

Συνέδρια της Ελληνικής Επιστημονικής Ένωσης Τεχνολογιών Πληροφορίας & Επικοινωνιών στην Εκπαίδευση

Τόμ. 1 (2009)

1ο Εκπαιδευτικό Συνέδριο «Ένταξη και χρήση των ΤΠΕ στην εκπαιδευτική διαδικασία»



Αξιοποίηση του Περιβάλλοντος Mortran στη Διδασκαλία Αλγοριθμικών Δομών

Ε. Κανίδης, Ρ. Σταθακοπούλου

Βιβλιογραφική αναφορά:

Κανίδης Ε., & Σταθακοπούλου Ρ. (2024). Αξιοποίηση του Περιβάλλοντος Mortran στη Διδασκαλία Αλγοριθμικών Δομών. *Συνέδρια της Ελληνικής Επιστημονικής Ένωσης Τεχνολογιών Πληροφορίας & Επικοινωνιών στην Εκπαίδευση, 1*, 050–056. ανακτήθηκε από <https://eproceedings.epublishing.ekt.gr/index.php/cetpe/article/view/6405>

Αξιοποίηση του Περιβάλλοντος Mortran στη Διδασκαλία Αλγοριθμικών Δομών

Ε. Κανίδης¹, Ρ. Σταθακοπούλου²

¹Σχολικός Σύμβουλος Πληροφορικής
vkaidis@di.uoa.gr

²Εκπαιδευτικός Πληροφορικής,
sreg@di.uoa.gr

Περίληψη

Στην παρούσα εργασία παρουσιάζεται μια πρόταση αξιοποίησης του περιβάλλοντος Mortran στη διδασκαλία των αλγοριθμικών δομών επανάληψης "όσο... επανάλαβε", "επανάλαβε μέχρις ότου ..." και της δομής επιλογής "Αν ... τότε ... αλλιώς". Στην εργασία παρουσιάζεται το περιβάλλον του εκπαιδευτικού μικρόκοσμου Mortran και προτείνεται μια σειρά από εκπαιδευτικές δραστηριότητες με χρήση του συγκεκριμένου περιβάλλοντος. Οι δραστηριότητες δοκιμάστηκαν πειραματικά σε μαθητές της Γ' τάξης ΕΠΑΛ ειδικότητας Πληροφορικής. Διαπιστώθηκε ότι η χρήση του περιβάλλοντος Mortran και η εκπαιδευτική διαδικασία που ακολουθήθηκε έδωσαν την ευκαιρία στους μαθητές να διερευνήσουν τα χαρακτηριστικά των αλγοριθμικών δομών και είχαν ενθαρρυντικά αποτελέσματα στη μάθηση.

Λέξεις κλειδιά: διδασκαλία αλγοριθμικών δομών, Mortran.

1. Εισαγωγή

Ο προγραμματισμός διδάσκεται σε όλους τους τύπους και τις βαθμίδες της δευτεροβάθμιας εκπαίδευσης (Γυμνάσιο, Γενικό Λύκειο, ΕΠΑΛ), καθώς και στην τριτοβάθμια εκπαίδευση. Δύο από τους στόχους της διδασκαλίας προγραμματισμού στη δευτεροβάθμια εκπαίδευση είναι οι μαθητές να μπορούν να χρησιμοποιούν προγραμματιστικές δομές και να επιλύουν προβλήματα χρησιμοποιώντας ποικίλα εργαλεία (Linn & Dalbey, 1989). Κατά τη διδασκαλία του προγραμματισμού έχει διαπιστωθεί ότι μαθητές και φοιτητές αντιμετωπίζουν δυσκολίες στην κατανόηση ορισμένων εννοιών, όπως αυτές που σχετίζονται με τις δομές επανάληψης και τις δομές επιλογής (Χαρίσης, Μικρόπουλος, 2008; Pane, & Myers, 2000; Spohrer & Soloway, 1986; Ebrahimi, 1994).

Οι παραπάνω δομές παρουσιάζουν αυξημένη πολυπλοκότητα, οι εκπαιδευόμενοι έχουν δυσκολίες στην επιλογή της κατάλληλης δομής καθώς και στην ακριβή σύνταξή της. Σύμφωνα με ορισμένους ερευνητές τα προβλήματα αυτά επιδεινώνονται από τη χρήση μη κατάλληλων εκπαιδευτικών προγραμματιστικών περιβαλλόντων. Συγκεκριμένα, υποστηρίζουν ότι τα περισσότερα προγραμματιστικά περιβάλλοντα είναι προσανατολισμένα για προγραμματιστική χρήση και όχι για τη διδασκαλία του προγραμματισμού (Ματάνας, Παπαβασιλείου, & Παπαμήτσιου, 2003; Τζιμογιάννης, 2005).

