



ΕΠΙΜΟΡΦΩΣΗ ΤΩΝ ΕΚΠΑΙΔΕΥΤΙΚΩΝ
ΓΙΑ ΤΗΝ ΑΞΙΟΠΟΙΗΣΗ ΚΑΙ ΕΦΑΡΜΟΓΗ ΤΩΝ
ΤΠΕ ΣΤΗ ΔΙΔΑΚΤΙΚΗ ΠΡΑΞΗ

ΠΑΚΕ Αν.Μακ.-Θράκης 121/4873/1
Μάιος –Δεκέμβριος 2011



Εργασία 4η

του Τριανταφύλλου Χρήστου, ΑΜ: 178267

Υπεύθυνος Διδάσκων Δρ. Ηλίας Σαράφης

*Τίτλος Εργασίας: " Δημιουργία παιχνιδιού με λογισμικό Γεγονοστραφούς
Προγραμματισμού MsKodu "*

Καβάλα, 16 Φεβρουαρίου 2012

ΕΚΠΑΙΔΕΥΤΙΚΟ ΣΕΝΑΡΙΟ

A' ΜΕΡΟΣ. ΣΧΕΔΙΑΣΗ

1. Τίτλος σεναρίου

"Δημιουργία παιχνιδιού με λογισμικό Γεγονοστραφούς Προγραμματισμού MsKodu"

2. Δημιουργός του σεναρίου

Χρήστος Τριανταφύλλου, Καθηγητής Πληροφορικής ΠΤΕ19

3. Συνοπτική περιγραφή του σεναρίου

" Δημιουργία παιχνιδιού με λογισμικό Γεγονοστραφούς Προγραμματισμού MsKodu "
Διδακτικό Σενάριο για τις ανάγκες της παρούσας επιμόρφωσης Β' επιπέδου, Ειδικό Μέρος.

Το εκπαιδευτικό παιχνίδι, με το συγκεκριμένο διαδικτυακό λογισμικό, δίνει τη δυνατότητα στο μαθητή να δημιουργήσει δικά του παιχνίδια με τη βοήθεια του μικρού ήρωα Kodu, τον Μηχανάκια, το Υποβρύχιο, το Χελιδονόψαρο, το Κανόνι, την Χελώνα, κ.ά. Χαρακτήρες. Έχει τη δυνατότητα να προγραμματίσει την συμπεριφορά αυτών των χαρακτήρων επιλέγοντας να κάνουν διάφορες ενέργειες όπως, να πυροβολούν, να τρώνε μήλα, να ακούν ήχους, να κουβαλούν διάφορα αντικείμενα και κινήσεις σε νέες θέσεις.

Το MSKodu δεν είναι το μόνο περιβάλλον με το οποίο μπορούμε να μάθουμε προγραμματισμό με διασκεδαστικό τρόπο. Υπάρχουν πολλά άλλα παρόμοια προγραμματιστικά περιβάλλοντα που, όπως το Scratch, που μας επιτρέπουν να κατασκευάσουμε τα δικά μας παιχνίδια, να κατασκευάζουμε ρομπότ, να παρουσιάσουμε ιστορίες και να δημιουργούμε ζωγραφίες, εύκολα και γρήγορα ενώ θα κάνουμε παράλληλα μια εισαγωγή στις βασικές έννοιες του προγραμματισμού.

Ο ίδιος ο μαθητής πρέπει να σκεφτεί και να βάλει σε μια λογική σειρά εντολές, αλλά και να σκεφτεί δράσεις-γεγονότα ούτως ώστε να πετύχει το στόχο του. Με αυτό το τρόπο θα μπορέσει να κατανοήσει καλύτερα την οργάνωση, τη δομή και τη σειρά που πρέπει να ακολουθεί για τη δημιουργία ενός προγράμματος. Επίσης, παίζοντας το παιχνίδι ο μαθητής έχει τη δυνατότητα να βλέπει όλες τις κινήσεις ή δράσεις (που στην ουσία είναι εντολές) που έχουν καθοριστεί για τα αντικείμενα του εικονικού κόσμου που δημιουργεί, από την αρχή μέχρι το τέλος του παιχνιδιού.

4. Εκτιμώμενη διάρκεια

Θα χρειαστούμε τρεις διδακτικές ώρες, για ολοκλήρωση του σεναρίου από τους μαθητές, σε εργαστήριο Η/Υ.

5. Σκοπός και Διδακτικοί στόχοι, σε σχέση με τις ΤΠΕ

Βασικός σκοπός του εκπαιδευτικού σεναρίου είναι το να:

- κατανοήσουν οι μαθητές έννοιες του γενοστροφούς προγραμματισμού μέσα από παιχνίδι που θα δημιουργήσουν με το λογισμικό kodu.

Διδακτικοί στόχοι είναι:

- να εξοικειωθούν οι μαθητές με τις έννοιες του αλγόριθμου, του λογικού διαγράμματος, το πως δομούνται και πως αλληλεπιδρούν δυναμικά το ένα ως προς το άλλο
- μέσα από το εκπαιδευτικό παιχνίδι, με το συγκεκριμένο λογισμικό, να δοθεί η δυνατότητα στο μαθητή να κατανοήσει τον τρόπο δημιουργίας παρόμοιων πολυπλοκότερων και ολοκληρωμένων παιχνιδιών.
- να καταφέρει ο μαθητής να σκεφτεί και να οργανώσει event για τα αντικείμενα που προσθέτει στον εικονικό κόσμο που δημιουργεί, ακολουθώντας τη δομή επανάληψης «όσο...κάνε».
- να εξοικειωθούν οι μαθητές με τη χρήση προγραμματιστικών εργαλείων και να μάθουν να δημιουργούν προγράμματα, καθώς και να διορθώνουν, να εκτελούν, και να βελτιώνουν αυτά.
- Επίσης, να κατανοήσουν ότι η επανεξέταση και η βελτίωση των παραγόμενων έργων προγραμματισμού είναι στενά συνδεδεμένα με τις ΤΠΕ, οι οποίες επιτρέπουν τις διεργασίες αυτές να γίνονται γρήγορα, εύκολα και πλήρως οπτικοποιημένα.

Επιπρόσθετα, οι μαθητές μέσα από τη διαδικασία της μάθησης να οικειοποιηθούν δεξιότητες:

- επικοινωνίας (ομιλία, επιχειρηματολογία, διάλογος κ.ά.)
- συνεργασίας με άλλους μαθητές, με την ομαδική εργασία
- κριτικής επεξεργασίας πληροφοριών και παραδοχών
- δημιουργικής και κριτικής σκέψης
- προγραμματισμού και με την μαθηματική σκέψη.

