=true;
                }
                else
                {
                feed5.text =" Όχι η επιβάρυνση του κράτους δεν είναι "+se+". Θέλεις
βοήθεια;";
                nai._visible=true;
                oxi._visible=true;
                }
        }
```

Εδώ τη θέση του διακόπτη m την έλαβε ο «t» και τη θέση του g ο «x».

```
if (dy==27.7)
        {
                if (i==0)
                {
                        if(at==0)
                        {
                        n=n+1.5;
                        }
                at=1;
                }
                if (i==1)
```

```
                {
                        if (ax!==1)
                                {
                                if(at!==1)
                                {
                                n=n+1;
                                }
                                ax=1;
                                }
                        else if(t==1)
                        {
                        n=n;
                        }
                }
        feed6.text="Η ΔΥ% όντως είναι 27,7%";

                }
        else if (dy!==27.7)
        {
                if (i==1)
                {
                feed6.text ="Όχι η ΔΥ% δεν είναι "+dy+". Θέλεις τη λύση;";
                nai._visible=true;
                oxi._visible=true;
                }
                else
                {
                feed6.text =" Όχι η ΔΥ% δεν είναι "+dy+". Θέλεις βοήθεια;";
                nai._visible=true;
                oxi._visible=true;
                }
        }
}
```

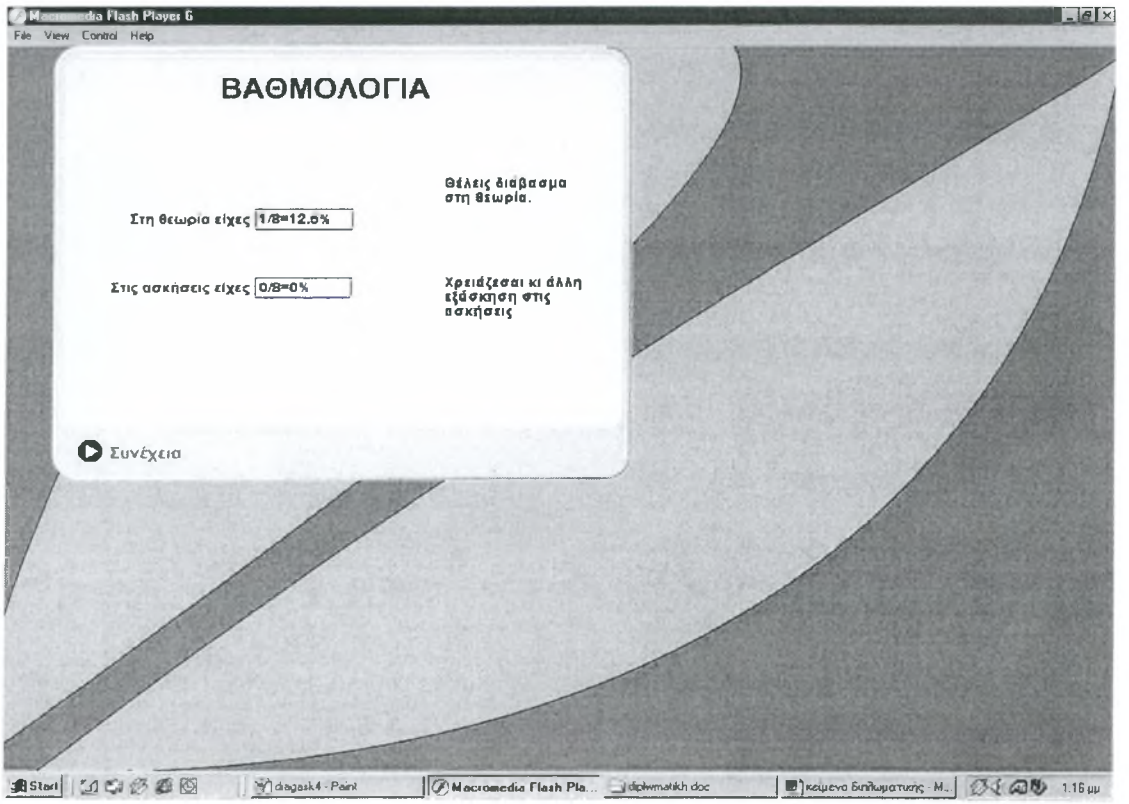
Εδώ τη θέση του διακόπτη m την έλαβε ο «at» και τη θέση του g ο «ax».
Παρακάτω παρατίθεται ο κώδικας του κουμπιού αποδοχής της βοήθειας/λύσης:

```
on (release)
{
        i=i+1;
        nai._visible=false;
        oxi._visible=false;
        feed1.text="";
        feed2.text="";
        feed3.text="";
        feed4.text="";
        feed5.text="";
        feed6.text="";

        if (i==1)
```

```
        {
        lysha._visible=true;
        }
        if (i==2)
        {
        lyshb._visible=true;
        q=180;
        k=-3;
        z=80;
        qistonos=2;
        dse=1500;
        dyy=27.7;
        nextFrame();
        }
}
```

στην αρχή ορίζεται ότι με το πάτημα του κουμπιού, η μεταβλητή i να λαμβάνει την τιμή 1. Επίσης ορίζεται να εξαφανίζονται τα κουμπιά αποδοχής και απόρριψης της βοήθειας/λύσης, και τα δυναμικά πεδία κειμένου να μην έχουν κείμενο. Έπειτα γίνεται ο έλεγχος πόσες φορές έχει πατηθεί το κουμπί αποδοχής βοήθειας if(i==1). Αν έχει πατηθεί μία φορά, τότε εμφανίζεται η βοήθεια και το i αυξάνεται κατά 1. Αν έχει πατηθεί δεύτερη φορά, τότε εμφανίζεται όλη η λύση και στα πεδία εισαγωγής κειμένου, ενφανίζονται οι σωστές απαντήσεις και η εφαρμογή προχωράει στο επόμενο καρέ, που είτε είναι η επόμενη άσκηση είτε είναι η τελική βαθμολογία.

Στο τελευταίο καρέ κάθε διαγωνίσματος περιλαμβάνεται η τελική βαθμολογία του χρήστη, όσον αφορά τη θεωρία και τις ασκήσεις. Ξεχωριστά βαθμολογείται στη θεωρία και ξεχωριστά στις ασκήσεις όπως ακριβώς δείχνονται στην εικόνα παρακάτω.



Στο ίδιο το καρέ εμπεριέχεται ο κώδικας:

```
stop();
int(ko);
ko=w+n;
thewria.text=w+"/8="+w*100/8+"%";
bathmos.text=n+"/8="+n*100/8+"%";
if (w<6)
{
        diabasma.text="Θέλεις διάβασμα στη θεωρία.";
        }
if (w==6||w==7)
{
        diabasma.text="Καλά τα πας στη θεωρία.";
        }
        if (w==8)
{
        diabasma.text="Μπράβο!";
        }
if (n<6)
{
        exaskisi.text="Χρειάζεσαι κι άλλη εξάσκηση στις ασκήσεις";
        }
if (5<n&&n<8)
{
        exaskisi.text="Καλά τα πας στις ασκήσεις.";
        }
        if (n==8)
{
        exaskisi.text="Μπράβο!";
        }
```

στην αρχή ορίζεται η μεταβλητή ko η οποία είναι το άθροισμα των μεταβλητών n και w. Η w μετρούσε τη βαθμολογία του χρήστη στη θεωρία και η n στις ασκήσεις. Σε ένα δυναμικό πεδίο κειμένου εμφανίζεται η βαθμολογία w ως ποσοστό επί τοις εκατό, και το ίδιο γίνεται για την βαθμολογία των ασκήσεων n. Έπειτα έχουμε τους ελέγχους, όπου ανάλογα με τον αριθμό της μεταβλητής w, εμφανίζοτναι και διάφορα μηνύματα σχετικά με το πόσο αποτελεσματικός ήταν ο χρήστης στη θεωρία. Το ίδιο γίνεται και με τις ασκήσεις, μόνο που εδώ οι έλεγχοι γίνονται στη μεταβλητή n. Στο τέλος, τελείται η κρίση για το αν ο χρήστης μπορεί να συνεχίσει σε πιό δύσκολο διαγώνισμα. Αυτός ο έλεγχος τελείται με το πάτημα ενός κουμπιού. Ο κώδικας που τελεί τον έλεγχο είναι :

```
on (release) {
        if(ko<13.5)
        {
                feedback.text="Για να συνεχίσεις σε πιο δύσκολο διαγώνισμα χρειάζεσαι κι άλλο διάβασμα. Ξαναδιάβασε τη θεωρία και ξαναπροσπάθησε.";
                proceed._visible=false;
```

```
                }

        else if (ko>=13.5)
        {
                feedback.text="Επειδή η βαθμολογία σου ήταν καλή μπορείς να
συνεχίσεις σε πιό δύσκολο διαγώνισμα";
                proceed._visible=false;
                proceed2._visible=true;
                }
}
```

Αν η μεταβλητή ko είναι πάνω από έναν συγκεκριμένο βαθμό, τότε εμφανίζεται το ανάλογο μήνυμα, καθώς επίσης ένα κουμπί το οποίο μπορεί να σε παραπέμψει στο πιό δύσκολο διαγώνισμα. Αν η τιμή της μεταβλητής ko είναι κάτω από κάποιον αριθμό, τότε αυτό το κουμπί δεν εμφανίζεται, αλλά εμφανίζεται ένα μήνυμα.
Σε περίπτωση που ο χρήστης έχει ικανοποιητική βαθμολογία, όπως είπαμε, θα εμφανιστεί το κουμπί πρόσβασης στο πιό δύσκολο διαγώνισμα. Όπως είπαμε το 2° επίπεδο δυσκολίας περιλαμβάνει άλλα τέσσερα διαγωνίσματα. Οπότε χρησιμοποιείται κώδικας για να επιλεγεί τυχαία ένα από τα τέσσερα.

```
on (release) {
        int(bir);
        bir=random(4);
        unloadMovie("_parent.invdiag");

