

Πρόταση για τη Διδασκαλία του Προγραμματισμού στο Γυμνάσιο με Χρήση του Ρομπότ Karel

Ξυνόγαλος Στέλιος

**Καθηγητής Πληροφορικής, 2^ο Πειραματικό Γυμνάσιο Θεσσαλονίκης
Αιδάσκοντας ΠΔ 407/80, Τμ. Διοίκησης Τεχνολογίας, Παν. Μακεδονίας
stelios@uom.gr**

ΠΕΡΙΛΗΨΗ

Στην παρούσα εργασία παρουσιάζεται μια πρόταση διδασκαλίας του προγραμματισμού σε μαθητές Γυμνασίου χρησιμοποιώντας τον μικρόκοσμο προγραμματισμού Karel, ο οποίος αναπτύχθηκε πρόσφατα στα πλαίσια του έργου ΕΠΕΑΕΚ «Πυθαγόρας II». Η πρωτοτυπία του μικρόκοσμου έγκειται στο γεγονός ότι ενσωματώνει την προτεινόμενη σειρά μαθημάτων και δραστηριοτήτων, ένα συντάκτη δομής για την ανάπτυξη των προγραμμάτων, τη δυνατότητα βηματικής εκτέλεσης και επεξηγηματικής οπτικοποίησης. Στόχος της εργασίας αποτελεί η παρουσίαση στην εκπαιδευτική κοινότητα μιας ολοκληρωμένης πρότασης διδασκαλίας του προγραμματισμού στο Γυμνάσιο, η οποία περιλαμβάνει μια συνοπτική περιγραφή των λειτουργιών του μικρόκοσμου προγραμματισμού Karel και μια περιγραφή της οργάνωσης των μαθημάτων ανά διδακτική ώρα (θεωρία, δραστηριότητες, ασκήσεις). Επίσης, παρουσιάζονται τα πρώτα αποτελέσματα από την εφαρμογή της συγκεκριμένης πρότασης διδασκαλίας στην τάξη.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Προγραμματισμός, μικρόκοσμος προγραμματισμού.

ΕΙΣΑΓΩΓΗ

Η διδασκαλία του προγραμματισμού σε αρχάριους έχει αποτελέσει αντικείμενο μελέτης μεγάλου αριθμού ερευνητών και εκπαιδευτικών τις τελευταίες δεκαετίες, μιας και συνοδεύεται από ποικίλες δυσκολίες. Ένα κοινό συμπέρασμα όλων των ερευνών είναι το γεγονός ότι ο προγραμματισμός αποτελεί ένα ιδιαίτερα δύσκολο γνωστικό αντικείμενο και η διδασκαλία του απαιτεί τη σχεδίαση ιδιαίτερων διδακτικών καταστάσεων. Όσο μικρότερη είναι η ηλικία των εκπαιδευομένων μάλιστα τόσο μεγαλύτερη προσοχή απαιτείται στην οργάνωση των μαθημάτων εισαγωγής στον προγραμματισμό. Μιας και στα Ελληνικά σχολεία η εισαγωγή των μαθητών στον προγραμματισμό πραγματοποιείται στη Γ' τάξη του Γυμνασίου ιδιαίτερο ενδιαφέρον παρουσιάζεται από Έλληνες ερευνητές και εκπαιδευτικούς για τον εντοπισμό μιας αποτελεσματικής προσέγγισης διδασκαλίας του προγραμματισμού σε μαθητές Γυμνασίου (Γόγουλου κ.α., 2008; Καγκάνη κ.α., 2005; Μπέλλου και Μικρόπουλος, 2005).

Στην παρούσα εργασία παρουσιάζεται μια ολοκληρωμένη πρόταση διδασκαλίας του προγραμματισμού στο Γυμνάσιο, η οποία βασίζεται στην προσέγγιση των μικρόκοσμων προγραμματισμού (Brusilovsky et al., 1997). Συγκεκριμένα, παρουσιάζεται η οργάνωση μιας σειράς μαθημάτων που βασίζονται στον μικρόκοσμο προγραμματισμού Karel. Ο μικρόκοσμος Karel ενσωματώνει εκτός από το περιβάλλον προγραμματισμού και την προτεινόμενη σειρά μαθημάτων και δραστηριοτήτων. Στην εργασία παρουσιάζονται το περιεχόμενο, οι δραστηριότητες και οι ασκήσεις του κάθε μαθήματος ανά διδακτική ώρα, καθώς επίσης και τα πρώτα αποτελέσματα από την εφαρμογή της συγκεκριμένης πρότασης σε μαθητές Γυμνασίου.

www.e-diktyo.eu

www.epyna.gr

ΤΑ ΠΑΙΔΑΓΩΓΙΚΑ ΟΦΕΛΗ ΤΟΥ ΜΙΚΡΟΚΟΣΜΟΥ Karel

Ο μικρόκοσμος προγραμματισμού Karel αποτελεί επέκταση του μικρόκοσμου objectKarel, ο οποίος αποσκοπεί στη στήριξη της διδασκαλίας του αντικειμενοστραφούς προγραμματισμού. Η επιτυχημένη χρήση του μικρόκοσμου objectKarel οδήγησε στην απόφαση επέκτασης του για την υποστήριξη της διδασκαλίας του διαδικαστικού προγραμματισμού, ο οποίος διδάσκεται τις τελευταίες δεκαετίες στα Ελληνικά Γυμνάσια. Ο μικρόκοσμος Karel αναπτύχθηκε στα πλαίσια του έργου ΕΠΕΑΕΚ με τίτλο «ΠΥΘΑΓΟΡΑΣ II» και χρησιμοποιεί μια C-like έκδοση της γλώσσας προγραμματισμού του ρομπότ Karel (Pattis et al., 1995). Οι δύο μικρόκοσμοι χρησιμοποιούν ένα γραφικό ενδιάμεσο με την ίδια εμφάνιση, τις ίδιες λειτουργίες και κοινό τρόπο εκτέλεσης των λειτουργιών, έτσι ώστε να καταστεί πιο εύκολη και αποτελεσματική η διδασκαλία και εκμάθηση των δύο προαναφερθέντων παραδειγμάτων προγραμματισμού.