Προκειμένου να αντιμετωπιστούν οι συγκεκριμένες μαθησιακές δυσκολίες έχουν προταθεί διάφορες διδακτικές προσεγγίσεις. Ορισμένες από τις προσεγγίσεις αυτές εστιάζονται στον τρόπο με τον οποίο διδάσκονται οι αλγοριθμικές δομές και προτείνουν εναλλακτικές διδακτικές προσεγγίσεις. Παραδείγματα τέτοιων προσεγγίσεων αποτελούν η προσέγγιση «Μαύρο-Κουτί», η προσέγγιση που βασίζεται στις «Διερευνήσεις» και η προσέγγιση που βασίζεται στη συνεργασία δύο ατόμων (Γρηγοριάδου, Γόγολου, & Γούλη, 2002).

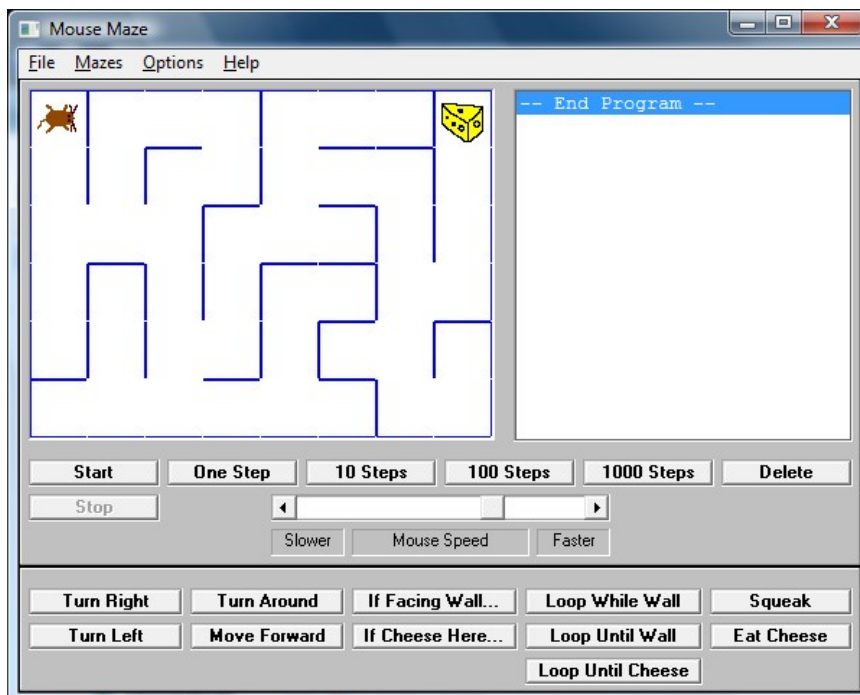
Ορισμένες άλλες διδακτικές προσεγγίσεις εστιάζονται στα μέσα που θα χρησιμοποιηθούν στη διδασκαλία των αλγοριθμικών δομών και προτείνουν τη χρήση μικρόκοσμων. Οι προγραμματιστικοί μικρόκοσμοι είναι ανοιχτά περιβάλλοντα, στα οποία συνήθως ο εκπαιδευόμενος ελέγχει τις κινήσεις μιας οντότητας που "ζει" στο περιβάλλον αυτό. Η οντότητα αυτή μπορεί να είναι μια χελώνα (Logo) ένα άλλο ζώακι (γάτα, ποντίκι κ.α) ή ένα ρομπότ. Ο εκπαιδευόμενος έχει στη διάθεσή του μια σειρά από εντολές και το αποτέλεσμα της εκτέλεσής τους είναι άμεσα ορατό στην οθόνη που εμφανίζει την οντότητα να εκτελεί μία προς μία τις εντολές που έχει εισάγει (Ξυνογαλάς, Σατρατζέμη, & Δαγδελής, 2000).

Ένας προγραμματιστικός μικρόκοσμος που μπορεί να χρησιμοποιηθεί στη διδασκαλία εισαγωγικών μαθημάτων προγραμματισμού και ιδιαίτερα στη διδασκαλία των επαναληπτικών δομών είναι το περιβάλλον Mortran. Το Mortran είναι ένα ελεύθερο (freeware) λογισμικό που έχει γραφτεί από το Mark Pelczarsk το 1995. Η διδακτική προσέγγιση που προτείνουμε για τη διδασκαλία των αλγοριθμικών δομών περιλαμβάνει

ένα συνδυασμό των ανωτέρω προσεγγίσεων και χρησιμοποιεί το μικρόκοσμο Mortran σε συνδυασμό με μια συγκεκριμένη διδακτική προσέγγιση.