        if(bir==0)
        {
                loadMovie("6odiag.swf",_parent.invdiag);
                }
        if(bir==1)
        {
                loadMovie("7odiag.swf",_parent.invdiag);
                }
        if(bir==2)
        {
                loadMovie("8odiag.swf",_parent.invdiag);
                }
        if(bir==3)
        {
                loadMovie("9odiag.swf",_parent.invdiag);
                }
}
```

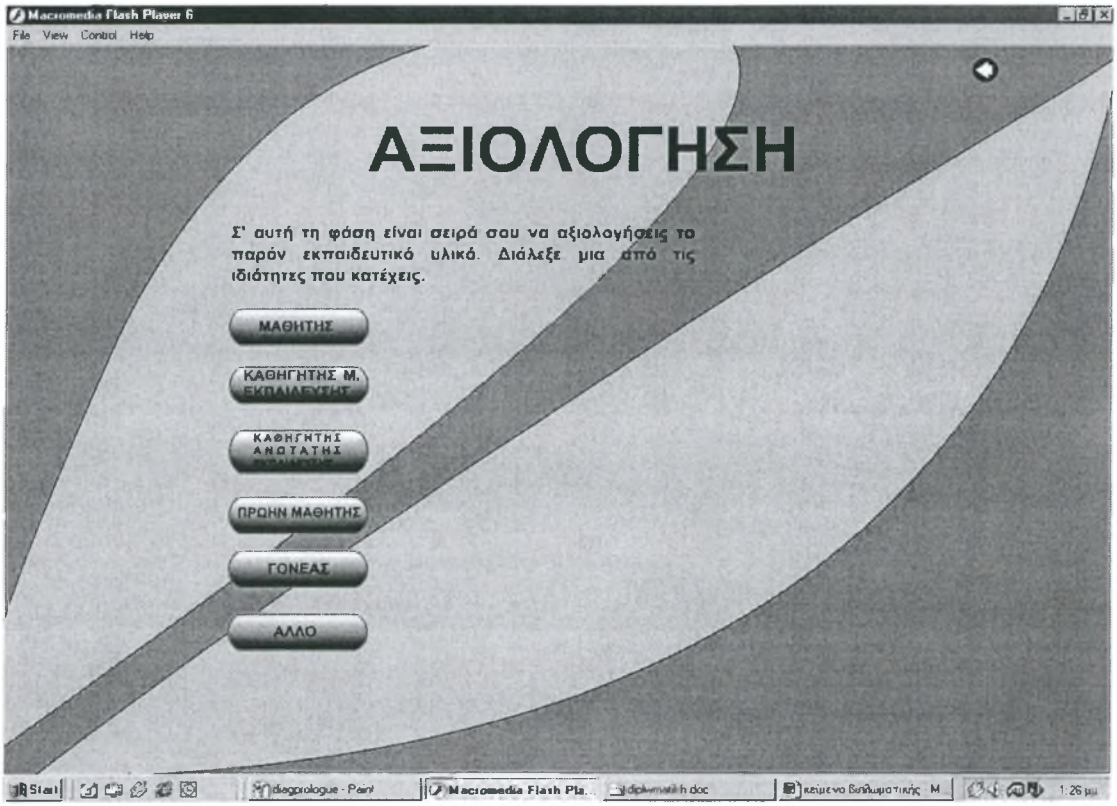
όπως παρατηρούμε ορίζεται η μεταβλητή bir, η οποία λαμβάνει έναν τυχαίο αριθμό μεταξύ 0 και 3. Αποφορτώνεται το προηγούμενο swf με το προηγούμενο διαγώνισμα, και ανάλογα με την τιμή που θα πάρει η bir, ανοίγει το swf με το αντίστοιχο διαγώνισμα. Σ' αυτή τη φάση δεν θεωρήθηκε απαραίτητο να ληφθούν μέτρα, έτσι ώστε αν ο χρήστης φύγει και ξαναεπιστρέψει, να μην πέσει πάλι στο ίδιο διαγώνισμα, γιατί αν γίνει κάτι τέτοιο, τότε ο χρήστης για να επιστρέψει σε διαγώνισμα δευτέρου επιπέδου, θα πρέπει πρώτα να λύσει ένα διαγώνισμα πρώτου επιπέδου, έτσι

καθίσταται ελάχιστα πιθανό, ο χρήστης, και σε περίπτωση που πέσει σε ίδιο διαγώνισμα δευτέρου επιπέδου, να θυμάται απόλυτα τις ερωτήσεις.

Στα τεστ που υπάρχουν στο τέλος του κάθε κεφαλαίου, πρόκειται για ένα τεστ ανά κεφάλαιο, οπότε δεν τίθεται θέμα τυχαίας επιλογής από έναν αριθμό τεστ. Ούτως ή άλλως, σκοπός των τεστ στο τέλος κάθε κεφαλαίου είναι περισσότερο η εξάσκηση του χρήστη, και λιγότερο η αξιολόγησή του. Κατά τα άλλα, τα τεστ στο τέλος κάθε κεφαλαίου, δομούνται όπως και τα διαγωνίσματα.

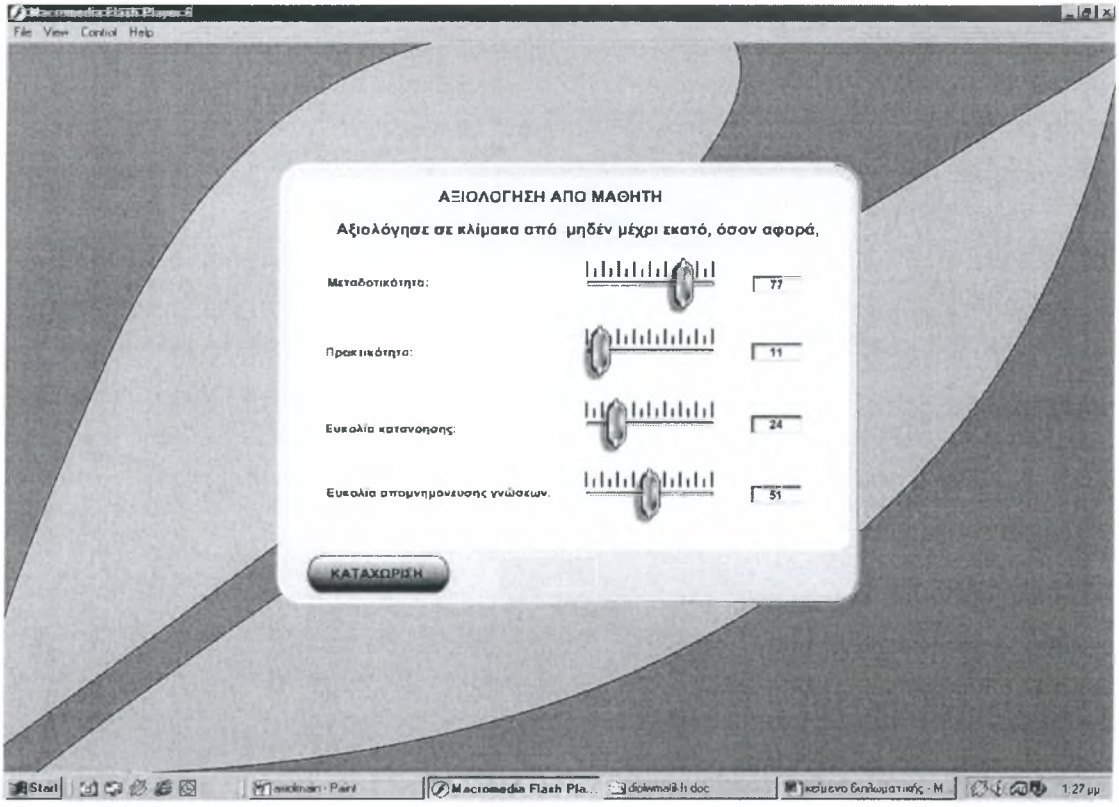
ΑΞΙΟΛΟΓΗΣΗ

Η εργασία περιλαμβάνει και έναν τομέα αξιολόγησης, όπου ο χρήστης μπορεί να αξιολογεί την εργασία. Στο πρώτο καρέ επιλέγει ο χρήστης την ιδιότητά του.



Υπάρχουν έξι ιδιότητες, του μαθητή, του τέως μαθητή, του καθηγητή μέσης εκπαίδευσης, του καθηγητή ανώτατης εκπαίδευσης, του γονέα, και μια ιδιότητα ονόματι «άλλο» σε περίπτωση που ο χρήστης έχει μια ιδιότητα που δεν έχει προβλεφθεί.

Μόλις ο χρήστης διαλέξει την ιδιότητα με την οποία θα αξιολογήσει, εισέρχεται σε ένα περιβάλλον με τέσσερις μπάρες αξιολόγησης, που περιλαμβάνει η καθεμιά κι από ένα κυλιόμενο κουμπί.



Η κάθε μπάρα αξιολογεί την εργασία και σε έναν τομέα. Συνολικά η εργασία αξιολογείται στους τομείς μεταδοτικότα, πρακτικότητα, ευκολία κατανόησης, και ευκολία απομνημόνευσης γνώσεων.

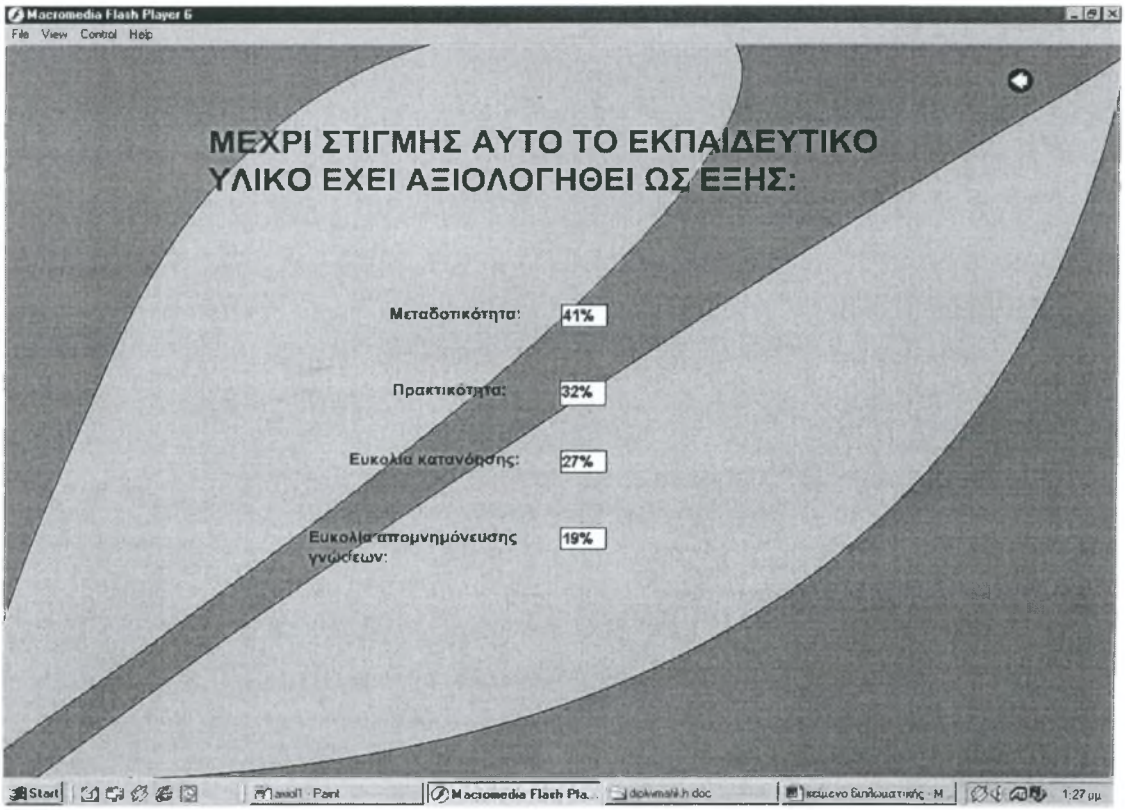
```
int (top);
top = vol._y;
int(left);
left = vol._x;
int(right);
right = vol._x;
int(bottom);
bottom = vol._y+100;
level = 0;