Οι λειτουργίες που ενσωματώνει ο προγραμματιστικός μικρόκοσμος Karel είναι οι εξής:

- Μια **σειρά μαθημάτων**, κάθε ένα από τα οποία περιλαμβάνει σύντομη και περιεκτική θεωρία και μία ή περισσότερες δραστηριότητες.
- Ένα **συντάκτη δομής**, ο οποίος επιτρέπει την ανάπτυξη προγραμμάτων μέσω ενός μενού με τις διαθέσιμες εντολές και πλαισίων διαλόγου. Το μενού ενημερώνεται αυτόματα με τις νέες εντολές που ορίζει ο μαθητής.
- Ένα σύστημα δυναμικής προσομοίωσης εκτέλεσης των προγραμμάτων που παρέχει τη δυνατότητα της **βηματικής εκτέλεσης** και της **επεξηγηματικής οπτικοποίησης**, εμφάνισης δηλαδή επεξηγήσεων για τη σημασία της τρέχουσας εντολής χρησιμοποιώντας φυσική γλώσσα.
- Καταγραφή των ενεργειών των σπουδαστών και παρουσίαση τους σε ένα διαφορετικό παράθυρο.

Στον Πίνακα 1 παρουσιάζονται τα παιδαγωγικά οφέλη που απορρέουν από τα προαναφερθέντα χαρακτηριστικά του μικρόκοσμου Karel και τα οποία τον καθιστούν ιδανικό για μαθήματα εισαγωγής στον προγραμματισμό, ιδιαίτερα σε μικρές ηλικίες.

Χαρακτηριστικό	Παιδαγωγικά οφέλη
Χρήση της προσέγγισης των μικρόκοσμων	εξαλείφει το πρόβλημα του εκτεταμένου ρεπερτορίου εντολών και την πολυπλοκότητα των γλωσσών προγραμματισμού γενικού σκοπού μειώνει τη διανοητική “απόσταση” ανάμεσα στα νοητικά μοντέλα ή τις περιγραφές των αλγορίθμων σε φυσική γλώσσα και την περιγραφή τους στη γλώσσα προγραμματισμού επιλύονται προβλήματα που προκαλούν το ενδιαφέρον των μαθητών
Ενσωμάτωση τμήματος με μαθήματα (θεωρία, δραστηριότητες)	όλο το απαραίτητο υλικό για το μαθητή συμπεριλαμβάνεται στο περιβάλλον οι δραστηριότητες βοηθάνε τους μαθητές να εξοικειωθούν με τις έννοιες πριν να τους ζητηθεί να τις υλοποιήσουν
Συντάκτης δομής	δεν χρειάζεται να απομνημονευθούν οι συντακτικές λεπτομέρειες της γλώσσας προγραμματισμού και δίνεται η δυνατότητα επικέντρωσης στις έννοιες

<i>Φιλικά προς το χρήστη μηνύματα λάθους</i>	αποφεύγεται η χρονοβόρα διαδικασία εντοπισμού και διόρθωσης των λαθών, η οποία συνήθως προκαλεί την απογοήτευση των μαθητών οι λανθασμένες αντιλήψεις, στις οποίες συνήθως οφείλονται τα λάθη, αποκαλύπτονται στους μαθητές
<i>Βηματική εκτέλεση και επεξηγηματική οπτικοποίηση</i>	αποκαλύπτει τη δυναμική φύση της εκτέλεσης προγραμμάτων στηρίζει τους μαθητές στην κατανόηση της σημασιολογίας των δομών ελέγχου και της ροής εκτέλεσης υποστηρίζει τους μαθητές στον εντοπισμό λογικών λαθών δίνει τη δυνατότητα στο διδάσκοντα να καλύψει περισσότερη ύλη σε μικρότερο χρονικό διάστημα
<i>Καταγραφή των ενεργειών των μαθητών</i>	στηρίζει τους διδάσκοντες στη μελέτη των δυσκολιών και των λανθασμένων αντιλήψεων, καθώς και των τεχνικών επίλυσης προβλημάτων που χρησιμοποιούν οι μαθητές

Πίνακας 1: Τα παιδαγωγικά οφέλη του μικρόκοσμου προγραμματισμού Karel

ΠΡΟΤΕΙΝΟΜΕΝΗ ΟΡΓΑΝΩΣΗ ΤΩΝ ΜΑΘΗΜΑΤΩΝ

Στην ενότητα αυτή παρουσιάζεται μια ενδεικτική οργάνωση των μαθημάτων, η οποία εφαρμόζεται για τη διδασκαλία του προγραμματισμού στη Γ' τάξη του 2^{ου} Πειραματικού Γυμνασίου Θεσσαλονίκης την τρέχουσα σχολική χρονιά. Στον Πίνακα 2 παρουσιάζεται συνοπτικά η οργάνωση και το περιεχόμενο της προτεινόμενης σειράς μαθημάτων και στη συνέχεια δίνεται μια πιο αναλυτική περιγραφή.