6. Εμπλεκόμενες γνωστικές περιοχές-Θεματικές ενότητες

Υπηρετούνται βασικές επιδιώξεις των θεματικών ενοτήτων του μαθήματος των ΤΠΕ για την Γ΄ Τάξη Γυμνασίου:

- Ενότητα 1: Γνωρίζω τον υπολογιστή ως ενιαίο σύστημα - Προγραμματισμός, Κεφάλαιο 2 - «Ο Προγραμματισμός στην Πράξη», και ως συνέχεια αυτού, μπορεί να ανατεθεί ως εργασία στους μαθητές της ίδια τάξης, η ανάπτυξη ενός παιχνιδιού με το λογισμικό MsKodu, στα πλαίσια της Ενότητας 2: Χρήση εργαλείων έκφρασης, επικοινωνίας, ανακάλυψης και δημιουργίας

7. Τάξεις στις οποίες απευθύνεται

Το διδακτικό σενάριο μπορεί να αξιοποιηθεί από μαθητές της Γ΄ τάξης του Γυμνασίου.

Το θέμα που διαπραγματεύεται είναι ένα πρόβλημα προγραμματισμού εύκολο και κατανοητό, δεν επεξεργάζεται όμως μόνο τις γνωστές δομές ακολουθίας, επιλογής και επανάληψης, αλλά και του event programming καθώς και του αντικειμενοστραφούς προγραμματισμού, τα οποία αναλυτικά διδάσκονται σε μεγαλύτερες ηλικίες και τάξεις πλην όμως με το συγκεκριμένο λογισμικό, χωρίς αυτό να σημαίνει ότι δεν μπορεί να διδαχθεί και με άλλο λογισμικό, όπως το scratch, μπορεί επίσης να φανεί πολύ χρήσιμο και στους μαθητές της Γ' Λυκείου ή των ΕΠΑΛ στον προγραμματισμό, προσαρμόζοντας τις δραστηριότητες του σεναρίου στο γνωστικό επίπεδο των μαθητών.

8. Προσπαιτούμενες γνώσεις των μαθητών

Βασική χρήση του λογισμικού MsKodu, διαμόρφωση παρασκηνίου-κόσμου, επιλογή-εισαγωγή χαρακτήρων, και θεωρητικά την επαναληπτική δομή «όσο...κάνε».

9. Συσχετισμός με το Αναλυτικό Πρόγραμμα

Γ΄ Τάξη Γυμνασίου,

- ✓ Ενότητα 1^η, «Γνωρίζω τον υπολογιστή ως ενιαίο σύστημα - Προγραμματισμός», Κεφάλαιο 2 - «Ο Προγραμματισμός στην Πράξη»,
- ✓ Ενότητα 2^η, «Χρήση εργαλείων έκφρασης, επικοινωνίας, ανακάλυψης και δημιουργίας»

Το εκπαιδευτικό σενάριο υποστηρίζει τη μαθησιακή διαδικασία μέσω της χρήσης και της αξιοποίησης των ΤΠΕ στην εκπαιδευτική πράξη. Είναι συμβατό με το σχολικό Αναλυτικό Πρόγραμμα Σπουδών, αφού υπηρετεί τόσο στόχους που σχετίζονται με τις ΤΠΕ όσο και στόχους που σχετίζονται με την καλλιέργεια της δημιουργικής κριτικής σκέψης.

Συντελεί στην ανάπτυξη επικοινωνιακών δεξιοτήτων, οργάνωσης, σύνθεσης και ανάλυσης ώστε να φθάσουμε στη λύση του προβλήματος. Καθώς επίσης, συντελεί στην ανάπτυξη της φαντασίας και της αισθητικής καλλιέργειας.

10. Διδακτικό υλικό και απαιτούμενη υλικοτεχνική υποδομή για το σενάριο

Το διδακτικό σενάριο υλοποιείται, στο εργαστήριο Πληροφορικής, υποδομή που διαθέτει η σχολική μονάδα, με τη συνεργασία του εκπαιδευτικού Πληροφορικής και των μαθητών της τάξης.

Είναι καλό οι μαθητές να χωριστούν σε ομάδες των 2-3 ατόμων με την απαίτηση του

Θράκης

Απαιτούμενο διδακτικό υλικό είναι ένας σε κάθε υπολογιστή ο Πληροφορικός να έχει εγκαταστήσει το λογισμικό MsKodu που θα χρησιμοποιήσουν οι μαθητές, και να ελέγξει τη σωστή λειτουργία του Τοπικού Δικτύου, ώστε να μπορέσουν οι μαθητές να προσπελάσουν τα ηλεκτρονικά φύλλα εργασίας. [Το Kodu απαιτεί οι υπολογιστές να έχουν αρκετή μνήμη και Vista, Seven ή service pack 3 στο Λ.Σ. XP]

11. Οι εναλλακτικές αντιλήψεις (αναπαραστάσεις, ιδέες, κλπ.) των μαθητών

Από προηγούμενη εμπειρία είναι γνωστό ότι οι μαθητές σε ανάλογα σενάρια δεν ακολουθούν πιστά τα βήματα, με αποτέλεσμα να συναντούν δυσκολίες.

Η λύση εν προκειμένω είναι να επανέλθουν σε προηγούμενα βήματα ή να εκτελέσουν το πρόγραμμα που γράφουν βήμα-βήμα ώστε να οδηγηθούν στην ολοκλήρωση της άσκησης, Αν και δεν θεωρείται απαραίτητο η κάθε ομάδα μαθητών να υλοποιήσει επακριβώς το σενάριο, όσον αφορά το είδος και τη ακριβή αντίστοιχη θέση των αντικειμένων που θα εισάγει στον «κόσμο» της.

12. Οργάνωση τάξης

Το μάθημα γίνεται μέσα στο εργαστήριο Πληροφορικής και οι μαθητές εργάζονται ατομικά, αν αρκούν οι υπολογιστές, ή μπορεί να γίνει χρήση ολιγομελών ομάδων των δύο ή το πολύ τριών μαθητών, σε κάθε υπολογιστή.

Η δραστηριότητα μπορεί να ξεκινήσει με τον καθηγητή να εξηγεί, και να μεταφέρει στις οθόνες των υπολογιστών των μαθητών (μέσω προεγκατεστημένου italc) ή τη χρήση βιντεοπροβολέα, βήμα-βήμα την δημιουργία του σκηνικού του παιχνιδιού και των αντικειμένων που θα εισαχθούν, και τον τρόπο λειτουργίας του παιχνιδιού.

