ePublishing
ΕΘΝΙΚΟ ΚΕΝΤΡΟ ΤΕΚΜΗΡΙΩΣΗΣ NATIONAL DOCUMENTATION CENTRE

Συνέδρια της Ελληνικής Επιστημονικής Ένωσης Τεχνολογιών Πληροφορίας & Επικοινωνιών στην Εκπαίδευση

Τόμ. 1 (2012)

6ο Συνέδριο Διδακτική της Πληροφορικής

Διδακτική Προσέγγιση για τον Αντικειμενοστρεφή Προγραμματισμό: εφαρμογή στο BlueJ

Σ. Ξυνόγαλος

Βιβλιογραφική αναφορά:

Ξυνόγαλος Σ. (2022). Διδακτική Προσέγγιση για τον Αντικειμενοστρεφή Προγραμματισμό: εφαρμογή στο BlueJ. *Συνέδρια της Ελληνικής Επιστημονικής Ένωσης Τεχνολογιών Πληροφορίας & Επικοινωνιών στην Εκπαίδευση*, *1*, 063–072. ανακτήθηκε από https://eproceedings.epublishing.ekt.gr/index.php/cetpe/article/view/4425

https://epublishing.ekt.gr | **e-Εκδότης:** EKT | **Πρόσβαση:** 20/01/2026 07:04:20

Διδακτική Προσέγγιση για τον Αντικειμενοστρεφή Προγραμματισμό: εφαρμογή στο BlueJ

Σ. Ξυνόγαλος

Τμήμα Διοίκησης Τεχνολογίας, Πανεπιστήμιο Μακεδονίας
stelios@uom.gr

Περίληψη

Στην παρούσα εργασία παρουσιάζεται μια διδακτική προσέγγιση για τον αντικειμενοστρεφή προγραμματισμό, η οποία βασίζεται στα αποτελέσματα των μελετών που έχουν διεξαχθεί για τη διδασκαλία και εκμάθησή του. Συγκεκριμένα, προσδιορίζονται οι βασικότερες έννοιες στις οποίες πρέπει να επικεντρώνεται η διδασκαλία του αντικειμενοστρεφούς προγραμματισμού, συνοψίζονται οι σημαντικότερες δυσκολίες και παρανοήσεις που έχουν καταγραφεί στη βιβλιογραφία και προτείνονται συγκεκριμένες διδακτικές παρεμβάσεις για μια πιο επιτυχημένη διδασκαλία. Τέλος, παρουσιάζονται ενδεικτικές δραστηριότητες για το περιβάλλον BlueJ που βασίζονται στη συγκεκριμένη διδακτική προσέγγιση.
Λέξεις κλειδιά: *αντικειμενοστρεφής προγραμματισμός, διδακτική προσέγγιση, δραστηριότητες.*

Abstract

This paper presents a teaching approach for object oriented programming (OOP), which is based on the results of the research carried out regarding its teaching and learning. Specifically, the most fundamental concepts that the teaching of OOP should focus are defined, the most important difficulties and misconceptions recorded in the literature are reviewed and specific didactical interventions are proposed. Finally, indicative activities based on the proposed teaching approach for the environment BlueJ are presented.
Keywords: *object oriented programming, teaching approach, activities.*

1. Εισαγωγή

Η διδασκαλία του αντικειμενοστρεφούς προγραμματισμού, όπως προκύπτει από τη διεθνή βιβλιογραφία, συνοδεύεται από ποικίλες δυσκολίες. Στην πλειονότητα των περιπτώσεων οι δυσκολίες αυτές οφείλονται σε παρανοήσεις των σπουδαστών για τις βασικές αντικειμενοστρεφείς έννοιες (Holland et al., 1997; Sanders & Thomas, 2007; Thomasson et al., 2006). Σε αρκετές περιπτώσεις, οι σχετικές παρανοήσεις είναι απόρροια μιας διδασκαλίας που επικεντρώνεται στην παρουσίαση των βασικών προγραμματιστικών δομών (δομή ακολουθίας, επιλογής και επανάληψης) και όχι στην ανάδειξη των αρχών της συγκεκριμένης τεχνικής προγραμματισμού. Το αποτέλεσμα είναι να μην γίνονται κατανοητές οι βασικές αντικειμενοστρεφείς έννοιες και στην ουσία οι σπουδαστές να αναπτύσσουν προγράμματα που βασίζονται στην τεχνική του δομημένου προγραμματισμού χρησιμοποιώντας μια αντικειμενοστρεφή γλώσσα. Στην Ελλάδα ο αντικειμενοστρεφής προγραμματισμός

διδάσκεται στην Τριτοβάθμια εκπαίδευση. Ωστόσο, η διδασκαλία του είναι πλέον δυνατή και στη Δευτεροβάθμια εκπαίδευση, μιας και υπάρχουν διαθέσιμα εκπαιδευτικά εργαλεία/περιβάλλοντα και συσσωρευμένη διδακτική εμπειρία, όπως προκύπτει από τη διεθνή βιβλιογραφία (Ragonis & Ben-Ari, 2005).