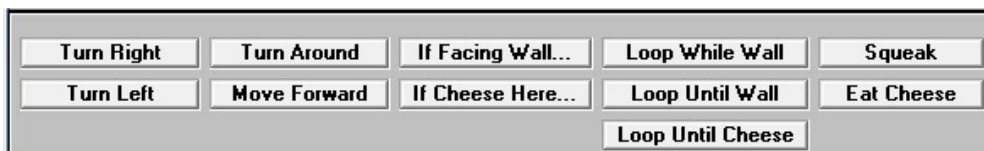
2. Το περιβάλλον Mortran

Ο προγραμματιστικός μικρόκοσμος Mortran στηρίζεται στη βασική ιδέα της ύπαρξης ενός ποντικού που προσπαθεί να βρει το δρόμο που οδηγεί σε ένα τυρί περνώντας μέσα από ένα λαβύρινθο. Σύμφωνα με ορισμένους ερευνητές η ιδέα ενός ποντικού που τρέχει σε ένα λαβύρινθο προσπαθώντας να βρει την έξοδο ή ένα αντικείμενο προκαλεί πάντα το ενδιαφέρον των μαθητών (Dorf, 1992). Ο εκπαιδευόμενος, χρησιμοποιώντας συγκεκριμένες εντολές, πρέπει να γράψει έναν αλγόριθμο ο οποίος θα οδηγήσει το ποντίκι να βρει το δρόμο προς το τυρί (Εικόνα 1). Επειδή οι κινήσεις που πρέπει να κάνει το ποντίκι εξαρτώνται από το σχήμα του λαβυρίνθου, ο διδάσκων μπορεί να χρησιμοποιήσει διαφορετικούς λαβύρινθους ανάλογα με την προγραμματιστική δομή που επιθυμεί να διδάξει. Το σχήμα του λαβυρίνθου μπορεί να μεταβληθεί εύκολα είτε με το άνοιγμα ενός έτοιμου λαβυρίνθου, που διαθέτει το πρόγραμμα (μενού Mazes), είτε με τη δημιουργία ενός νέου λαβυρίνθου από το χρήστη (Μενού Mazes, εντολή Create Maze).



Εικόνα 1: Το περιβάλλον του προγράμματος Mortran

Για τη δημιουργία του προγράμματος που θα μετακινήσει τον ποντικό από το αρχικό του σημείο στο σημείο που βρίσκεται το τυρί, ο εκπαιδευόμενος έχει διαθέσιμες εννέα εντολές. Οι εντολές εμφανίζονται με τη μορφή κουμπιών-εντολών στο κάτω μέρος του περιβάλλοντος (Εικόνα 2).



Εικόνα 2: Οι διαθέσιμες εντολές του περιβάλλοντος

Οι εννέα εντολές χωρίζονται σε τριεξομάδες, που περιγράφονται αναλυτικά στη συνέχεια. Τέσσερις εντολές αλλαγής θέσης – κατεύθυνσης:

- στρίψε δεξιά (Turn Right),
- στρίψε αριστερά (Turn Left),
- γύρισε (Turn Around),
- προχώρα ένα βήμα (Move Forward).

Δύο εντολές δομών επιλογής:

Αν συναντήσεις τοίχο τότε Αλλιώς Τέλος_Αν	If Facing Wall then Else End If
Αν το Τυρί είναι εδώ τότε Αλλιώς Τέλος_Αν	If Cheese is Here Then Else End If

Τρεις εντολές δομών επανάληψης:

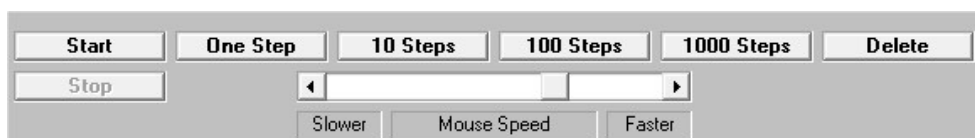
Επανάλαβε όσο βλέπεις τοίχο ... Τέλος_επανάληψης	Loop while facing wall End Loop
Επανάλαβε μέχρι να βρεις τοίχο Τέλος_επανάληψης	Loop until facing wall End Loop
Επανάλαβε μέχρι να βρεθεί το τυρί Τέλος_επανάληψης	Loop until cheese found End Loop

Τέλος, υπάρχουν άλλες δύο συμπληρωματικές εντολές:

- ένας ήχος που κάνει το ποντίκι (Squeak) και δεν έχει κανένα πρακτικό σκοπό πέρα από την ηχητική κάλυψη των κινήσεων,
- φάε το τυρί (Eat Cheese), μια εντολή που ολοκληρώνει τον αλγόριθμο και συνοδεύεται από ένα χαρακτηριστικό ήχο "φαγώματος".