Στη συνέχεια οι μαθητές, μπροστά από τον υπολογιστή τους θα μπορούν να συνεχίσουν την δημιουργία του παιχνιδιού, και τέλος να παρουσιάσουν τα αποτελεσμάτων τους.

Θεωρείται, ότι οι μαθητές είναι ήδη εξοικειωμένοι με το βασικό περιβάλλον του λογισμικού που θα χρησιμοποιήσουν, οπότε δεν αναμένονται ιδιαίτερα προβλήματα στην υλοποίηση του σεναρίου.

13. Διδακτικές προσεγγίσεις και στρατηγικές/ θεωρίες μάθησης

Πιθανός «*διδακτικός θόρυβος*» που μπορεί να προκύψει και να ανατρέψει το «*διδακτικού συμβολαίου*» στο συγκεκριμένο σενάριο θα μπορούσε να οφείλεται σε δυσκολία σύνδεσης με το LAN δίκτυο κάποιου ή κάποιων υπολογιστών του εργαστηρίου. Κρίνεται λοιπόν απαραίτητο να υπάρχουν και φωτοαντίγραφα Φ.Ε., όσα και το πλήθος των Η/Υ. Άλλο πιθανό πρόβλημα που πρέπει να προσεχθεί έχει να κάνει με την δυνατότητα των υπολογιστών. Χρειάζεται αρκετή μνήμη, καλή κάρτα γραφικών και σχετικά γρήγορο επεξεργαστή.

Θεωρητική Προσέγγιση:

Το διδακτικό σενάριο στηρίζεται σε αρχές και πρότυπα των θεωριών μάθησης του Κριτικού Εποικοδομισμού. Σε σχέση με τον Προγραμματισμό, επιζητείται μέσω του περιβάλλοντος του λογισμικού η αλληλεπίδραση με τον μαθητή.

Το σενάριο αναδεικνύει την ορθολογική χρήση των ΤΠΕ, και το προσδοκώμενο «διδακτικό κέρδος», βεβαίως είναι η δυνατότητα των μαθητών να διδαχθούν προγραμματισμό μέσα από το εργονομικό και εύκολο περιβάλλον του λογισμικού Kodu.

14. Φύλλα εργασίας

Μεθοδολογική Προσέγγιση

Οι δραστηριότητες αξιοποίησης εφαρμογών και εκπαιδευτικών λογισμικών των ΤΠΕ προτείνεται να πραγματοποιηθούν σε ένα ομαδοσυνεργατικό πλαίσιο διδασκαλίας (συνεργατική διερευνητική μάθηση).

Οι μαθητές μπορούν να χωριστούν ανάλογα με το πλήθος των ηλεκτρονικών υπολογιστών.

Συνοδευτικά Φύλλα Εργασίας

Τα συνοδευτικά φύλλα εργασίας περιλαμβάνουν σε ημιδομημένα σχέδια τις δραστηριότητες που μπορούν να πραγματοποιήσουν οι μαθητές.

Οι μαθητές/τριες, με τη συνεργασία του καθηγητή τους, καλούνται να αυτενεργήσουν και να δημιουργήσουν όχι με επακριβή βήματα, τις δραστηριότητες που προτείνονται προς λύση.

Στη συνέχεια δίνονται τα φύλλα εργασίας με επισημάνσεις και εξηγήσεις για τις ασκήσεις που καλούνται να φέρουν σε πέρας οι μαθητές, μέσω του συγκεκριμένου λογισμικού.

Β' ΜΕΡΟΣ. ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

1ο ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ

ΦΑΣΕΙΣ:

Φ1. Background- Πίστα παιχνιδιού

ΒΗΜΑ 1: Δημιουργία του κόσμου

Φ2. Αντικείμενα

ΒΗΜΑ 2: Δημιουργία νέου αντικειμένου, διαγραφή ή αλλαγή αντικειμένου

Φ3. Scripts- Ενέργειες

ΒΗΜΑ 3: Εισαγωγή συμπεριφορών

Φ5. Επέκταση

ΒΗΜΑ 4: Νέες συμπεριφορές σε αντικείμενα.

Ας ξεκινήσουμε θέτοντας κάποια ερωτήματα.

☞ Ποιος θα είναι ο κόσμος μας;

☞ Ποια θα είναι τα αντικείμενα σε αυτόν και τι συμπεριφορές θα έχει το καθένα;

1. ΣΕΝΑΡΙΟ ΠΑΙΧΝΙΔΙΟΥ

Θα δημιουργήσουμε λοιπόν ένα επίπεδο, φαίνεται στην επόμενη εικόνα 1, που θα είναι ο κόσμος μας, όπως φαίνεται στο παιχνίδι που μπορείτε επίσης να «φορτώσετε» και να παίξετε. Το γαλάζιο είναι η θάλασσα μας και το καφέ η ξηρά μας με κάποια βουνά.

Περίπου στο κέντρο της πίστας τοποθετήσαμε τον πρωταγωνιστή μας, που εδώ θα είναι ένα *Καραβάκι (Ship)*. Με τα βέλη από το αριστερό πληκτρολόγιο (W,A,S,D) ο παίκτης θα μπορεί να κινεί το *Καραβάκι*. Στο κέντρο της πίστας βάλουμε μερικά ιπτάμενα *Ψαράκια* (είναι η επιλογή *fly fish*) μπορείτε να προσθέσετε όσα θέλετε, τα οποία όταν βλέπει το *Καραβάκι* θα μπορούσε με τα πλήκτρα που προαναφέραμε να τα κυνηγάει και αν έχει επαφή μαζί τους (ακουμπήσει-σαν να τα ψαρεύει) να κερδίζει πόντους.

Προσθέσαμε και μία θαλάσσια *Χελώνα (Turtle)*, κάπου στο βάθος μπροστά, την οποία αν κάνει λάθος και ακουμπήσει να χάνει πόντους, μπορείτε πάλι να τοποθετήσετε όσες επιλέγετε.

Επίσης, τοποθετήσαμε μπροστά δεξιά ένα *Αερόστατο (Ballon)* που θα κινείται με τα βέλη (δεξί πληκτρολόγιο) και *Μήλα (Apples)* που αν τα «πιάσει-ακουμπήσει» το *Αερόστατο* με την κίνηση του, θα κερδίζει πόντους, ενώ αν σε αυτή την προσπάθεια «χτυπήσει» σε *Δένδρο*, που επίσης τοποθετήσαμε όπως φαίνεται στον κόσμο μας θα χάνει πόντους.