Στην παρούσα εργασία, παρουσιάζονται οι έννοιες στις οποίες πρέπει να επικεντρώνεται η διδασκαλία του αντικειμενοστρεφούς προγραμματισμού και οι σχετικές δυσκολίες και παρανοήσεις που έχουν καταγραφεί στη διεθνή βιβλιογραφία. Επίσης, προτείνονται γενικές διδακτικές παρεμβάσεις για την αντιμετώπιση των πιο σημαντικών δυσκολιών και παρανοήσεων, καθώς και συγκεκριμένες δραστηριότητες για το περιβάλλον BlueJ.

2. Βασικές έννοιες

Οι δύο βασικότερες έννοιες του αντικειμενοστρεφούς προγραμματισμού είναι η έννοια της κλάσης και του αντικειμένου, οι οποίες μάλιστα σύμφωνα με τη διεθνή βιβλιογραφία πολλές φορές συγχέονται (Holland et al., 1997; Sanders & Thomas, 2007). Η διδασκαλία της συγκεκριμένης τεχνικής προγραμματισμού πρέπει να επικεντρωθεί στην παρουσίαση των παρακάτω εννοιών/αρχών:

- Η αναπαράσταση των οντοτήτων/τμημάτων ενός συστήματος του κόσμου για το οποίο δημιουργούμε ένα μοντέλο στον υπολογιστή γίνεται με τα **αντικείμενα**.
- Τα αντικείμενα ανάλογα με τις ιδιότητες που τα χαρακτηρίζουν και τις λειτουργίες που μπορούν να εκτελούν κατηγοριοποιούνται και περιγράφονται με ένα γενικό τρόπο σε μια **κλάση**. Για κάθε διαφορετικό είδος αντικειμένων πρέπει να ορίσουμε μία ξεχωριστή κλάση, η οποία περιγράφει:
    - τις *ιδιότητες* που χαρακτηρίζουν το συγκεκριμένο είδος αντικειμένων ή αλλιώς τις τιμές δεδομένων που χαρακτηρίζουν την *κατάσταση* τους. Οι τιμές των ιδιοτήτων αυτών αποθηκεύονται σε ειδικές μεταβλητές που δηλώνονται στην κλάση και ονομάζονται **πεδία**.
    - τις *λειτουργίες* που καθορίζουν τη *συμπεριφορά* ενός είδους αντικειμένων και προσδιορίζουν, μεταξύ άλλων, τον τρόπο επεξεργασίας των δεδομένων που περικλείει το κάθε αντικείμενο. Η περιγραφή του τρόπου εκτέλεσης μιας συγκεκριμένης λειτουργίας ονομάζεται **μέθοδος**.
- Από τη στιγμή που θα ορίσουμε μια κλάση *μπορούμε να δημιουργήσουμε όσα αντικείμενα, ή αλλιώς* **στιγμιότυπα***, χρειαζόμαστε* για την επίλυση ενός δεδομένου προβλήματος. Η κλάση αποτελεί το «καλούπι» βάσει του οποίου κατασκευάζονται τα αντικείμενα.
- Όταν δημιουργούμε ένα νέο αντικείμενο του δίνουμε ένα *όνομα* προκειμένου να μπορούμε στη συνέχεια να αναφερθούμε σε αυτό μέσω του προγράμματός μας και να του στείλουμε μηνύματα, και επίσης *αρχικοποιούμε* τα πεδία του καθορίζοντας έτσι την αρχική του κατάσταση. Η δημιουργία ενός νέου

αντικειμένου γίνεται καλώντας μια ειδική μέθοδο που ορίζεται στην κλάση του αντικειμένου και ονομάζεται **κατασκευαστής**.

Βασική έννοια στον αντικειμενοστρεφή προγραμματισμό αποτελεί και η **κληρονομικότητα**, ο ορισμός δηλαδή μιας κλάσης ως επέκταση κάποιας άλλης. Η κληρονομικότητα μπορεί εύκολα να διδαχθεί σε ειδικά σχεδιασμένα περιβάλλοντα, όπως για παράδειγμα σε ένα προγραμματιστικό μικρόκοσμο, αλλά η διδασκαλία σε μια συμβατική αντικειμενοστρεφή γλώσσα είναι σημαντικά πιο δύσκολη.