Αν και όλες οι εντολές είναι στα Αγγλικά διαπιστώθηκε ότι οι μαθητές δεν αντιμετωπίζουν σοβαρό πρόβλημα καθώς οι διαθέσιμες εντολές είναι μόλις δέκα και οι λέξεις που χρησιμοποιούνται είναι τυποποιημένες και κοινές. Ο αλγόριθμος γράφεται στο πάνω δεξιό παράθυρο του περιβάλλοντος Morgran με τη χρήση των παραπάνω εντολών. Για την εκτέλεση του αλγορίθμου χρησιμοποιούνται οι εντολές που βρίσκονται στο μεσαίο τμήμα του περιβάλλοντος (Εικόνα 3).

Η εντολή Start (Έναρξη) μεταφέρει τον έλεγχο και το δρομέα στην αρχή του αλγορίθμου. Η εντολές One Step, 10 Steps, 100 Steps, 1000 Steps προκαλούν την εκτέλεση ενός 10, 100 ή 1000 βημάτων του αλγορίθμου. Η εντολή Stop χρησιμοποιείται για τον τερματισμό του αλγορίθμου (κυρίως όταν υπάρχει ατέρμων βρόχος) και η εντολή Delete (Διαγραφή) χρησιμοποιείται για τη διαγραφή εντολών του αλγορίθμου. Τέλος η ρυθμιζόμενη ράβδος χρησιμοποιείται για τον έλεγχο της ταχύτητας με την οποία το ποντίκι εκτελεί τις εντολές του αλγορίθμου.



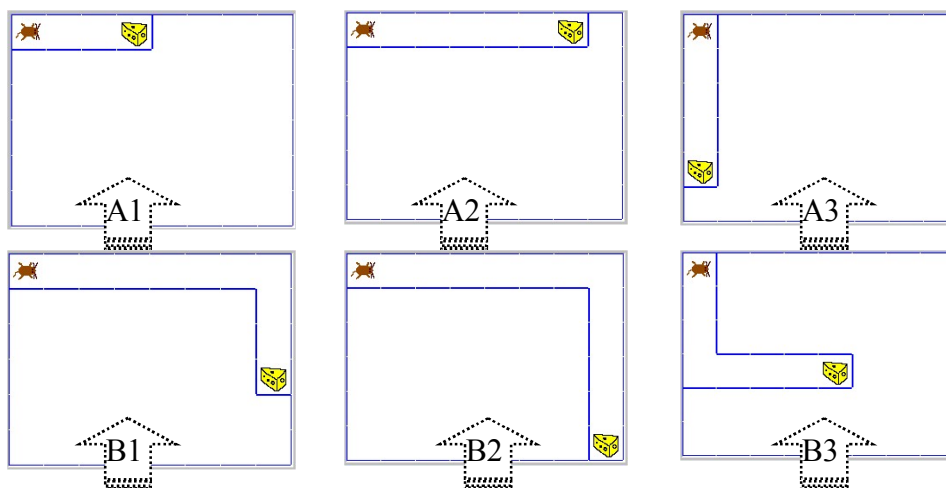
Εικόνα 3: Εντολές εκτέλεσης αλγορίθμου

3. Διαδικασία εφαρμογής των δραστηριοτήτων

3.1 Προετοιμασία

Για την πρακτική εφαρμογή των δραστηριοτήτων που αποσκοπούν στην κατανόηση των δομών επανάληψης και της δομής επιλογής πρέπει να έχει προηγηθεί η θεωρητική διδασκαλία τους. Επιπλέον πρέπει να έχει προηγηθεί η παρουσίαση από το διδάσκοντα του μικρόκοσμου και της λειτουργίας του. Οι λαβύρινθοι που χρησιμοποιούνται στην πρακτική εφαρμογή είναι οι A1, A2, A3, B1, B2, B3 και μπορούν είτε να είναι

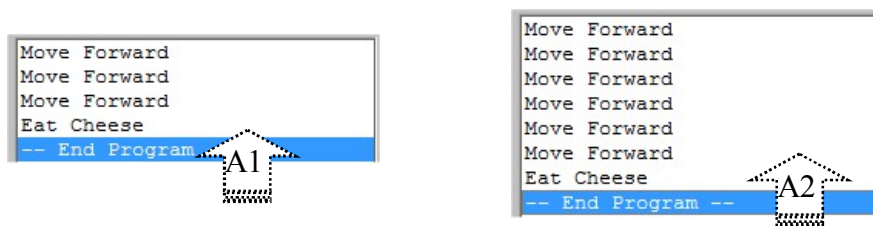
προκατασκευασμένοι είτε να κατασκευαστούν από τους μαθητές. Η μορφή τους απεικονίζεται στην εικόνα 4.