Έτσι το παιχνίδι θα μπορούσε να παίζεται και με δύο παίκτες κινώντας ο καθένας τον πρωταγωνιστή του (*Καράβι* ή *Αερόστατο*) και νικητής θα είναι αυτός που έχει μεγαλύτερο σκορ σε κάποιο συγκεκριμένο χρόνο που θα ορίσετε εσείς για λήξη του παιχνιδιού.

Το παιχνίδι μπορεί να έχει πολλές επεκτάσεις, για παράδειγμα τι θα γίνεται όταν τελειώσουν όλα τα *Μήλα* (θα ολοκληρώνεται με νίκη του αντίστοιχου παίκτη;) και το ίδιο για τα *Ψάρια* κλπ.

Δοκιμάστε να φορτώσετε και να παίξετε λίγο το παιχνίδι για να εξοικειωθείτε καλύτερα με την ιδέα του προτού αρχίσετε να το δημιουργείται (πατήστε διπλό κλικ στο αρχείο στο συνοδευτικό υλικό και στη συνέχεια πατήστε *Αρχή Παιχνιδιού (Play Game)* από την *Παλέτα Εργαλείων (Tool Palette)*.



Εικόνα 1

2. ΣΥΜΠΕΡΙΦΟΡΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ ΤΟΥ ΚΟΣΜΟΥ ΜΑΣ

Κάθε αντικείμενο (έχουμε αναφέρει στη θεωρία) ότι έχει συγκεκριμένες. Ο κόσμος-πίστα από την άλλη δεν έχει συμπεριφορά. Είναι απλά η εικόνα στην οποία τοποθετούμε τα διάφορα αντικείμενα και στην οποία αυτά θα κινούνται.

Το *Καράβι* ή *Αερόστατο* από την άλλη θέλουμε να έχει συμπεριφορές. Συγκεκριμένα, τρεις συμπεριφορές το κάθε ένα:

Ας πάρουμε το *Καράβι*, για το οποίο θέλουμε:

1η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* αισθανθεί-ακούσει ότι πατήθηκε το πλήκτρο **W** (ένα **ΓΕΓΟΝΟΣ**), **ΤΟΤΕ** να κινείται στην αντίστοιχη κατεύθυνση πάνω στην πίστα (να κάνει μία **ΕΝΕΡΓΕΙΑ**).

2η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* ακουμπήσει ένα *Ψάρι* (**ΓΕΓΟΝΟΣ**), **ΤΟΤΕ** να «τρώει» το Ψάρι (**ΕΝΕΡΓΕΙΑ**).

3η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* ακουμπήσει ένα *Ψάρι* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να κερδίζει πόντους (ΕΝΕΡΓΕΙΑ).
4η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* ακουμπήσει μία *Χελώνα* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να χάνει πόντους (ΕΝΕΡΓΕΙΑ).

Ας πάρουμε τώρα το *Αερόστατο*:

1η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* αισθανθεί-ακούσει ότι πατήθηκε ένα βέλος (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να κινείται στην αντίστοιχη κατεύθυνση πάνω στην πίστα (ΕΝΕΡΓΕΙΑ).

2η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* ακουμπήσει ένα *Μήλο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τρώει το *Μήλο* (ΕΝΕΡΓΕΙΑ).

3η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* ακουμπήσει ένα *Μήλο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να κερδίζει πόντους (ΕΝΕΡΓΕΙΑ).

4η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* ακουμπήσει ένα *Δένδρο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να χάνει πόντους (ΕΝΕΡΓΕΙΑ).

5η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ ο περάσει χρόνος πχ. 60sec (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τερματίζει το παιχνίδι με νίκη για τον παίκτη με τους περισσότερους πόντους (ΕΝΕΡΓΕΙΑ).

Αναφέραμε στις επεκτάσεις ότι μπορεί επίσης να σκεφτούμε:

ΟΤΑΝ με το *Αερόστατο* φάμε όλα τα μήλα (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τερματίζει το παιχνίδι με νίκη για τον αντίστοιχο παίκτη (ΕΝΕΡΓΕΙΑ).

Γενικά, όσα θέλουμε στον κόσμο μας να έχουν συμπεριφορές, θα είναι *αντικείμενα*.

Αφού λοιπόν διακρίναμε ποιος θα είναι ο κόσμος και ποια τα αντικείμενα του παιχνιδιού μας, ήρθε η ώρα να ανοίξουμε το περιβάλλον του MSKodu. Είναι σημαντικό να τονίσουμε ότι περιγράψαμε τις εντολές σε φυσική γλώσσα και όχι στη γλώσσα του MSKodu, τις οποίες δεν είναι δύσκολο να μετασχηματίσουμε.

3. ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΚΟΣΜΟΥ ΤΟΥ ΠΑΙΧΝΙΔΙΟΥ

Το πρώτο πράγμα που θα φτιάξουμε είναι ο κόσμος, και είναι λογικό, αφού χωρίς κόσμο, πού θα τοποθετούσαμε μετά τα διάφορα αντικείμενα, πού θα πατούσαν οι πρωταγωνιστές μας; Πηγαίνετε λοιπόν στο *ΚΕΝΤΡΙΚΟ ΜΕΝΟΥ (MAIN MENU)* και από εκεί επιλέξτε *ΦΟΡΤΩΣΗ ΚΟΣΜΟΥ (LOAD WORLD)*, όπως φαίνεται και στην παρακάτω εικόνα 2:



Εικόνα 2

Στη συνέχεια πατήστε την επιλογή *Όλοι (All)* ώστε να εμφανιστούν όλοι οι διαθέσιμοι κόσμοι που υπάρχουν. Εμείς είπαμε θέλουμε να φτιάξουμε από την αρχή έναν καινούργιο κόσμο, οπότε ψάξτε στους κόσμους αυτούς, χρησιμοποιώντας τα βέλη στο πληκτρολόγιο ή τη ροδέλα του ποντικιού, μέχρι να βρείτε την επιλογή *Νέος Κόσμος (New World)*. Πατήστε *Παίξε (Play)* και περιμένετε να φορτώσει.



Εικόνα 3

Θα πρέπει να σας εμφανίζετε ένας κενός κόσμος (πράσινο γρασίδι) σαν αυτόν της επόμενης εικόνας.