vol.onPress = function()
{
	startDrag("vol", false, left, top, right, bottom);
	dragging = true;
}
vol.onRelease = function()
{
	stopDrag();
	dragging = false;
}
vol.onReleaseOutside = function()
{
	dragging = false;
}
```

```
this.onEnterFrame = function()
{
        if (dragging)
        {
                level = 100-(vol._y-top);
        }
        else
        {
                if (level>100)
                {
                        level = 100;
                }
                else if (level<0)
                {
                        level = 0;
                }
                else
                {
                        vol._y = -level+100+top;
                }
        }
}
```

πρωτίστως ορίζεται η απόσταση στην οποία θα μπορεί να κινηθεί το κουμπί. Αυτή είναι 100 pixels προς τα πάνω και δεν μπορεί να κινηθεί δεξιά ή αριστερά. vol είναι το όνομα του κουμπιού, και _x και _y οι συντεταγμένες του. Το κυλιόμενο κουμπί έχει όμοιο κώδικα με τα προηγούμενο κυλιόμενο κουμπί, που είχαμε δει. Σ΄αυτή την περίπτωση όμως, εκτός από το κουμπί βρίσκεται και ένα δυναμικό πεδίο κειμένου που λαμβάνει τις τιμές του κυλιόμενου κουμπιού, με διαφορετικό τρόπο από το προηγούμενο κυλιόμενο κουμπί. Κάθε φορά που περνάει και ένα καρέ του movie clip που περιέχει το κουμπί, ορίζεται μια συνάρτηση this.onEnterFrame = function(), η οποία εκτελεί μια σειρά ενεργειών όταν το κουμπί κυλιέται. Στο πεδίο κειμένου έχει οριστεί μια μεταβλητή ονόματι level, για να δέχεται τις τιμές που θα παίρνει το κυλιόμενο κουμπί. Έτσι ορίζεται μέσα στη συνάρτηση, η μεταβλητή level να παίρνει τιμές 100 πλην τον αριθμό των pixels που απέχει η θέση του κουμπιού από την κορυφή. Αν όμως το κουμπί δεν κυλιέται, και η μεταβλητή level έχει τιμή πάνω από εκατό, τότε να προσδίδεται στην level η τιμή 100. Επίσης αν η level έχει τιμή κάτω από μηδέν, να προσδίδεται σ' αυτή η τιμή μηδέν. Τέλος αν το κουμπί δεν κυλιέται, τότε η μεταβλητή level να παίρνει τιμή 100 πλην την απόσταση του κουμπιού από την κορυφή. Στον χρήστη αυτό το κυλιόμενο κουμπί φαίνεται να κινείται δεξιά και αριστερά. Δεν υπάρχει όμως θέμα ανυσηχίας, γιατί απλώς το movie clip που εμπεριέχει το κουμπί έχει τοποθετηθεί πλαγίως κι αυτό είναι όλο. Κατά τ' άλλα, οι εντολές ισχύουν θεωρώντας ότι το clip είναι όρθιο. Αυτό γιατί οι συντεταγμένες που ορίστηκαν να αποδίδονται στο πεδίο κειμένου, και αυτές που ορίστηκαν να μετακινείται το κουμπί, είναι οι συντεταγμένες του συγκεκριμένου movie clip, και όχι της κύριας ταινίας. Οπότε, αφού όλο το movie clip είναι τοποθετημένο πλαγίως, οι συντεταγμένες του συνεχίζουν να ισχύουν ως είχαν.

Τα αποτελέσματα της αξιολόγησης πρέπει να κρατούνται και κάθε φορά που κάποιος μπαίνει και την αξιολογεί, να κρατούνται και να προστίθενται με τις παλιές αξιολογήσεις, έτσι ώστε να προκύπτει συνολική αξιολόγηση από όλους όσους έχουν

αξιολογήσει μέχρι στιγμής. Στο πρώτο καρέ της αξιολόγησης ορίζονται τέσσερις μεταβλητές, όπου η καθεμιά αντιστοιχεί και σε κάθε τομέα αξιολόγησης. Επίσης ορίζεται και μια μεταβλητή που θα αποτελεί τον αριθμό των χρηστών που έχουν αξιολογήσει την εργασία, έτσι ώστε στο τέλος ο βαθμός της αξιολόγησης να διαιρείται με τον αριθμό των χρηστών και να προκύπτει το ποσοστό επί τοις εκατό. Επειδή η αξιολόγηση σε κάθε πεδίο είναι σε εκατοστιαία κλίμακα, γι' αυτό και ο αριθμός των χρηστών που θα αποδίδεται στην μεταβλητή tot, θα είναι πολλαπλασιασμένος με το εκατό. Αυτά φαίνονται στην επόμενη εικόνα.



Παρακάτω παρατίθεται κι ο αντίστοιχος κώδικας:

```
int(metadot);
int(prakt);
int(katan);
int(mem);
int(tot);
```

Έπειτα έχει τοποθετηθεί ένα κουμπί με τον κώδικα:

```
on (release)
{
        metadot=f1.level+metadot;
        prakt=f2.level+prakt;
        katan=f3.level+katan;
        mem=f4.level+mem;
        tot=tot+100;
        gotoAndPlay(11);
```

```
}
```

αυτό το κουμπί μόλις πιεστεί, ορίζει στις μεταβλητές που έχουν δηλωθεί πιο πάνω, να προστεθούν οι τιμές που έχει κάθε πεδίο κειμένου σε κάθε clip αξιολόγησης, καθώς επίσης στη μεταβλητή tot να προστεθεί το 100 που αντιπροσωπεύει έναν χρήστη. Έπειτα δίνεται η εντολή να προχωρήσει η ταινία στο καρέ όπου φαίνονται τα τελικά αποτελέσματα της αξιολόγησης. Ο ίδιος κώδικας ενυπάρχει και στα άλλα περιβάλλοντα αξιολόγησης που αφορούν χρήστες άλλης ιδιότητας.

Στο τελευταίο καρέ της αξιολόγησης διακρίνουμε τον κώδικα:

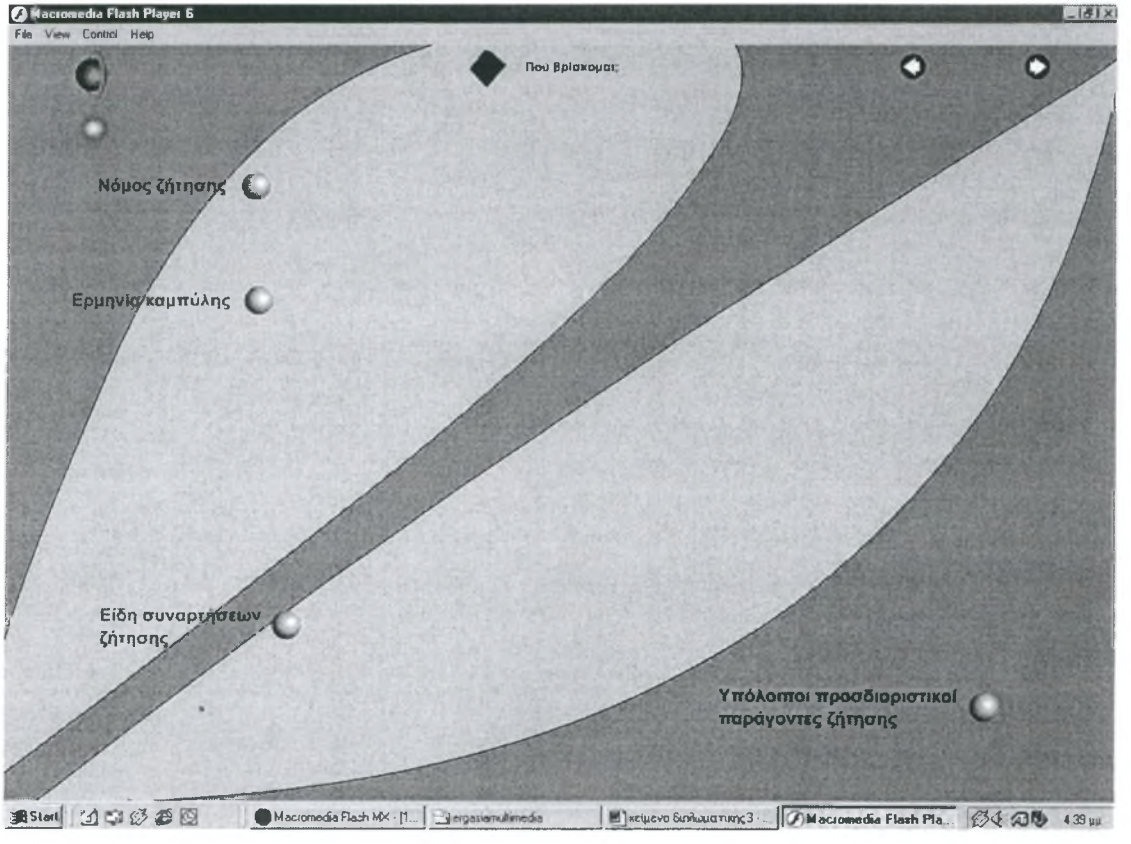
```
stop();
assessmet=SharedObject.getLocal("assessmet");
assessmet.data.arith=assessmet.data.arith+metadot;
assessmet2=SharedObject.getLocal("assessmet2");
assessmet2.data.arith2=assessmet2.data.arith2+tot;
meta.text=int(100*(assessmet.data.arith)/(assessmet2.data.arith2))+"%";

assessprakt=SharedObject.getLocal("assessprakt");
assessprakt.data.arithpr=assessprakt.data.arithpr+prakt;
prak.text=int(100*(assessprakt.data.arithpr)/(assessmet2.data.arith2))+"%";

assesskatan=SharedObject.getLocal("assesskatan");
assesskatan.data.arithka=assesskatan.data.arithka+katan;
kata.text=int(100*(assesskatan.data.arithka)/(assessmet2.data.arith2))+"%";