Εικόνα 4: Οι εκπαιδευτικοί λαβύρινθοι

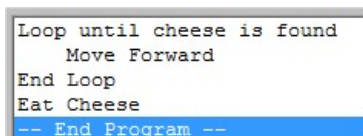
3.2 Επίλυση με δομή επανάληψης

Στο πρώτο στάδιο, αρχικά οι μαθητές καλούνται να αντιμετωπίσουν το πρόβλημα του λαβυρίνθου A1 στο οποίο το ποντίκι πρέπει να κάνει τρία βήματα μπροστά και να φάει το τυρί. Στη συνέχεια καλούνται να αντιμετωπίσουν το πρόβλημα του λαβυρίνθου A2 που αποτελεί επέκταση του A1, καθώς το ποντίκι χρειάζεται τώρα έξι βήματα μπροστά για να φτάσει το τυρί. Οι σχετικοί αλγόριθμοι εμφανίζονται στην εικόνα 5.



Εικόνα 5: Αλγόριθμοι επίλυσης των λαβυρίνθων A1 και A2

Στο δεύτερο στάδιο, ο διδάσκων προτρέπει τους μαθητές να κατασκευάσουν έναν αλγόριθμο ο οποίος θα λύνει και τους δυο παρπάνω λαβυρίνθους (A1 & A2) ανεξάρτητα από τα βήματα που πρέπει να κάνει το ποντίκι, για να φτάσει το τυρί. Αν οι μαθητές δυσκολεύονται στην εύρεση λύσης, ο διδάσκων μπορεί να τους υποδείξει ότι πρέπει να σκεφτούν μια εντολή επανάληψης σύμφωνα με την οποία το ποντίκι πρέπει να προχωράει μπροστά μέχρι να βρει το τυρί. Η εντολή αυτή είναι η Loop until cheese is found και ο αλγόριθμος που λύνει το πρόβλημα παρουσιάζεται στην εικόνα 6.



Εικόνα 6: Αλγόριθμος επίλυσης των δυο λαβυρίνθων (A1 και A2)

Οι μαθητές επιβεβαιώνουν ότι ο αλγόριθμος της εικόνας 6 λύνει κάθε πρόβλημα στο οποίο το τυρί βρίσκεται στην πρώτη γραμμή του λαβυρίνθου (ο διδάσκων μπορεί να προτρέψει τους μαθητές να δοκιμάσουν τον αλγόριθμο με το τυρί να βρίσκεται στο δεύτερο ή στο τελευταίο τετράγωνο της πρώτης γραμμής).

Στο τρίτο στάδιο, δίνεται στους μαθητές το πρόβλημα του λαβυρίνθου B1. Αρχικά γίνεται μια συζήτηση με τους μαθητές για τον τρόπο αντιμετώπισης του προβλήματος. Ο διδάσκων παρουσιάζει τη λύση με τον απλό τρόπο, κατά βήματα (βλέπε εικόνα 7 αριστερά) και ζητά από τους μαθητές να λύσουν το πρόβλημα χρησιμοποιώντας δομές επανάληψης. Προτείνεται ο διδάσκων να θέσει το ερώτημα πόσες δομές επανάληψης θα χρειαστούν για τη λύση του προβλήματος. Ο διδακτικός στόχος του παραπάνω ερωτήματος είναι οι μαθητές να εντοπίσουν ότι απαιτούνται δύο δομές επανάληψης, μια μέχρι το ποντίκι να φτάσει στο τοίχο και ακόμα μια (αφού τρίψει) μέχρι να φτάσει το τυρί. Ο αλγόριθμος που λύνει το πρόβλημα με δομές

επανάληψης εμφανίζεται στο δεξιό τμήμα της εικόνας 7. Στη συνέχεια ο διδάσκων καλεί τους μαθητές να διαπιστώσουν ότι ο αλγόριθμος που λύνει το πρόβλημα του B1 λύνει και το πρόβλημα του λαβυρίνθου B2. Τέλος, οι μαθητές καλούνται να επιλύσουν το πρόβλημα του λαβυρίνθου B3 χρησιμοποιώντας δομές επανάληψης.

```

Move Forward
Move Forward
Move Forward
Move Forward
Move Forward
Move Forward
Move Forward
Turn Right
Move Forward
Move Forward
Move Forward
Eat Cheese
-- End Program --

Loop until facing wall
  Move Forward
End Loop
Turn Right
Loop until cheese is found
  Move Forward
End Loop
Eat Cheese
-- End Program --
    
```