Εικόνα 4

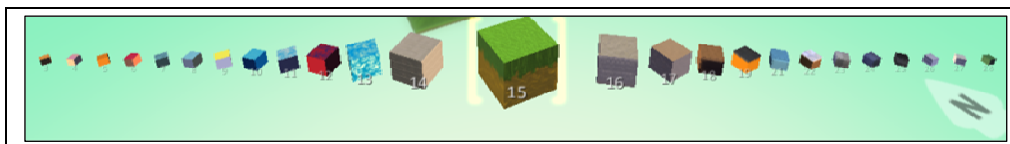
Πλέον είμαστε έτοιμοι να δημιουργήσουμε. Στον κενό αυτό κόσμο πατήστε το πλήκτρο Esc ώστε να εμφανιστεί η **Παλέτα Εργαλείων (Tool Palette)**, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 5

Αρχικά, θέλουμε να αλλάξουμε το γρασίδι με θάλασσα και να προσθέσουμε έδαφος και βουνά. Αυτά γίνονται με τη **Βούρτσα Εδάφους (Ground Brush)**, που φαίνεται στην εικόνα επιλεγμένη και ταυτόχρονα θα δούμε ότι θα εμφανιστούν πάνω αριστερά και δεξιά της δύο άλλα εικονίδια.

	<p>Επιλέξτε το αριστερό (επιλογή αλλαγής του υλικού). Εμφανίζεται μια λίστα με διαφορετικούς τύπους εδάφους και μπορείτε να διαλέξετε τον τύπο εδάφους που θέλετε να προσθέσετε με τη Βούρτσα.</p>
--	--



Εικόνα 6

Τώρα μπορείτε να αρχίσετε να **Προσθέτετε Έδαφος (Add Ground)** στον κόσμο, κάνοντας αριστερό κλικ με το ποντίκι σε διάφορα σημεία του κόσμου. Προσπαθήστε να δημιουργήσετε την πίστα της εικόνας 1.



Εικόνα 7

Αν σας χρειάζεται να μεγαλώσετε λίγο το μέγεθος της *Βούρτσας* μπορείτε να το κάνετε πατώντας το δεξί βέλος του πληκτρολογίου, και μικρύνετε το μέγεθος της *Βούρτσας* πατώντας το αριστερό. Έτσι μπορείτε να προσθέσετε μεγάλα παρόμοια κομμάτια πίστας πιο γρήγορα.

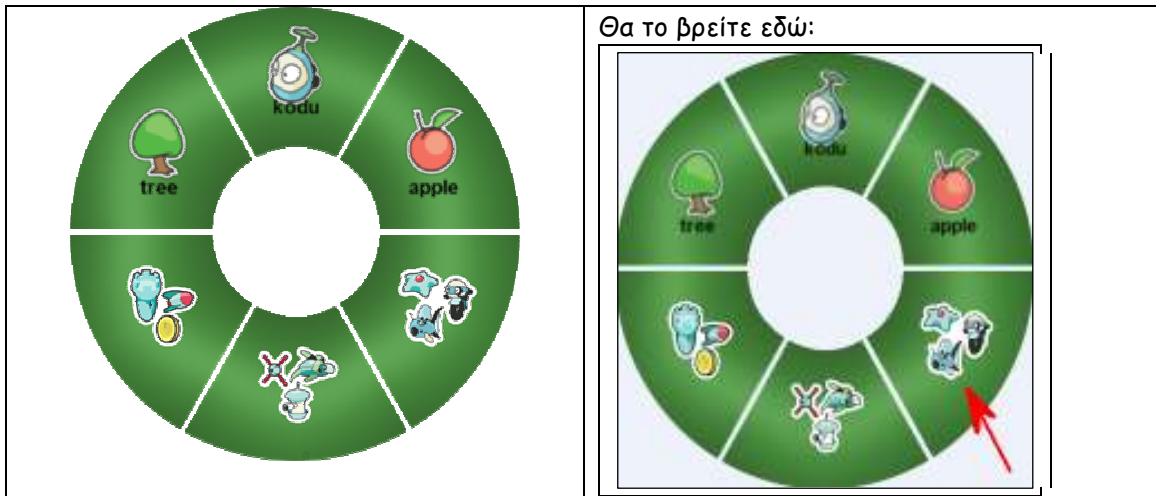
Συνεπώς μπορούμε ακόμη και με ένα μόνο κλικ να δημιουργήσουμε ένα μεγάλο ορθογώνιο που θέλουμε περίπου για την πίστα μας. Αν θέλετε να διαγράψετε κομμάτια πίστας, πατήστε το δεξί πλήκτρο του ποντικιού. Και αφού ο κόσμος μας είναι έτοιμος, μπορούμε να εισάγουμε τα αντικείμενά μας.

4. ΕΙΣΑΓΩΓΉ ΑΝΤΙΚΕΙΜΈΝΩΝ

Μπορείτε να βρείτε όλα τα διαθέσιμα αντικείμενα στο MSKodu, αν επιλέξετε το *Εργαλείο Αντικειμένων (Object Tool)* από την παλέτα εργαλείων. Αφού το επιλέξετε, κάντε αριστερό κλικ με το ποντίκι σας σε κάποιο σημείο της πίστας και θα σας εμφανιστεί ένα μενού, λέγεται «πίτα», με όλα τα αντικείμενα που μπορείτε να εισάγετε.

Η πρώτη πίτα έχει έξι εικόνες, τρία αντικείμενα (ένα δέντρο, τον Kodu, και ένα μήλο) και τρεις κατηγορίες αντικειμένων που εμφανίζουν νέες πίτες και περιέχουν επιπλέον αντικείμενα. Ας ξεκινήσουμε με τον πρωταγωνιστή μας, το *Καραβάκι (Ship)*.





Εικόνα 8

Τοποθετήστε το *Καραβάκι* στη θέση που θέλετε, και φαίνεται στην παραπάνω εικόνα 8 και την επόμενη εικόνα 9.




Μπορείτε να αλλάζετε τη θέση των αντικειμένων κάνοντας απλώς αριστερό κλικ πάνω τους, κρατώντας πατημένο το αριστερό πλήκτρο του ποντικιού και μετακινώντας το ποντίκι στη θέση που θέλετε να μεταφερθεί το εκάστοτε αντικείμενο.

Για να εισάγουμε το επόμενο αντικείμενο, μετακινούμε το ποντίκι ώστε ο δείκτης να βρεθεί στο κέντρο της πίστας και κάνουμε αριστερό κλικ με το ποντίκι, όπως και πριν, ώστε να εμφανιστούν πάλι τα αντικείμενα που μπορούμε να εισάγουμε.

Θα χρειαστούμε σύμφωνα με το σενάριο Καράβι, Ψάρια, Χελώνα, Αερόστατο Δένδρα και μήλα.