Διδακτική ώρα	Τίτλος μαθήματος	Περιεχόμενο
1 ^η – 2 ^η	Βασικές εντολές	Περιγραφή του μικρόκοσμου, πρόγραμμα, βασικές εντολές, μπλοκ εντολών, κυρίως πρόγραμμα
3 ^η – 4 ^η	Δημιουργία νέων εντολών	Ορισμός νέας εντολής, τα πλεονεκτήματα της δημιουργίας νέων εντολών
5 ^η – 6 ^η	Δομή επιλογής	Συνθήκη, λογικές τιμές, χρησιμότητα των εντολών υπό συνθήκη, if, if/else
7 ^η – 8 ^η	Δομές επανάλ.	while, loop

Πίνακας 2: Το περιεχόμενο της προτεινόμενης σειράς μαθημάτων

1^η Διδακτική Ώρα: παρουσίαση του μικρόκοσμου και των βασικών εντολών

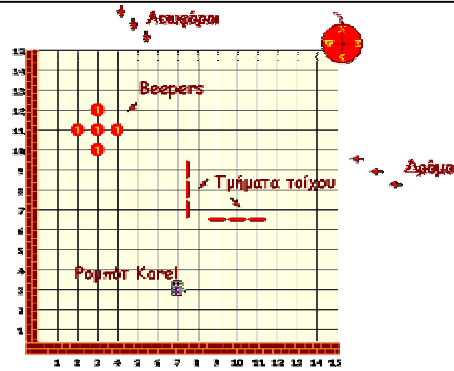
Την 1^η διδακτική ώρα παρουσιάζεται στους μαθητές ο μικρόκοσμος του ρομπότ Karel και οι βασικές εντολές που μπορεί να εκτελέσει, χρησιμοποιώντας τις ενότητες «Εισαγωγή» και «Βασικές Εντολές» που ενσωματώνονται στο τμήμα των μαθημάτων του περιβάλλοντος. Αρχικά, χρησιμοποιείται η ενότητα «Εισαγωγή» για την παρουσίαση του μικρόκοσμου (χρησιμοποιώντας το κείμενο που ακολουθεί).

Ο Karel είναι ένα ρομπότ που ζει σε ένα κόσμο που αποτελείται από οριζόντιους δρόμους και κάθετες λεωφόρους που διασταυρώνονται δημιουργώντας μπλοκ (τμήματα δρόμου ή λεωφόρου) του ίδιου μήκους. Οι δρόμοι και οι λεωφόροι είναι αριθμημένες και η θέση του Karel μέσα στον κόσμο προσδιορίζεται από τον αριθμό του δρόμου και της

www.e-diktyo.eu

www.epyna.gr

λεωφόρου, δηλαδή της διασταύρωσης που βρίσκεται το ρομπότ (1ος δρόμος – 4η λεωφόρος \Rightarrow διασταύρωση (1, 4). Δυτικά και νότια αυτού του κόσμου υπάρχουν ένας κάθετος και ένας οριζόντιος τοίχος αντίστοιχα, που εμποδίζουν την έξοδο του Karel εκτός των ορίων του κόσμου. Στον κόσμο αυτό εκτός από τον Karel είναι δυνατόν να υπάρχουν:



- τμήματα τοίχου μήκους ενός μπλοκ που εμποδίζουν την άμεση μετάβαση του Karel από τη μια διασταύρωση στην άλλη και
- beepers, μικροί πλαστικοί κώνοι που παράγουν ένα ήχο (μπιπ).

Στον απλό αυτό κόσμο ο Karel καλείται να φέρει εις πέρας συγκεκριμένες αποστολές, όπως για παράδειγμα να κατορθώσει να βγει από ένα λαβύρινθο, να «θερίσει» ένα χωράφι από beepers που έχουν μια συγκεκριμένη διάταξη κ.τ.λ. Το ρομπότ Karel όμως δεν διαθέτει νοημοσύνη και δεν έχει τη δυνατότητα να φέρει εις πέρας τις αποστολές αν δεν του δοθούν ακριβείς οδηγίες (εντολές). Συγκεκριμένα, ο Karel ενεργεί διαβάζοντας και ακολουθώντας ένα τέτοιο σύνολο εντολών που ονομάζεται πρόγραμμα. Ο Karel, ωστόσο, έχει περιορισμένες δυνατότητες. Αν ο Karel δεν μπορεί να ολοκληρώσει μια αποστολή ή είναι δύσκολο να καθορίσει ένα αποτελεσματικό τρόπο αντιμετώπισης του προβλήματος, μπορούμε να ορίσουμε νέες εντολές.

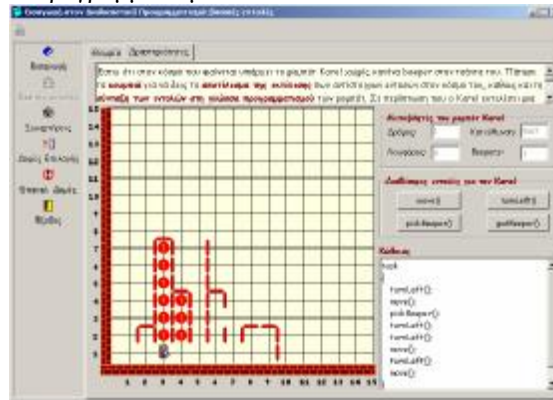
Η γλώσσα που χρησιμοποιούμε για να προγραμματίσουμε το ρομπότ Karel δεν είναι η φυσική μας γλώσσα, αλλά μια ειδική γλώσσα προγραμματισμού που όπως η φυσική μας γλώσσα έτσι και αυτή έχει λεξιλόγιο, σημεία στίξης και κανόνες γραμματικής.