Εικόνα 7: Αλγόριθμοι επίλυσης του λαβυρίνθου B1

3.3 Επίλυση με δομή επιλογής

Σε αυτό το στάδιο οι μαθητές καλούνται να αντιμετωπίσουν το πρόβλημα του λαβυρίνθου B2 αλλά με διαφορετική λογική η οποία παρουσιάζεται από το διδάσκοντα και είναι η ακόλουθη. "Το ποντίκι τώρα είναι πιο επιφυλακτικό και εξετάζει το κάθε του βήμα προσεκτικά. Συγκεκριμένα, πριν από κάθε βήμα εξετάζει αν βρίσκεται μπροστά σε κάποιο τοίχο και τότε αν πράγματι βρίσκεται, κάνει μια στροφή δεξιά, σε διαφορετική περίπτωση (αν δεν είναι μπροστά σε τοίχο) κάνει ένα βήμα μπροστά. Η παραπάνω διαδικασία επαναλαμβάνεται μέχρι το ποντίκι να βρει το τυρί". Ο αλγόριθμος που υλοποιεί την παραπάνω λογική παρουσιάζεται στην εικόνα 8.

```

Loop until cheese is found
  If Facing Wall Then
    Turn Right
  Else
    Move Forward
  End If
End Loop
Eat Cheese
-- End Program --
    
```

Εικόνα 8: Αλγόριθμος επίλυσης του λαβυρίνθου B2

Ο καθηγητής θέτει το ερώτημα αν ο παραπάνω αλγόριθμος μπορεί να χρησιμοποιηθεί για τη λύση του λαβυρίνθου B3. Αφού γίνει η σχετική συζήτηση και ο καθηγητής σημειώσει και ομαδοποιήσει στον πίνακα τις σχετικές απόψεις οι μαθητές θα πρέπει να εφαρμόσουν τον αλγόριθμο, για να διαπιστώσουν αν πράγματι λύνει το πρόβλημα του λαβυρίνθου B3. Οι μαθητές θα παρατηρήσουν την κίνηση του ποντικιού και θα βεβαιωθούν ότι το ποντίκι κινείται μόνο στον κάθετο διάδρομο και ο λαβύρινθος δεν λύνεται. Στη συνέχεια οι μαθητές καλούνται να εξηγήσουν τη συμπεριφορά αυτή και να διορθώσουν τον αλγόριθμο. Εύκολα θα βρουν ότι τώρα ο ποντικός θα πρέπει να στρίβει αριστερά αντί για δεξιά.

```

Loop until cheese is found
  If Facing Wall Then
    Turn Right
    If Facing Wall Then
      Turn Around
    Else
      End If
  Else
    Move Forward
  End If
End Loop
Eat Cheese
-- End Program --
    
```

Εικόνα 9: Αλγόριθμος επίλυσης των λαβυρίνθων A3, B2 και B3

Ο καθηγητής μπορεί να θέσει το ερώτημα αν είναι δυνατή η δημιουργία ενός αλγορίθμου που θα έλυνε το πρόβλημα στη γενική του μορφή. Δηλαδή έναν αλγόριθμο που θα έλυνε και το λαβύρινθο A3, και το B2 και

το B3. Η λογική δομή του αλγορίθμου (ή οποία μπορεί να δοθεί από τον καθηγητή) στηρίζεται στην προηγούμενη λύση αλλά επιπλέον έχει την επέκταση ότι όταν το ποντίκι φτάσει σε κάποια "γωνία" ή "διασταύρωση", δεν αρκεί μια μόνο στροφή δεξιά (ή αριστερά), θα πρέπει να γίνει ξανά έλεγχος μήπως υπάρχει τοίχος μπροστά στο ποντίκι και αν υπάρχει, θα πρέπει να ακολουθήσει αντίθετη πορεία. Ο αλγόριθμος που υλοποιεί την παραπάνω διαδικασία παρουσιάζεται στην εικόνα 9.