<p>Τέλος, θα εισάγουμε τρία-τέσσερα νέα αντικείμενα <i>Μήλα (Apples)</i>, σε διάφορες θέσεις.</p>	
---	--

<p>Σχετικά με την πλοήγηση μέσα στον κόσμο μας και για να μην βλέπουμε τον κόσμο μόνο από μια θέση-οπτική γωνία, μπορούμε να μετακινήσουμε την κάμερα, μέσα από την οποία βλέπουμε στο MSKodu με την επιλογή Μετακίνηση Κάμερας (Move Camera) που βρίσκεται στην <i>Παλέτα Εργαλείων (Tool Palette)</i> όπως φαίνεται από την επόμενη εικόνα.</p>	
--	---



Εικόνα 8

Επειδή όμως πολλές φορές θα θέλαμε να μετακινήσουμε την κάμερα του κόσμου ενώ ταυτόχρονα πραγματοποιείτε μια άλλη εργασία, το Kodu μας ενεργοποιεί όλες τις προηγούμενες επιλογές μετακίνησης με το ποντίκι απλά πατώντας παράλληλα και το πλήκτρο Space. Έτσι, αφού έχουμε ενεργοποιήσει το «χεράκι», μπορούμε με το ποντίκι να μετακινηθούμε σε όποια θέση του κόσμου επιθυμούμε και από όποια οπτική γωνιά μας βολεύει.

Ενέργεια ποντικιού	Κίνηση κάμερας
Μετακίνηση ποντικιού με πατημένο το αριστερό πλήκτρο	Η κάμερα μετακινείται ανάλογα προς τα βόρεια, νότια, ανατολικά και δυτικά της πίστας
Μετακίνηση ποντικιού με πατημένο το δεξί πλήκτρο	Η κάμερα αλλάζει θέση στους τρεις άξονες, δηλαδή αλλάζει ύψος και περιστρέφει τον κόσμο μας
πλήκτρο κύλισης του ποντικιού	Αλλάζει η απόσταση (zoom in ή out) της κάμερας από τον κόσμο

Μένει τώρα στο επόμενο βήμα θα δώσουμε συγκεκριμένες συμπεριφορές στα αντικείμενά μας.

4. ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΤΑ ΑΝΤΙΚΕΙΜΕΝΑ-ΣΥΜΠΕΡΙΦΟΡΕΣ

Μέχρι εδώ μάλλον τα καταφέραμε, όμως αν πατήσουμε την *Αρχή Παιχνιδιού (Play Game)* από την *Παλέτα Εργαλείων (Tool Palette)* για να παίξουμε το παιχνίδι που δημιουργήσαμε, θα δούμε ότι δεν γίνεται τι γίνεται! Το οποίο βέβαια είναι λογικό, αφού δεν δώσαμε ακόμη καμιά συμπεριφορά στα αντικείμενα.

Για να προγραμματίσουμε ένα αντικείμενο πρέπει να πατήσουμε Esc για να εμφανιστεί η *Παλέτα Εργαλείων (Tool Palette)*, να επιλέξουμε το *Εργαλείο Αντικειμένων (Object Tool)* και να κάνουμε δεξί κλικ με το ποντίκι μας πάνω στο αντικείμενο στο οποίο θέλουμε να αποδώσουμε συμπεριφορά.



Αν ξεκινήσουμε από το Καραβάκι και κάνουμε δεξί κλικ πάνω του, και μετά επιλέγουμε **Προγραμμάτισε (Program)**.



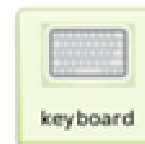
Εδώ θα παρατηρήσουμε ότι η εντολή που μπορούμε να δημιουργήσουμε έχει συγκεκριμένη μορφή, που αποτελείται από δύο τμήματα: **ΟΤΑΝ (WHEN)** συμβεί κάτι... **ΚΑΝΕ (DO)** κάτι..., όπως φαίνεται και παρακάτω.



Η συμπεριφορά όλων των αντικειμένων ακολουθεί την ακόλουθη Αρχή:



Κάθε αντικείμενο στο MSKodu έχει μία σειρά από αισθητήρες. Τα αντικείμενα μπορούν να αντιληφθούν για παράδειγμα ποιο πλήκτρο πατάει ο χρήστης στο πληκτρολόγιο και επομένως εμείς ως προγραμματιστές να προσδιορίσουμε αντίστοιχες ενέργειες για το αντικείμενο. Υπάρχουν αισθητήρες ακοής, αφής, όρασης κλπ. που θα τους μελετήσουμε σε επόμενα μαθήματα.



Για τη συγκεκριμένη συμπεριφορά θέλουμε να χρησιμοποιήσουμε τον αισθητήρα αλληλεπίδρασης με το Πληκτρολόγιο (Keyboard). Αυτό βεβαίως δεν μας είναι αρκετό αφού πρέπει να προσδιορίσουμε ποιο πλήκτρο του πληκτρολογίου θα προκαλεί την επιθυμητή ενέργεια. Κάθε αισθητήρας συνήθως συνοδεύεται από κάποια προσδιοριστικά, από στοιχεία που προσδιορίζουν καλύτερα το γεγονός που θέλουμε να αντιληφθεί το αντικείμενό μας, όπως το όταν δει το αντικείμενο μας με το αισθητήριο του ένα Ψάρι, ή όταν πατήσει (αισθητήρας) το πλήκτρο space (προσδιοριστικό), ή όταν ακούσει (αισθητήρας) μία Μηχανή (προσδιοριστικό).

Αφού τα αντικείμενά μας αντιληφθούν ένα γεγονός, θα πρέπει να εκτελέσουν κάποια ενέργεια. Κάθε αντικείμενο στο MSKodu έχει μια πληθώρα ενεργειών που μπορεί να εκτελέσει όπως να μιλήσει, να μετακινηθεί, παίξει έναν ήχο, να ρίξει έναν πύραυλο, να χρωματίσει κ.ά.

1η ΣΥΜΠΕΡΙΞΟΡΑ:

ΟΤΑΝ το Καράβι αισθανθεί-ακούσει ότι πατήθηκε το πλήκτρο **W** (ένα **ΓΕΓΟΝΟΣ**), **ΤΟΤΕ** να κινείται στην αντίστοιχη κατεύθυνση πάνω στην πίστα (να κάνει μία **ΕΝΕΡΓΕΙΑ**).