Στη συνέχεια, παρουσιάζονται στους μαθητές οι τέσσερις βασικές εντολές που μπορεί να εκτελέσει ο Karel:

- `move()`: μετακίνηση προς τα εμπρός κατά ένα μπλοκ, χωρίς να αλλάζει κατεύθυνση ο Karel. Για την αποφυγή ζημιάς ο Karel δεν προχωράει αν διαπιστώσει (με την κάμερα που διαθέτει) ότι μπροστά του υπάρχει τοίχος. Αντίθετα, τερματίζει τη λειτουργία του με μια ενέργεια γνωστή ως κλείσιμο λόγω λάθους, `error shutoff` στη γλώσσα του Karel.
- `turnLeft()`: ο Karel στρίβει κατά 90 μοίρες προς τα αριστερά παραμένοντας στην ίδια διασταύρωση.
- `pickBeeper()`: ο Karel σηκώνει με το μηχανικό του χέρι ένα beeper από τη διασταύρωση που βρίσκεται και το τοποθετεί στην ηχομονωμένη τσάντα του. Αν δεν υπάρχει κανένα beeper στη διασταύρωση ο Karel προχωρά σε κλείσιμο λόγω λάθους.
- `putBeeper()`: ο Karel βγάζει ένα beeper από την τσάντα του και το τοποθετεί στη διασταύρωση που βρίσκεται. Αν δεν υπάρχει κανένα beeper στην τσάντα ο Karel προχωρά σε κλείσιμο λόγω λάθους.

Στα πλαίσια της δραστηριότητας της ενότητας «Βασικές Εντολές» οι μαθητές κατευθύνουν τον Karel να φέρει εις πέρας την πρώτη του αποστολή πατώντας κουμπιά που έχουν ως ετικέτες τις τέσσερις βασικές εντολές. Οι μαθητές μπορούν να δουν (Εικόνα 1): (1) ποιο είναι το αποτέλεσμα εκτέλεσης της κάθε εντολής από τον Karel, (2) πώς

μεταβάλλονται οι τιμές των τεσσάρων μεταβλητών που ενημερώνονται με τις μεταβολές της κατάστασης του Karel, (3) τη σύνταξη των εντολών, τις οποίες εκτελούν πατώντας κουμπιά, στη γλώσσα προγραμματισμού.

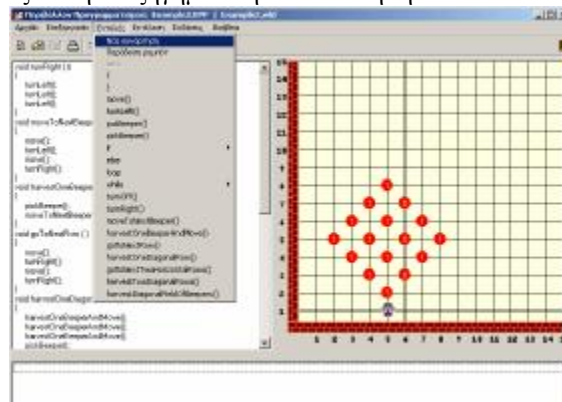


Εικόνα 1: Η δραστηριότητα της ενότητας «Βασικές Εντολές».

2^η Διδακτική Ώρα: ανάπτυξη προγραμμάτων με τις βασικές εντολές

Οι μαθητές αναπτύσσουν προγράμματα χρησιμοποιώντας τις βασικές εντολές. Το πρώτο πρόγραμμα αναπτύσσετε με τη βοήθεια του διδάσκοντα προκειμένου να παρουσιαστεί στους μαθητές:

- ο τρόπος χρήσης του συντάκτη δομής για την ανάπτυξη και διόρθωση ενός προγράμματος (Εικόνα 2)
- η διαδικασία της μεταγλώττισης
- η διαδικασία της αποσφαλμάτωσης
- η αξιοποίηση των πληροφοριών που παρέχει η βηματική εκτέλεση ενός προγράμματος και η επεξηγηματική οπτικοποίηση.

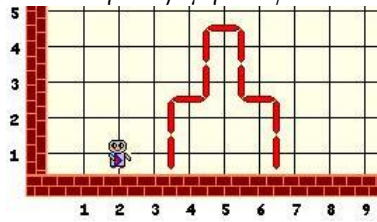


Εικόνα 2: Ο συντάκτης δομής του μικρόκοσμου προγραμματισμού Karel.

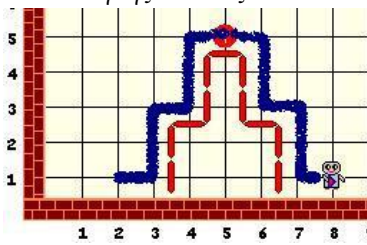
Στη συνέχεια, παρουσιάζεται ενδεικτικά μία από τις προτεινόμενες ασκήσεις.

Τα τμήματα τοίχου στην παρακάτω Εικόνα αναπαριστούν ένα βουνό. Προγραμματίστε τον Karel ώστε να ανέβει στην κορυφή του βουνού και να τοποθετήσει μια σημαία, που

αναπαριστάνεται με ένα *beeper*. Στη συνέχεια ο *Karel* πρέπει να κατέβει από την άλλη μεριά του βουνού. Υπέθεσε ότι ο *Karel* ξεκινάει με μια σημαία-*beeper* στην τσάντα του και ακολουθεί τη διαδρομή που φαίνεται στην τελική κατάσταση της Εικόνας.



Αρχική κατάσταση



Τελική κατάσταση

Η αποστολή αναρρίχησης στο βουνό και τοποθέτησης της σημαίας.