assessmem=SharedObject.getLocal("assessmem");
assessmem.data.arithme=assessmem.data.arithme+mem;
memo.text=int(100*(assessmem.data.arithme)/(assessmet2.data.arith2))+"%";
```

σ' αυτό το σημείο γίνεται η καταχώρηση των τιμών της αξιολόγησης και εμφανίζεται το τελικό αποτέλεσμα. Οι καταχωρίσεις αυτές δεν γίνεται να λαμβάνουν χώρα στην εικονική μνήμη του flash, γιατί κάθε φορά που θα τερματιζόταν η εφαρμογή, οι αξιολογήσεις θα μηδενιζόταν. Γι' αυτό το λόγο θεωρήθηκε απαραίτητο, τα αποτελέσματα της αξιολόγησης να αποθηκεύονται στο σκληρό δίσκο σε ένα αρχείο που δημιουργεί αυτομάτως κι από μόνο του το flash player, μέσα σε έναν φάκελο του ίδιου του flsh player. Έτσι κάθε φορά που θα χρειάζεται να εμφανιστεί η συνολική αξιολόγηση, θα λαμβάνονται τα στοιχεία μέσα από αυτό το αρχείο και θα διαβάζονται. Ο κώδικας που χρησιμοποιείται είναι ο υποφαινόμενος. Στην πρώτη γραμμή μετά την εντολή stop(); , ορίζεται ένα αντικείμενο κοινής χρήσης με το όνομα «assessmet», όπου και θα αποθηκεύονται κάθε φορά οι τιμές της αξιολόγησης στον τομέα «μεταδοτικότητα». Η μέθοδος getLocal() δημιουργεί ένα κοινόχρηστο αντικείμενο. Αν αυτό ήδη υπάρχει, τότε ανακαλεί τα δεδομένα που υπάρχουν μέσα σ' αυτό. Καθώς παρατηρούμε, το όνομα του κοινόχρηστου αντικειμένου είναι αυτό μέσα στην παρένθεση ("assessmet"). Στην actionscript όμως δεν μπορούμε να αναφέρουμε το κοινόχρηστο αντικείμενο χρησιμοποιώντας το όνομα μέσα στην παρένθεση, γι' αυτό ορίζουμε ένα όνομα με το οποίο θα αναφερόμαστε σ' αυτό το κοινόχρηστο αντικείμενο. Εδώ εμείς, ορίσαμε το ίδιο όνομα. Παρ' όλο το γεγονός ότι το όνομα είναι το ίδιο, έπρεπε να δηλωθεί για να μπορούμε να αναφερόμαστε σ' αυτό το κοινόχρηστο αντικείμενο. Με τη χρήση της δεύτερης γραμμής, και ειδικότερα με την χρήση της ιδιότητας data, γίνεται η αποθήκευση των δεδομένων στο κοινόχρηστο

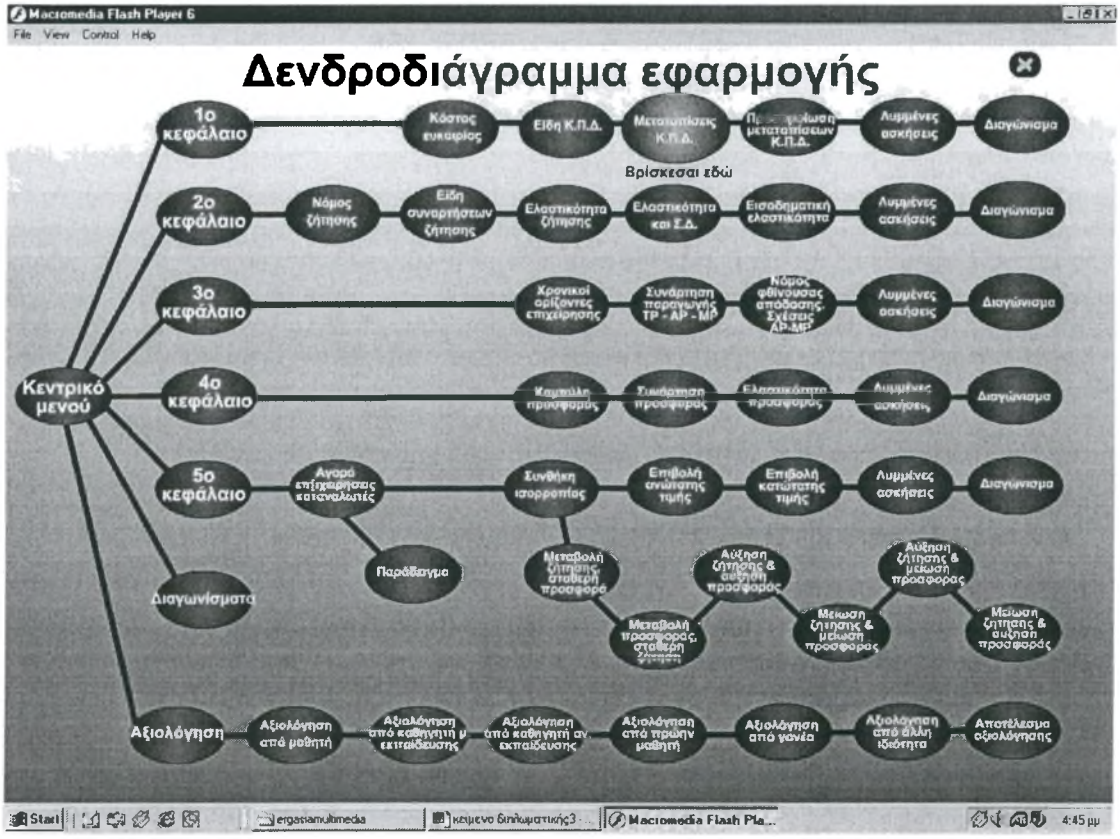
αντικείμενο. Εδώ ορίζεται, στα υπάρχοντα δεδομένα του κοινόχρηστου αντικειμένου, να προστεθεί και η τιμή της μεταβλητής metadot, που έχει οριστεί από την αρχή της αξιολόγησης. Στην πέμπτη γραμμή δίνεται η εντολή μετά από τα προηγούμενα, να οριστεί σαν κείμενο στο δυναμικό πεδίο κειμένου «meta», το πηλίκο της τιμής της συνολικής αξιολόγησης, όσον αφορά την «μεταδοτικότητα», και του συνολικού αριθμού των χρηστών που έλαβαν μέρος στην αξιολόγηση. Η ίδια λογική ακολουθείται και για την δημιουργία και των υπολοίπων κοινόχρηστων αντικειμένων, όπου θα προστίθενται οι τιμές των υπολοίπων μεταβλητών της αξιολόγησης που θα αφορούν τα αντίστοιχα πεδία αξιολόγησης. Δεν προτιμήθηκε οι τιμές αξιολόγησης να μεταφέρονται μέσω δικτύου σε κάποια άλλη τοποθεσία, το οποίο θα ήταν εύκολο, γιατί καθόσον η εφαρμογή απευθύνεται σε μαθητές τρίτης λυκείου, είναι γεγονός σήμερα ότι δεν έχουν όλοι οι μαθητές σύνδεση με το διαδίκτυο. Επομένως μια τέτοια περίπτωση θα αφαιρούσε από τους μη έχοντες σύνδεση στο διαδύκτυο, το δικαίωμα της αξιολόγησης. Όσον αφορά τη σύνδεση της εφαρμογής με μία βάση δεδομένων, προκειμένου να εναποτίθενται τα δεδομένα σ' αυτή, δεν θα υπήρχε καμία διαφορά με τον τρόπο που ακολουθήθηκε, όσον αφορά το αποτέλεσμα, γιατί και οι δύο τρόποι αποθηκεύουν τα δεδομένα στον σκληρό δίσκο. Αντιθέτως κάτι τέτοιο θα αποδεικνυόταν πιο χρονοβόρο, γιατί θα απαιτούσε από τον χρήστη το στήσιμο κάποιου server, προκειμένου για την επικοινωνία της εφαρμογής με τη βάση δεδομένων, ενώ τα αποτελέσματα θα ήταν τα ίδια.

ΤΟ ΚΟΥΜΠΙ «ΠΟΥ ΒΡΙΣΚΟΜΑΙ;»

Στην εφαρμογή, σε κάθε καρέ εχει τοποθετηθεί ένα κουμπί ώστε άν πιεστεί να δείχνει στο χρήστη όλο το δενδροδιάγραμμα της εργασίας, και να μπορεί να προσανατολίζεται, και να γνωρίζει σε σημείο της εφαρμογής βρίσκεται. Το κουμπί αυτό είναι ένα κόκκινο τετραγωνάκι που περιστρέφεται στο πάνω μέρος της οθόνης.



Μόλις πιεστεί αυτό το κουμπί, ο χρήστης εισάγεται σε ένα άλλο περιβάλλον όπου βλέπει που ακριβώς βρίσκεται.

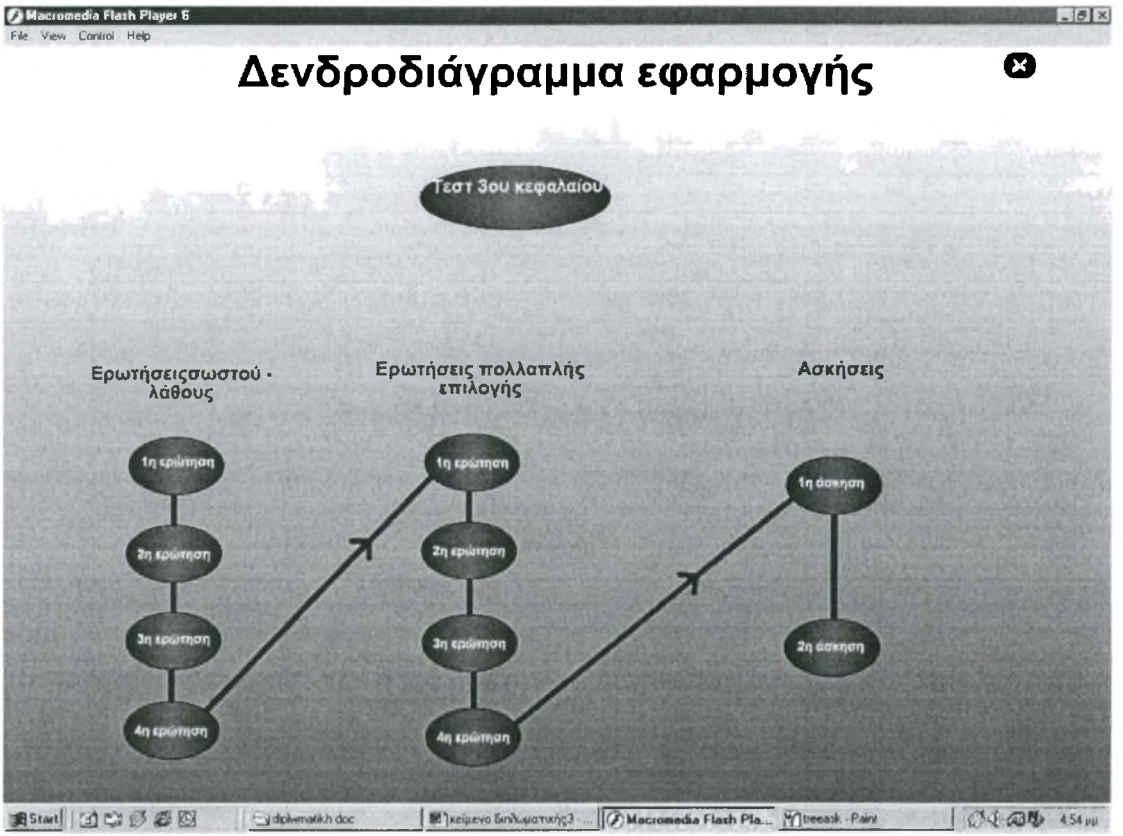


Παρατηρούμε ότι κάθε κυκλάκι φέρει τον τίτλο κάθε θεματικής ενότητας. Η συγκεκριμένη θεματική ενότητα από την οποία έχει έλθει ο χρήστης φαίνεται από το διαφορετικό μέγεθος και χρώμα που διαθέτει. Αν ο χρήστης πατήσει το κουμπί της εξόδου, θα ξαναβρεθεί στο σημείο της εφαρμογής που βρισκόταν.

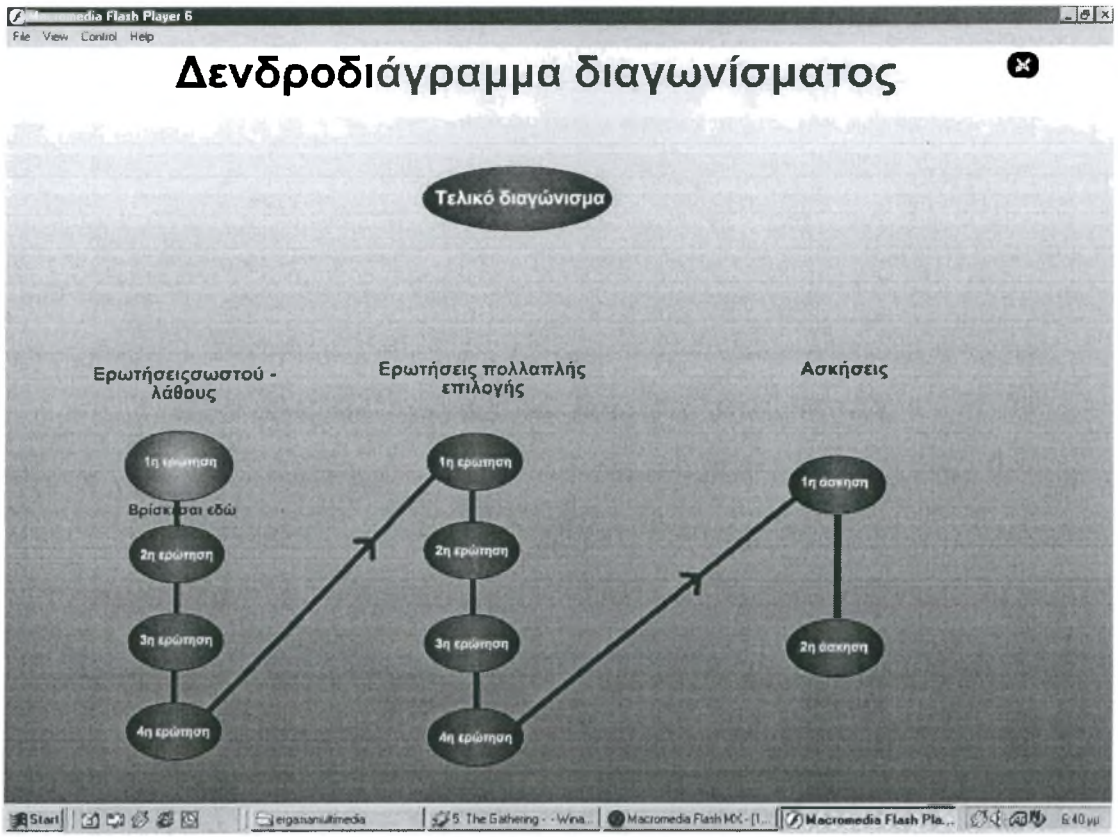
Σ' αυτό το περιβάλλον που βρίσκεται ο χρήστης, έχει τη δυνατότητα να πιέσει ένα από τα κουμπιά με τον τίτλο «Λυμμένες ασκήσεις» ή με τον τίτλο «Διαγώνισμα» που υπάρχουν σε κάθε κεφάλαιο. Έτσι ο χρήστης θα έχει τη δυνατότητα να δεί και τα δενδροδιαγράμματα των λυμμένων ασκήσεων και των διαγωνισμάτων. Και στα δύο αυτά ο χρήστης μπορεί να μπεί και όταν βρίσκεται σε ένα καρέ λυμμένης άσκησης ή σε ένα καρέ διαγωνίσματος. Το αντίστοιχο δενδροδιάγραμμα για τις λυμμένες ασκήσεις είναι το εξής:

Παρατηρούμε ότι αναγράφεται σαν τίτλος «Ασκήσεις 2^{ου} κεφαλαίου» γιατί πιέσαμε το κουμπί των λυμμένων ασκήσεων του δευτέρου κεφαλαίου. Οπως ειπώθηκε και πιο πριν, ο χρήστης έχι τη δυνατότητα να μπεί στο ίδιο περιβάλλον απ'ευθείας από την εφαρμογή. Σ' αυτή την περίπτωση θα δείχνεται και σε ποιά άσκηση απ' όλες βρίσκεται ο χρήστης. Αυτό φαίνεται παρακάτω.

Η ίδια λογική επικρατεί και για το δενδροδιάγραμμα των διαγωνισμάτων. Αν ο χρήστης εισηχθεί από το κύριο δενδριδιάγραμμα, στο δενδροδιάγραμμα των διαγωνισμάτων, τότε θα αναφανεί ο τίτλος που θα δείχνει από ποιό κεφάλαιο εισήλθε στο δενδροδιάγραμμα των διαγωνισμάτων.



Επίσης όπως και στην περίπτωση των λυμμένων ασκήσεων, ο χρήστης έχει τη δυνατότητα να επισκεφθεί το ίδιο περιβάλλον, μέσα από το ίδιο το διαγώνισμα. Σ' αυτή την περίπτωση, το δενδροδιάσγραμμα του διαγωνίσματος του δείχνει σε ποιά ερώτηση ή άσκηση βρίσκεται.



Ένα δείγμα του κώδικα που χρησιμοποιείται για την υλοποίηση και έλεγχο του δενδροδιαγράμματος είναι ο παρακάτω:

```
stop();
tb44.stop();
if(kotara==1||kotara==8||kotara==15||kotara==22||kotara==29)
{
tb44.gotoAndPlay(8);
}
if(abganx==1)
{
title2.text="Λυμμένες ασκήσεις 1ου Κεφαλαίου";
}
```

Κατ'αρχήν δηλώνεται τα κόκκινα movie clips που δείχνουν τη θέση που βρίσκεται ο χρήστης μέσα στην εφαρμογή, να είναι σταματημένα. Στην περίπτωση που η μεταβλητή «kotara» λαμβάνει κάποιες τιμές, τότε παίζει το αντίστοιχο movie clip. Οι τιμές που λαμβάνει η μεταβλητή «kotara», δίνονται όταν πιεστεί το κόκκινο περιστρεφόμενο κυβάκι της κυρίως εφαρμογής. Έτσι ανάλογα με το που βρίσκεται ο χρήστης, παίζει και το αντίστοιχο κλιπ για να δείξει τη θέση του. Επίσης γίνεται κι ένας έλεγχος για να μπεί ο κατάλληλος τίτλος. Αν η μεταβλητή «abganx» λαμβάνει κάποιες τιμές τότε μπαίνει και ο ανάλογος τίτλος. Η μεταβλητή «abganx» λαμβάνει τιμές όταν πιέζεται το κόκκινο κυβάκι, έτσι ώστε να τοποθετηθεί ο κατάλληλος τίτλος.

Τη στιγμή που είναι να βγει ο χρήστης από το δενδροδιάγραμμα, πιέζοντας το κουμπί του κλεισίματος, γίνεται έλεγχος, τί τιμή έχει η μεταβλητή «kotara» προκειμένου να επιστρέψει ο χρήστης εκεί απ' όπου ήλθε.