Η ενέργεια που θέλουμε να εκτελεστεί με το πάτημα των πλήκτρων του πληκτρολογίου στο παράδειγμά μας, είναι η μετακίνηση του Καραβιού, δηλαδή η Κινήσου (Move), την οποία και επιλέγουμε και δημιουργούμε την παρακάτω εντολή, πατώντας επάνω στο «+» του **ΟΤΑΝ(WHEN)** και επιλέγουμε τον αισθητήρα Πληκτρολόγιο (Keyboard), και μετά



επιλέγουμε το προσδιοριστικό του αισθητήρα, που εδώ θα είναι τα πλήκτρα





Εικόνα

A circular menu with various action blocks. A red arrow points to the 'move' block.	<p>Το ίδιο και με το <i>KANE (DO)</i> και επιλέγουμε την ενέργεια <i>Κινήσου (Move)</i>, και μπορούμε αν θέλουμε να προσθέσουμε <i>γρήγορη (quickly)</i> κίνηση.</p> <p>Έτσι ετοιμάσαμε την πρώτη μας εντολή, που φαίνεται στην παρακάτω εικόνα, και μπορούμε με ένα «Esc» και <i>Αρχή Παιχνιδιού (Play Game)</i> από την <i>Παλέτα Εργαλείων (Tool Palette)</i>, να δοκιμάσουμε αν καταφέρνουμε να κινήσουμε το <i>Καραβάκι</i>.</p>
---	---



Ομοίως συνεχίζουμε:

2η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* ακουμπήσει ένα *Ψάρι* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να «τρώει» το Ψάρι (ΕΝΕΡΓΕΙΑ).
Αφήνετε ως δραστηριότητα στις ομάδες μαθητών.

3η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* ακουμπήσει ένα *Ψάρι* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να κερδίζει 10 πόντους (ΕΝΕΡΓΕΙΑ).



4η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Καράβι* ακουμπήσει μία *Χελώνα* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να χάνει 20 πόντους (ΕΝΕΡΓΕΙΑ).

Τα αντικείμενα στο Kodu έχουν στη διάθεσή τους τον αισθητήρα **Πέσεις Πάνω** (*bump*) μπορούν δηλαδή να αντιληφθούν πότε πέφτουν πάνω σε ένα άλλο αντικείμενο. Ο αισθητήρας απαιτεί ως προσδιοριστικό το όνομα του άλλου αντικειμένου της σύγκρουσης, στη δική μας περίπτωση της *Χελώνας* (*Turtle*).



Για να υλοποιήσετε αυτό το τμήμα της συμπεριφοράς, πατήστε πάνω στο "+" του **ΟΤΑΝ (WHEN)** και επιλέξτε τον αισθητήρα **Πέσεις Πάνω (Bump)** και μετά ξαναπατήστε στο "+" του **ΟΤΑΝ (WHEN)** και επιλέξτε ως προσδιοριστικό του αισθητήρα τη *Χελώνα (Turtle)* που βρίσκεται στην κατηγορία *αντικείμενα (objects)*. Θα πρέπει να βλέπετε στην οθόνη σας κάτι τέτοιο:



Ακριβώς παρόμοια θα σκεφτούμε για να προγραμματίσουμε συμπεριφορές για το **Αερόστατο**:

1η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* αισθανθεί-ακούσει ότι πατήθηκε ένα βέλος (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να κινείται στην αντίστοιχη κατεύθυνση πάνω στην πίστα (ΕΝΕΡΓΕΙΑ).

2η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* ακουμπήσει ένα *Μήλο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τρώει το *Μήλο* (ΕΝΕΡΓΕΙΑ).

Η ενέργεια που θέλουμε να εκτελέσει το *Αερόστατο* όταν συμβεί το παραπάνω γεγονός είναι να φάει το μήλο. Το Kodu μας παρέχει μια ενέργεια **Ψάε (Eat)**, που εξαφανίζει



το αντίστοιχο αντικείμενο. Προσέξτε ότι για να εκτελεστεί επιτυχημένα η ενέργεια δεν αρκεί μόνο το **Φάε (Eat)**, αλλά να προσδιορίσουμε το τι θα φάει, εν προκειμένω θα χρησιμοποιήσουμε το προσδιοριστικό **To (it)** δηλαδή θα πούμε στο Kodu «Φάε Το». Αυτό σημαίνει ότι θα φάει το αντικείμενο με το οποίο έχει έρθει σε επαφή, άρα το προσδιοριστικό ενέργειας **To (it)** αναφέρεται στο αντικείμενο *Μήλο (Apple)*.

Για να καταγράψετε την ενέργεια, πατήστε πάνω στο "+" του **KANE (DO)** και επιλέξτε την ενέργεια **Φάε (Eat)** και μετά ξαναπατάμε στο "+" του **KANE (DO)** και επιλέγουμε το προσδιοριστικό ενέργειας **To (It)**. Θα πρέπει να βλέπετε στην οθόνη σας τις επόμενες εντολές:



3η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* ακουμπήσει ένα *Μήλο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να κερδίζει πόντους (ΕΝΕΡΓΕΙΑ).

4η ΣΥΜΠΕΡΙΦΟΡΑ:

ΟΤΑΝ το *Αερόστατο* ακουμπήσει ένα *Δένδρο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να χάνει πόντους (ΕΝΕΡΓΕΙΑ).

5η ΣΥΜΠΕΡΙΦΟΡΑ-ΤΕΛΟΥΣ:

ΟΤΑΝ ο περάσει χρόνος πχ. 60sec (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τερματίζει το παιχνίδι με νίκη για τον παίκτη με τους περισσότερους πόντους (ΕΝΕΡΓΕΙΑ).

Στο τέλος της προσπάθειάς μας θα πρέπει να έχουμε κάτι ανάλογο με την επόμενη εικόνα.



Τώρα μπορούμε να σώσουμε πατώντας στο «σπίτι» και επιλέγοντας "Save my world", και αν θέλουμε με το *Play Game* τρέχουμε το παιχνίδι να δούμε αν όλα δουλεύουν καλά.



Τελικά, ο κόσμος μας είναι έτοιμος, τα αντικείμενα με τις συμπεριφορές είναι έτοιμα, άρα είμαστε έτοιμοι να δοκιμάσουμε το παιχνίδι μας.

Πατήστε Esc και μετά **Αρχή Παιχνιδιού (Play Game)**.



ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Προσπαθήστε να απαντήσετε στις ακόλουθες δραστηριότητες. Η υλοποίησή τους θα φαίνεται στο παραδοτέο αρχείο:

1. **ΟΤΑΝ** το *Καράβι* ακουμπήσει ένα *Ψάρι* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να «τρώει» το Ψάρι (ΕΝΕΡΓΕΙΑ).

Δύσκολη στη σκέψη δραστηριότητα.

2. **ΟΤΑΝ** το *Αερόστατο* φάει όλα τα *Μήλα* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τερματίζει το παιχνίδι (ΕΝΕΡΓΕΙΑ).