3^η Διδακτική Ώρα: παρουσίαση της δυνατότητας δημιουργίας νέων εντολών

Χρησιμοποιώντας την ενότητα «Συναρτήσεις» του τμήματος των μαθημάτων παρουσιάζεται στους μαθητές η ανάγκη δημιουργίας νέων εντολών που προκύπτει σε πολλές περιπτώσεις. Προκειμένου να κατανοήσουν οι μαθητές την αξία της δυνατότητας δημιουργίας νέων εντολών χρησιμοποιείται ένα απλό πρόβλημα για το οποίο, όμως, απαιτείται η ανάπτυξη ενός προγράμματος μεγάλης έκτασης που είναι δύσκολο να αποσφραλιστεί ή να τροποποιηθεί για την επίλυση ενός παρόμοιου προβλήματος. Στη συνέχεια, παρατίθεται μέρος του κειμένου της ενότητας.

Πολλές φορές ακόμα και για απλά προβλήματα απαιτείται η ανάπτυξη προγραμμάτων που αποτελούνται από πάρα πολλές εντολές. Έστω για παράδειγμα ότι πρέπει να προγραμματίσουμε τον *Karel* έτσι ώστε να ταξιδεύει διανύοντας μεγάλες αποστάσεις (πρόβλημα του ρομπότ-ταξιδιώτη). Ας υποθέσουμε ότι ο *Karel* ξεκινώντας από την 1^η λεωφόρο και τον 2^ο δρόμο πρέπει να κινηθεί ανατολικά κατά μήκος του 2^{ου} δρόμου για 10 χιλιόμετρα (1 χλμ = 8 μπλοκ), να σηκώσει ένα *beeper* και στη συνέχεια να κινηθεί 5 χλμ βόρεια. Αφού το ρομπότ *Karel* μπορεί να κινείται κατά ένα μπλοκ και δεν κατανοεί την έννοια του χιλιομέτρου πρέπει να μεταφράσουμε τη λύση μας σε εντολές που μετακινούν τον *Karel* κατά ένα μπλοκ κάθε φορά. Αυτό σημαίνει ότι το πρόγραμμά μας θα περιλαμβάνει 120 εντολές *move()*, θα έχει επομένως μεγάλη έκταση και θα είναι δύσκολο να διορθωθεί (αποσφραλιστεί) σε περίπτωση λάθους και πολύ περισσότερο να τροποποιηθεί προκειμένου να καθοδηγήσει τον *Karel* στη διεξαγωγή μιας παρόμοιας αποστολής. Στη συνέχεια παρατίθεται το πρόγραμμα για το πρόβλημα του ρομπότ-ταξιδιώτη (παρουσιάζεται σε πλήρη ανάπτυξη στους μαθητές):

```
task
{
    move(); // η εντολή επαναλαμβάνεται 80 φορές
    ...
    pickBeeper();
    turnLeft();
    move(); // η εντολή επαναλαμβάνεται 40 φορές
}
```

Το παραπάνω πρόβλημα έγκειται στο γεγονός ότι οι λύσεις μας πρέπει να μεταφράζονται στις στοιχειώδεις, βασικές εντολές του ρομπότ *Karel*. Προκειμένου να λυθεί το πρόβλημα αυτό, η γλώσσα προγραμματισμού του ρομπότ *Karel* μας δίνει τη δυνατότητα να δημιουργούμε νέες εντολές που προσθέτουν νέες δυνατότητες στο ρομπότ *Karel*. Ο *Karel*

www.e-diktyo.eu

www.epyna.gr

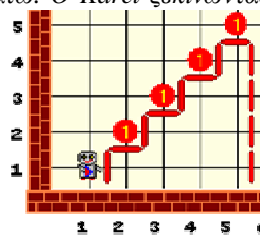
εφοδιάζεται με ένα λεξικό με τα ονόματα και τους ορισμούς των βασικών εντολών και των εντολών που ορίζουμε εμείς οι ίδιοι. Ο ορισμός μιας νέας εντολής αποτελείται από μια σειρά απλούστερων εντολών, ήδη γνωστών στα ρομπότ. Για το παράδειγμά μας μπορούμε να ορίσουμε μια εντολή `moveK1m()` η οποία θα καλεί 8 φορές την εντολή `move()`. Έτσι όταν δίνεται στο ρομπότ Karel η εντολή `moveK1m()` το ρομπότ βρίσκει τον ορισμό που αντιστοιχεί σ' αυτή την εντολή και την εκτελεί, εκτελεί δηλαδή 8 φορές την εντολή `move()`. Χρησιμοποιώντας αυτή τη μέθοδο το πρόγραμμά μας θα αποτελείται από 23 εντολές.

Στα πλαίσια της δραστηριότητας της παρούσας ενότητας παρουσιάζεται στους μαθητές το πρόβλημα σκουπίσματος της σκάλας που παρατίθεται στη συνέχεια. Αρχικά, οι μαθητές εκτελούν βηματικά ένα πρόγραμμα που χρησιμοποιεί μόνο τις τέσσερις βασικές εντολές και στη συνέχεια ένα πρόγραμμα όπου δηλώνονται δύο νέες εντολές: η `turnRight()` για τη στροφή κατά 90 μοίρες δεξιά και η `climbStair()` για το ανέβασμα στο επόμενο σκαλοπάτι και το σκούπισμά του.

Στον κόσμο του Karel υπάρχει η σκάλα που φαίνεται παρακάτω. Ο Karel ξεκινώντας από τη διασταύρωση (1,1) με κατεύθυνση προς τα ανατολικά πρέπει να ανέβει τη σκάλα συγκεντρώνοντας τα αντικείμενα (beeper) που βρίσκονται σε κάθε σκαλοπάτι.