4. Πειραματική εφαρμογή

Η προτεινόμενη διαδικασία διδασκαλίας εφαρμόστηκε τη σχολική χρονιά 2008-09 σε μια ομάδα 20 μαθητών της Γ' τάξης ΕΠΑΛ ειδικότητας Πληροφορικής. Οι μαθητές της Γ' τάξης Πληροφορικής, σύμφωνα με το Πρόγραμμα Σπουδών, δεν έχουν διδαχθεί τις δομές επανάληψης και επιλογής στη Β' τάξη, παρά το γεγονός ότι έχουν παρακολουθήσει αρκετά μαθήματα ειδικότητας. Οι μαθητές είχαν διδαχθεί τη θεωρία των δομών επανάληψης και επιλογής στη Γ' τάξη στα πλαίσια του μαθήματος Δομημένος Προγραμματιστικός, το οποίο εξετάζεται για την εισαγωγή τους στις Ανώτατες Σχολές. Το συγκεκριμένο μάθημα θεωρείται δύσκολο από τους μαθητές και η ανάπτυξη προγραμμάτων δεν τους κινεί το ενδιαφέρον επειδή δε βλέπουν άμεσες πρακτικές εφαρμογές. Στη συγκεκριμένη ομάδα μαθητών διαπιστώθηκε ότι οι περισσότεροι από αυτούς δυσκολεύονταν να εφαρμόσουν τις δομές επανάληψης και επιλογής στην επίλυση προβλημάτων που εμφανίζονται στα προγραμματιστικά περιβάλλοντα. Επιπλέον δεν είχαν σαφή εικόνα της διαφοράς των αλγοριθμικών δομών επανάληψης "όσο...επανάλαβε" και "επανάλαβε μέχρις ότου...". Σε σχετική ερώτηση απάντησε μόνο ένας μαθητής.

Αρχικά, ζητήθηκε από τους μαθητές να κατασκευάσουν τους λαβυρίνθους A1, A2, A3, B1, B2, B3, της εικόνας 4, έτσι ώστε με την κατασκευή τους να παρατηρήσουν προσεκτικά τις διαφορές τους. Οι μαθητές εργάστηκαν σε ομάδες των δύο ατόμων σε κάθε υπολογιστή. Στη συνέχεια ακολουθήθηκαν τα στάδια που παρουσιάζονται στην ενότητα 3. Επιπλέον, ζητήθηκε από τους μαθητές να δικαιολογήσουν το λόγο για τον οποίο επέλεξαν ανάμεσα στις δομές "όσο... επανάλαβε" και "επανάλαβε μέχρις ότου...".

Η χρήση ενός πρωτότυπου περιβάλλοντος με άμεση ανάδραση προκάλεσε το ενδιαφέρον των μαθητών και τη δραστηριοποίησή τους. Συγκεκριμένα συμμετείχαν ενεργά 14 μαθητές ενώ σε παρόμοια προβλήματα που τους είχαν δοθεί με μολύβι και χαρτί ή σε άλλα περιβάλλοντα προγραμματισμού συμμετείχαν μόνον 8.

Μετά το τέλος της δραστηριότητας οι 8 από τους 14 μαθητές έδειξαν ότι έχουν κατανοήσει τη διαφορά μεταξύ των δομών "όσο... επανάλαβε" και "επανάλαβε μέχρις ότου...". Ενδεικτικά απάντησαν "έβαλα until, γιατί πρέπει να προχωράει μέχρι να βρει τον τοίχο ή το τυρί και όχι να κάνει κάτι, όταν το βρει".

5. Συμπεράσματα

Τα αποτελέσματα της παρούσας εργασίας οδηγούν στο συμπέρασμα ότι η χρήση του περιβάλλοντος Mortran με τη χρήση κατάλληλων εκπαιδευτικών ασκήσεων βελτιώνει την κατανόηση των αλγοριθμικών δομών.

Στο περιβάλλον Mortran οι μαθητές εύκολα ανακαλύπτουν το είδος της προγραμματιστικής δομής που πρέπει να χρησιμοποιηθεί για την επίλυση μιας άσκησης. Ενδεχομένως η δυνατότητα άμεσης ανάδρασης, οι περιορισμένες προγραμματιστικές δομές που παρέχει το περιβάλλον Mortran και η τυποποιημένη δομή τους διευκολύνουν την επιλογή των μαθητών.

Κατά τη διάρκεια υλοποίησης της πρακτικής άσκησης οι μαθητές ενεπλάκησαν ενεργά στη μαθησιακή διαδικασία, ανέπτυξαν δεξιότητες κριτικής σκέψης μέσα από μια διαδικασία οικοδόμησης της γνώσης και έδειξαν ενδιαφέρον για πειραματισμό με το συγκεκριμένο περιβάλλον. Τέλος το περιβάλλον Mortran φάνηκε φιλικό και ενδιαφέρον σε όλους τους μαθητές, αφού εξοικειώθηκαν εύκολα με τα εικονίδια που περιέχει και τις διαθέσιμες εντολές.