Για βοήθεια σχετικά με το παραπάνω: Θέλουμε, για να το λύσουμε αυτό, το *Αερόστατο* να έχει έναν αισθητήρα «τα έφαγε όλα» ώστε να χτίσουμε την εντολή που σκεφτήκαμε. Δυστυχώς όμως δεν υπάρχει τέτοιος αισθητήρας στο MSKodu. Και είναι λογικό μια γλώσσα προγραμματισμού να έχει ένα συγκεκριμένο αριθμό εντολών και συνεπώς να μην υλοποιεί οποιαδήποτε σκέψη, μας έρχεται στο μυαλό.

Γι' αυτό οι προγραμματιστές μεταφράζουν τις σκέψεις τους βάσει των εντολών που έχουν στη διάθεσή τους. Έτσι, πρέπει να βρούμε έναν τρόπο για να αντιληφθεί το *Αερόστατο* ότι έφαγε όλα τα μήλα. Θυμηθείτε προηγούμενα είδαμε ότι υπάρχει ο αισθητήρας της όρασης. Αν έχουν φαγωθεί όλα τα μήλα, τότε δεν θα υπάρχουν πάνω στην πίστα μας άλλα. Αυτό σημαίνει ότι δεν θα μπορεί να δει το *Αερόστατο* μας κανένα άλλο μήλο. Συνεπώς θα μπορούσαμε να μετασχηματίσουμε την προηγούμενη εντολή σε:

ΟΤΑΝ το *Αερόστατο* δεν βλέπει μήλα (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να τερματίζει το παιχνίδι (ΕΝΕΡΓΕΙΑ).

Οι δυο εντολές είναι ισοδύναμες.

Το *Αερόστατο* αντιλαμβάνεται το γεγονός αυτό, με τον αισθητήρα **Βλέπεις (See)** και το προσδιοριστικό **Μήλο (Apple)**. Το MSKodu δέχεται την άρνηση ως τελευταίο προσδιοριστικό του εκάστοτε αισθητήρα, δηλαδή πρέπει να προσθέσουμε το **Δεν (Not)** στο τέλος του γεγονότος. Σαν φράση ακούγεται λίγο περίεργο: "**Βλέπεις Μήλο Δεν**", αυτή όμως είναι η άρνηση στο *Kodu*.

Θα πρέπει να βλέπετε κάτι τέτοιο:



3. **ΟΤΑΝ** το *Μήλο* ακουμπήσει με το *Αερόστατο* (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να εξαφανίζεται (ΕΝΕΡΓΕΙΑ).

Στην ουσία έχουμε γράψει ξανά τέτοια εντολή στο *Αερόστατο*. Έτσι δεν χρειάζεται να δώσουμε αντίστοιχη εντολή στα *Μήλα*!



Έχουμε δηλ. ήδη εξασφαλίσει ότι κάθε φορά που το *Αερόστατο* ακουμπάει ένα *Μήλο* (ή ένα *Μήλο* ακουμπάει ένα *Αερόστατο*, το ίδιο είναι) θα το τρώει, οπότε αυτό θα εξαφανίζεται. Οπότε ήδη δώσουμε και την επιθυμητή συμπεριφορά στα τρία *Μήλα*.

4. **ΟΤΑΝ** το *Σκορ-Scored* είναι πχ. 50μονάδες (ΓΕΓΟΝΟΣ), **ΤΟΤΕ** να έχω τον νικητή στο παιχνίδι (ΕΝΕΡΓΕΙΑ).



5. Δώστε την εξής συμπεριφορά στον *Kodu*.

ΟΤΑΝ ο *Kodu* βλέπει ένα *μήλο* **ΤΟΤΕ** να κινείται από μόνος του προς τα μήλα και να το τρώει. Δηλαδή, ο *Kodu* να κινείται μόλις πατήσουμε *Αρχή Παιχνιδιού (Play Game)*.

6. Πως νομίζετε (διαισθητικά) ότι λειτουργούν τα προσδιοριστικά της ενέργειας *Κινήσου (Move)* που δεν χρειάστηκε να χρησιμοποιήσετε, στο σενάριο;

ΒΙΒΛΙΟΓΡΑΦΙΑ

- Διεύθυνση Σπουδών Δευτεροβάθμιας Εκπαίδευσης - ΥΠΕΠΘ. (2009). Οδηγίες για το μάθημα Πληροφορικής Γ' Γυμνασίου. Επιστολή προς Γυμνάσια. Ανακτήθηκε από http://www.pi-schools.gr/lessons/computers/gymnasio/od_didask_logo.doc , ανακτήθηκε στις 12 Φεβρουαρίου 2012
- ΥΠΕΠΘ. (2003). Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών (ΔΕΠΠΣ) - Αναλυτικό Πρόγραμμα Σπουδών (ΑΠΣ). Ανακτήθηκε από http://www.pi-schools.gr/lessons/computers/epps/epps_informatics_gim_fek304.pdf , ανακτήθηκε στις 1 Φεβρουαρίου 2012
- Γραβάνη, Μ. (2008). Ο πληροφορικός γραμματισμός στα Σχολεία Δεύτερης Ευκαιρίας: μια μελέτη των απόψεων και των εμπειριών εκπαιδευτών πληροφορικής. Στο Β. Κόμης (επιμ.), Πρακτικά 4ου Πανελληνίου Συνεδρίου "Διδακτική της Πληροφορικής" (σ. 405-414), Πάτρα: Εκδόσεις Νέων Τεχνολογιών.
- ΔΕΠΠΣ (2003). Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών. Αθήνα: Παιδαγωγικό Ινστιτούτο, Ανακτήθηκε στις 19 Νοεμβρίου 2009 από <http://www.pi-schools.gr/programs/depps>
- Ντρενογιάννη, Ε., & Πριμεράκης, Γ. (2008). Ψηφιακά σενάρια διδασκαλίας για το δημοτικό σχολείο: Η διερεύνηση του περιεχομένου τους με έμφαση στα δομικά, μορφολογικά και μεθοδολογικά στοιχεία σχεδιασμού. Θέματα Επιστημών και Τεχνολογίας στην Εκπαίδευση, 1(2), 143-175.
- Τζιμογιάννης, Α., Πολίτης, Π. & Κόμης, Β. (2005). Μελέτη των αναπαραστάσεων τελειόφοιτων μαθητών Ενιαίου Λυκείου για την έννοια της μεταβλητής. Στο Α. Τζιμογιάννης (επιμ.), Πρακτικά 3ου Πανελληνίου Συνεδρίου "Διδακτική της Πληροφορικής" (σ. 61-70), Κόρινθος: Εκδόσεις Νέων Τεχνολογιών.