Σύγκρινε τα δύο προγράμματα - «Λύση χωρίς τη Δημιουργία Νέας Εντολής» και «Λύση με τη Δημιουργία Νέας Εντολής» - τα οποία λύνουν το ίδιο πρόβλημα σωστά.

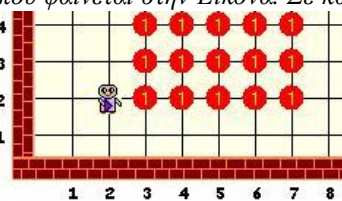
Ποιο από τα δύο θεωρείς «καλύτερο» και γιατί; Αν η σκάλα είχε δέκα σκαλοπάτια ποιόν από τους δύο τρόπους λύσης θα επέλεγες;



4^η Διδακτική Ώρα: ανάπτυξη προγραμμάτων με δημιουργία νέων εντολών

Έχοντας παρουσιάσει την προηγούμενη διδακτική ώρα τη δυνατότητα δημιουργίας νέων εντολών, οι μαθητές αναπτύσσουν προγράμματα στα οποία ορίζουν νέες εντολές. Αρχικά, οι μαθητές καλούνται να μελετήσουν το πρόγραμμα που αναπτύχθηκε τη 2^η διδακτική ώρα για το πρόβλημα της αναρρίχησης στο βουνό και τοποθέτησης της σημαίας. Οι μαθητές συζητάνε και προτείνουν τρόπους βελτίωσης του υπάρχοντος προγράμματος ορίζοντας νέες εντολές. Συμπερασματικά, προτείνεται ο ορισμός τριών νέων εντολών τις οποίες υλοποιούν οι μαθητές: στροφή δεξιά κατά 90 μοίρες, ανέβασμα στο επόμενο επίπεδο του βουνού και κατέβασμα στο επόμενο επίπεδο του βουνού. Οι μαθητές σχολιάζουν τα πλεονεκτήματα και μειονεκτήματα των δύο προγραμμάτων που αναπτύχθηκαν για το ίδιο πρόβλημα. Στη συνέχεια, ανατίθενται στους μαθητές ασκήσεις για το σπίτι, η επίλυση των οποίων ξεκινάει στο εργαστήριο. Παρουσιάζεται ενδεικτικά μία από αυτές.

Ο Karel πρέπει να θερίσει το χωράφι με τα beepers που φαίνεται στην Εικόνα. Σε κάθε διασταύρωση υπάρχει ένα beeper. Προγραμματίσε τον Karel να φέρει εις πέρας την αποστολή του θερίσματος. Επειδή το χωράφι αποτελείται από οριζόντιες γραμμές beeper ίδιου μήκους και το ρομπότ χρειάζεται να εκτελέσει τις ίδιες ενέργειες για το θέρισμα της κάθε γραμμής μπορείς να ορίσεις μια νέα



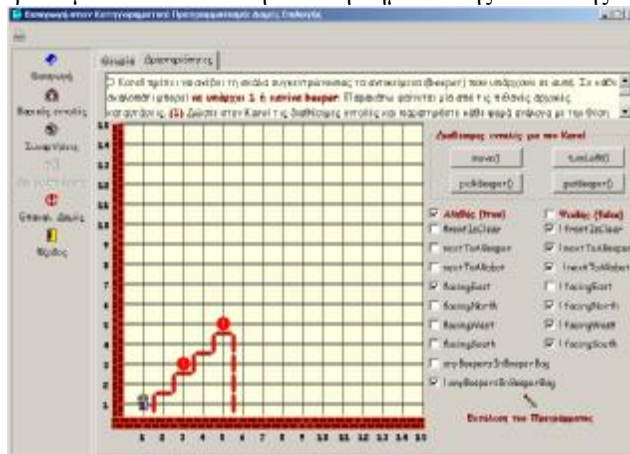
www.e-diktyo.eu

www.epyna.gr

εντολή, έστω `harvestOneRow()`. Την εντολή αυτή θα την καλέσεις στη συνέχεια τρεις φορές από το κυρίως πρόγραμμα (task) για το θέρισμα όλου του χωραφιού.

5^η Διδακτική Ωρα: παρουσίαση των δομών επιλογής

Για την παρουσίαση των δομών επιλογής χρησιμοποιείται η δραστηριότητα της ενότητας «Δομές Επιλογής», στην οποία παρουσιάζεται μια γενικότερη μορφή του γνωστού προβλήματος του σκουπίσματος της σκάλας (Εικόνα 3). Σε αυτή την περίπτωση σε κάθε σκαλοπάτι μπορεί να υπάρχει ένα ή κανένα σκουπίδι (beeper). Μετά από συζήτηση με τους μαθητές προκύπτει η ανάγκη ύπαρξης της δυνατότητας ελέγχου από το ρομπότ της κατάστασης του και του κόσμου όπου κινείται. Αρχικά, παρουσιάζονται στους μαθητές οι συνθήκες που μπορεί να ελέγξει ο Karel και η δομή επιλογής `if`. Οι μαθητές έχουν τη δυνατότητα πατώντας τα κουμπιά των τεσσάρων βασικών εντολών να παρακολουθήσουν το ρομπότ να εκτελεί τις εντολές, καθώς επίσης και να δουν ποιες συνθήκες αληθεύουν και ποιες όχι κάθε χρονική στιγμή. Τέλος, οι μαθητές μελετούν τη νέα έκδοση του προγράμματος και το εκτελούν βηματικά για διάφορες περιπτώσεις (σκάλα χωρίς κανένα σκουπίδι, σκάλα με ένα σκουπίδι σε κάθε σκαλοπάτι, σκάλα με σκουπίδι σε τυχαίες θέσεις) προκειμένου να κατανοήσουν τη σημασία της εκτέλεσης υπό συνθήκη.