Η παρούσα εργασία αποτέλεσε μια προκαταρκτική έρευνα. Στα μελλοντικά μας σχέδια περιλαμβάνεται η ανάπτυξη επιπλέον δραστηριοτήτων οι οποίες θα συνδυάζουν τη χρήση του περιβάλλοντος Mortran με συγκεκριμένες διδακτικές παρεμβάσεις για την πληρέστερη κατανόηση των αλγοριθμικών δομών από τους μαθητές διαφόρων τύπων σχολείων (Γυμνάσιο, Γεν. Λύκειο). Επίσης η σύνταξη ειδικών ερωτηματολογίων για την πληρέστερη αποτύπωση της κατανόησης των δομών από τους μαθητές.

Βιβλιογραφία

- Dorf, M.L.,(1992), Backtracking the Rat Way, *SIGCSE Bulletin*, vol.24, p 272-276.
- Du Boulay, B. (1989), Some difficulties of learning to program, In E. Soloway & J. C. Spohrer (Eds), *Studying the Novice Programmer*, Hillsdale, NJ, Lawrence Erlbaum Associates , 283-299.
- Ebrahimi, A. (1994). Novice programmer errors: language constructs and plan composition, *Int. J. Human-Computer Studies*, 41, 457-480.
- Linn, M. & Dalbey, J. (1989), Cognitive Consequences of Programming Instruction, In E. Soloway & J. C. Spohrer (Eds), *Studying the Novice Programmer*, 57-81, Hillsdale, NJ, Lawrence Erlbaum Associates.
- Mortran Freeware program by Mark Pelczarsk, [http://www.wsd1.org/LTCActivities/ K3%20Freeware /Mortran%20Mouse%20Maze.htm](http://www.wsd1.org/LTCActivities/K3%20Freeware/Mortran%20Mouse%20Maze.htm). Πρόσβαση 11/2008
- Pane, J. & Myers, B. (2000). The Influence of the Psychology of Programming on a Language Design: Project Status Report, *Proceedings of the 12th Annual Meeting of the Psychology of Programming Interest Group*, 193-205, Edizioni Memoria, Italy.
- Putnum, R. T., Sleeman, D., Baxter, J. & Kupsa, L. (1989). A summary of the misconceptions of high school BASIC programmers, In E. Soloway & J. C. Spohrer (Eds), *Studying the Novice Programmer*, 301-314, Hillsdale, NJ, Lawrence Erlbaum Associates.
- Spohrer, J. & Soloway, E. (1986). Alternatives to Construct-Based Programming Misconceptions, *Proceedings of the CHI '86*, 183-191.
- Γρηγοριάδου, Μ., Γόγουλου, Α., Γουλή, Ε. (2002), Εναλλακτικές διδακτικές προσεγγίσεις σε εισαγωγικά μαθήματα προγραμματισμού: Προτάσεις διδασκαλίας. Στα: Α.Δημητρακοπούλου (Επιμ.): *Πρακτικά 3ου Συνεδρίου ΕΤΠΕ «Οι ΤΠΕ στην Εκπαίδευση»*, 26-29 Σεπτεμβρίου 2002, Πανεπιστήμιο Αιγαίου, Ρόδος, Εκδόσεις Καστανιώτη, Τόμος Α, 239-248.
- Ματάνας, Ν., Παπαβασιλείου, Α., & Παπαμήτσιου, Ζ., (2003). Τα προβλήματα της διδασκαλίας της Πληροφορικής στη Δευτεροβάθμια Εκπαίδευση. *Πρακτικά 2ου Συνεδρίου Διδακτική της Πληροφορικής*, Βόλος, 302-306.
- Ξυνογαλάς, Σ., Σατρατζέμη, Μ. & Δαγδύλης, Β. (2000), Η εισαγωγή στον προγραμματισμό: Διδακτικές Προσεγγίσεις και Εκπαιδευτικά Εργαλεία, Στο Β. Κόμης (Επιμ.), *Πρακτικά Εισηγήσεων 2ου Πανελληνίου Συνεδρίου «Οι Τεχνολογίες της Πληροφορίας και της Επικοινωνίας στην Εκπαίδευση»*, 115-124, Πάτρα.
- Τζιμογιάννης, Α., (2005), Προς ένα Παιδαγωγικό Πλαίσιο Διδασκαλίας του Προγραμματισμού στη Δευτεροβάθμια Εκπαίδευση. *Πρακτικά 3ου Συνεδρίου Διδακτική της Πληροφορικής*, Κόρινθος.
- Χαρίσης, Χ., Μικρόπουλος, Τ., Ρομποτική, Οπτικός Προγραμματισμός και Βασικές Προγραμματιστικές Δομές, *Πρακτικά 4ου Συνεδρίου για τη Διδακτική της Πληροφορικής*, Πάτρα 2008, 121-130.