```
on (press) {
        if(kotara==1)
        {
                gotoAndPlay("1o kefalaio",14);
                }
```

Αυτό είναι ένα αντιπροσωπευτικό κομμάτι του κώδικα με τον οποίο γίνεται ο έλεγχος.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. «Εισαγωγή στο Flash MX για Windows & Macintosh» , Συγγραφέας: Katherine Ulrich Εκδόσεις Κλειδάριθμος Αθήνα 2002

2. «macromedia Flash Mx Actionscript ο εγκεκριμένος εκπαιδευτικός οδηγός» Συγγραφείς: Derek Franklin & Jobe Makar. Απόδοση: Μαίρη Γκλαβά. Εκδόσεις Γκιούρδα, Αθήνα 2002

3. «Εγχειρίδιο του Flash 5» Συγγραφέας: Philip Kerman Απόδοση: Ε. Γκαγκάτσιου. Εκδόσεις Γκιούρδας Αθήνα 2001.

4. «Flash MX Bible» Συγγραφείς: Robert Reinhardt & Snow Dowd Εκδόσεις: Wiley Publishing New York 2002

5. «Actionscript for Flash mx the definite guide» 2nd edition . Συγγραφέας: Colin Moock Εκδόσεις: O'Reilly & Associates; 2nd edition (December 19, 2002)

6. «Wizard test maker» Eduware Inc.

7. «www.macromedia.com/flash/ts/documents»

8. «www.macromedia.com/flash/support»

9. «http://www.cccoe.net/solutions/»

10. «http://www.thesnorkel.org/toolkit/software/ReadMePrincAid.pdf»

11. http://62.212.64.131/tutorials/triangle/triangle.html

12. http://www.cbtcafe.com/flash/