Εικόνα 3: Η δραστηριότητα της ενότητας «Δομές Επιλογής».

6^η Διδακτική Ωρα: ανάπτυξη προγραμμάτων με χρήση δομών επιλογής

Οι μαθητές αναπτύσσουν προγράμματα χρησιμοποιώντας τις δομές επιλογής `if` και `if/else`.

7^η Διδακτική Ωρα: παρουσίαση των δομών επανάληψης

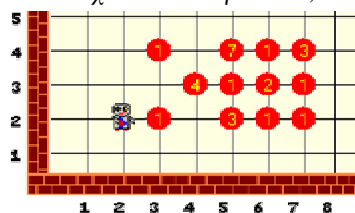
Οι δομές επανάληψης παρουσιάζονται χρησιμοποιώντας τη δραστηριότητα της ενότητας «Δομές Επανάληψης». Στα πλαίσια της δραστηριότητας αυτής παρουσιάζεται η πιο γενική μορφή του γνωστού προβλήματος του σκουπίσματος της σκάλας των δύο προηγούμενων ενότητων: σε κάθε σκαλοπάτι υπάρχει άγνωστος αριθμός σκουπιδιών (beeper). Όπως και στην προηγούμενη ενότητα γίνεται συζήτηση με τους μαθητές και προκύπτει η ανάγκη ύπαρξης της δυνατότητας ελέγχου από το ρομπότ της κατάστασης του και του κόσμου όπου κινείται κατ' επανάληψη. Οι μαθητές καταλήγουν εύκολα στο συμπέρασμα ότι ο Karel θα πρέπει να έχει τη δυνατότητα να ελέγχει αν υπάρχει κάποιο

beeper στη διασταύρωση του και στη συνέχεια να το σηκώνει, όχι όμως μία μόνο φορά όπως με τη δομή επιλογής `if`, αλλά επαναλαμβανόμενα μέχρι να σηκώσει όλα τα beeper. Σε αυτό το σημείο παρουσιάζεται η δομή επανάληψης υπό συνθήκη (`while`) και η σημασία της. Στη συνέχεια, οι μαθητές καλούνται να σκεφτούν τι θα γίνει σε μια σκάλα με 50 σκαλοπάτια και εύκολα διατυπώνουν την άποψη ότι σίγουρα θα υπάρχει μια εντολή που να «λέει» στον Karel πόσες φορές θέλουμε να εκτελέσει κάποιες ενέργειες! Η εντολή αυτή όντως υπάρχει (`loop`) και παρουσιάζεται στους μαθητές. Η λύση για το πρόβλημα με το σκούπισμα της σκάλας παρουσιάζεται στους μαθητές, οι οποίοι καλούνται να την ελέγξουν για οποιαδήποτε αρχική κατάσταση.

8^η Διδακτική Ώρα: ανάπτυξη προγραμμάτων με χρήση δομών επιλογής και επανάληψης

Οι μαθητές αναπτύσσουν προγράμματα χρησιμοποιώντας τις δομές ελέγχου `if`, `while` και `loop`. Προκειμένου οι μαθητές να κατανοήσουν την αξία των δομών ελέγχου καλούνται να αναπτύξουν προγράμματα για την επίλυση της γενικότερης μορφής προβλημάτων που είχαν ανατεθεί σε προηγούμενα μαθήματα. Ως παράδειγμα, παρατίθεται στη συνέχεια η γενικότερη μορφή του προβλήματος θερίσματος του χωραφιού.

Ο Karel πρέπει να θερίσει ένα χωράφι με beepers που έχει διάσταση 5×3 , όπως φαίνεται στη διπλανή Εικόνα. Σε κάθε διασταύρωση υπάρχει άγνωστος αριθμός beeper (κανένα, ένα ή περισσότερα). Προγραμματίστε τον Karel να φέρει εις πέρας την αποστολή του θερίσματος. Στην Εικόνα φαίνεται μία πιθανή αρχική κατάσταση. Επειδή το χωράφι αποτελείται από οριζόντιες γραμμές beeper ίδιου μήκους και το ρομπότ χρειάζεται να εκτελέσει τις ίδιες ενέργειες για το θέρισμα της κάθε γραμμής μπορείς να ορίσεις μια νέα εντολή, έστω `harvestOneRow()`. Την εντολή αυτή θα την καλέσεις στη συνέχεια τρεις φορές από το κυρίως πρόγραμμα (`task`) για το θέρισμα όλου του χωραφιού.



ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία παρουσιάστηκε μία πρόταση διδασκαλίας των βασικών αρχών του προγραμματισμού σε μαθητές Γυμνασίου, η οποία βασίζεται στον μικρόκοσμο προγραμματισμού Karel και την ενσωματωμένη στο περιβάλλον σειρά μαθημάτων και δραστηριοτήτων. Η προτεινόμενη σειρά μαθημάτων εφαρμόζεται την τρέχουσα σχολική χρονιά για τη διδασκαλία των βασικών αρχών του προγραμματισμού στη Γ' τάξη του 2^{ου} Πειραματικού Γυμνασίου Θεσσαλονίκης. Βασικός στόχος της διδασκαλίας αυτής είναι η πιλοτική εφαρμογή του μικρόκοσμου Karel και η αξιολόγηση της προτεινόμενης σειράς μαθημάτων σε πραγματικές συνθήκες. Τα πρώτα αποτελέσματα από τη συγκεκριμένη διδασκαλία, η οποία ολοκληρώνεται αυτό το διάστημα, είναι τα εξής:

- Η προτεινόμενη σειρά μαθημάτων μπορεί να εφαρμοστεί στην τάξη με ιδιαίτερα θετικά αποτελέσματα: η πλειονότητα των μαθητών κατανοεί εύκολα τις διδασκόμενες έννοιες και είναι σε θέση να αναπτύξει προγράμματα για προβλήματα όπως αυτά που παρουσιάστηκαν στην εργασία.
- Ο μικρόκοσμος και τα προβλήματα που επιλύονται προκαλούν το ενδιαφέρον μεγάλης μερίδας των μαθητών. Ορισμένοι μαθητές μάλιστα μαζί με τις ασκήσεις

που ανατέθηκαν για το σπίτι παρέδωσαν και εκφωνήσεις και λύσεις ασκήσεων που επινόησαν οι ίδιοι και παρουσίαζαν μεγαλύτερη πολυπλοκότητα (από εκείνες που τους ανατέθηκαν).

- Η φύση των προβλημάτων επιτρέπει την ανάπτυξη και τον έλεγχο των προγραμμάτων στο χαρτί, στην περίπτωση που δεν υπάρχει διαθέσιμος υπολογιστής στο σπίτι. Αρκετοί μαθητές ανέπτυξαν προγράμματα και τα «ανίχνευσαν» στο χαρτί σημειώνοντας πάνω στην εικόνα της εκφώνησης (προσομοιώνοντας στην ουσία τη λειτουργία της βηματικής εκτέλεσης που χρησιμοποιήσαν στο εργαστήριο).
- Η προσέγγιση που ακολουθείται στα μαθήματα για την παρουσίαση νέων εννοιών στους μαθητές φαίνεται να έχει ιδιαίτερα θετικά αποτελέσματα. Συγκεκριμένα, η παρουσίαση στους μαθητές προβλημάτων που δεν μπορούν να επιλυθούν με τις ήδη γνωστές έννοιες τους οδηγεί σε αδιέξοδο, καθιστώντας έτσι στη συνέχεια προφανή τη σημασία και την αξία των νέων προγραμματιστικών εννοιών/δομών που τους παρουσιάζονται.
- Η χρήση του μικρόκοσμου στην τάξη ανέδειξε σφάλματα στη λειτουργία του περιβάλλοντος, τα οποία δεν είχαν εντοπιστεί στη φάση ελέγχου του λογισμικού. Μετά τις διορθώσεις που έγιναν, οι μαθητές χρησιμοποιούν με ευκολία και χωρίς προβλήματα το περιβάλλον.
- Τα χαρακτηριστικά του μικρόκοσμου προγραμματισμού Karel, και κυρίως η ενσωμάτωση των δραστηριοτήτων στο περιβάλλον, η χρήση του συντάκτη δομής για την ανάπτυξη προγραμμάτων και η βηματική εκτέλεση παρέχουν μεγάλη στήριξη στους μαθητές και καθιστούν δυνατή την εφαρμογή των μαθημάτων στο μικρό χρονικό διάστημα των 8 διδακτικών ωρών.

Η αξιολόγηση των γνώσεων που απέκτησαν οι μαθητές, η οποία θα ακολουθήσει μετά την ολοκλήρωση των μαθημάτων, πιστεύουμε ότι θα επιβεβαιώσει τα ιδιαίτερα θετικά αποτελέσματα που καταγράφηκαν κατά τη διάρκεια των μαθημάτων. Στόχος μας είναι η αξιολόγηση του μικρόκοσμου και της προτεινόμενης σειράς μαθημάτων, καθώς επίσης και η εφαρμογή τους και την επόμενη σχολική χρονιά προκειμένου να εξασχθούν ασφαλή συμπεράσματα για την παιδαγωγική τους αξία.

BIBΛΙΟΓΡΑΦΙΑ

1. Brusilovsky, P., Calabrese, E., Hvorecky, J., Kouchnirenko, A., & Miller, P. (1997) Mini-languages: a way to learn programming principles. *International Journal of Education and Information Technologies*, 2, Kluwer Academic Publishers, 65-83.
2. Pattis, R. E., Roberts, J. & Stehlik, M. (1995) Karel - The Robot, A Gentle Introduction to the Art of Programming. 2nd edn. Wiley, New York.
3. Γόγουλου, Α., Γουλή, Ε. Και Γρηγοριάδου, Μ. (2008) Αξιοποίηση του e-ECLIP στη διδασκαλία βασικών προγραμματιστικών δομών, *Πρακτικά 4^ο Πανελληνίου Συνεδρίου Διδακτικής της Πληροφορικής*, Πάτρα 28-30 Μαρτίου, 35-44.
4. Καγκάνη, Κ., Δαγδιλέλης, Β., Σατρατζέμη, Μ. και Ευαγγελίδης, Γ. (2005) Μια Μελέτη Περίπτωσης της Διδασκαλίας του Προγραμματισμού στη Δευτεροβάθμια Εκπαίδευση με τα LEGO Mindstorms, *Πρακτικά 3^ο Πανελληνίου Συνεδρίου Διδακτικής της Πληροφορικής*, Κόρινθος 7- 9 Οκτωβρίου, 212-220.
5. Μπέλλου, Ι. και Μικρόπουλος, Τ. (2005) Μία Εναλλακτική Πρόταση για την Εισαγωγή στον Προγραμματισμό στο Γυμνάσιο, *Πρακτικά 3^ο Πανελληνίου Συνεδρίου Διδακτικής της Πληροφορικής*, Κόρινθος 7- 9 Οκτωβρίου, 157-165.