3. Δυσκολίες, παρανοήσεις και διδακτικές παρεμβάσεις

Η διδασκαλία και εκμάθηση του αντικειμενοστρεφούς προγραμματισμού συνοδεύεται από ποικίλες δυσκολίες και παρανοήσεις. Όταν ο διδάσκοντας έχει γνώση των σχετικών δυσκολιών και παρανοήσεων είναι σε θέση να σχεδιάσει και να αναπτύξει κατάλληλο εκπαιδευτικό υλικό προκειμένου να αποφευχθεί στο βαθμό που αυτό είναι δυνατό η εμφάνιση συγκεκριμένων δυσκολιών/παρανοήσεων, ή τουλάχιστον να αντιμετωπιστούν οι δυσκολίες αυτές ευκολότερα. Όσον αφορά στις παρανοήσεις που έχουν καταγραφεί ιδιαίτερη βαρύτητα πρέπει να δοθεί σε αυτές που συνοψίζονται στις ενότητες που ακολουθούν. Πρόκειται για παρανοήσεις που είναι συχνές και αντιπροσωπεύουν σημαντικά εμπόδια στην κατανόηση των βασικών εννοιών του αντικειμενοστρεφούς προγραμματισμού. Οι διδακτικές παρεμβάσεις που προτείνονται βασίζονται τόσο στη βιβλιογραφία, όσο και στην πολυετή εμπειρία του συγγραφέα στη διδασκαλία και αξιολόγηση του μαθήματος «Αντικειμενοστρεφής Σχεδίαση και Προγραμματισμός» του Τμήματος Διοίκησης Τεχνολογίας του Πανεπιστημίου Μακεδονίας (Xinogalos & Satratzemi, 2009).

3.1 Οι έννοιες της κλάσης και του αντικειμένου

Δυσκολίες και παρανοήσεις

(**A1**) *Αρκετοί σπουδαστές συγχέουν τις έννοιες της κλάσης και του αντικειμένου και δεν μπορούν να τις διαχωρίσουν* (Holland et al., 1997; Sanders & Thomas, 2007).

(**A2**) Μια κλάση είναι μια συλλογή αντικειμένων και όχι ένα πρότυπο δημιουργίας αντικειμένων (Ragonis & Ben-Ari, 2005; Sanders & Thomas, 2007; Thomasson et al., 2006).

(**A3**) Δυσκολία κατανόησης της στατικής φύσης μιας κλάσης και της δυναμικής φύσης ενός αντικειμένου (Ragonis & Ben-Ari, 2005).

Διδακτικές παρεμβάσεις

Η σχεδίαση της λύσης σε ένα πρόβλημα με την αντικειμενοστρεφή τεχνική προγραμματισμού προϋποθέτει: (1) τον *εντοπισμό των οντοτήτων* (βασικών στοιχείων) του προβλήματος, (2) την *μοντελοποίηση κάθε οντότητας*, τον εντοπισμό δηλαδή των ιδιοτήτων (πεδίων) και των λειτουργιών (μεθόδων) της με απώτερο στόχο τον ορισμό μιας κλάσης που την αναπαριστά, και (3) τη *δημιουργία των*

απαραίτητων αντικειμένων από κάθε κλάση και την αποστολή των κατάλληλων μηνυμάτων (εκτέλεση των κατάλληλων μεθόδων) για την επίλυση του προβλήματος.

Αν ο σπουδαστής δεν κατανοήσει ότι μια κλάση περιγράφει με αφηρημένο τρόπο ένα είδος αντικειμένων, καθώς επίσης και το γεγονός ότι μπορούμε στα πλαίσια επίλυσης ενός προβλήματος να δημιουργήσουμε οποιοδήποτε αριθμό αντικειμένων (στιγμιοτύπων) από μια κλάση τότε η δυσκολία σχεδίασης μιας αντικειμενοστρεφούς λύσης σε ένα πρόβλημα είναι πολύ μεγάλη.

Ο εντοπισμός της συγκεκριμένης παρανόησης από τον διδάσκοντα μπορεί να γίνει εύκολα απευθύνοντας σχετικές ερωτήσεις ανοικτού ή κλειστού τύπου στους σπουδαστές ή δίνοντας τον προσδιορισμό ενός απλού προβλήματος και ζητώντας από τους σπουδαστές να καθορίσουν τις απαιτούμενες κλάσεις και αντικείμενα. Επίσης, ο ορισμός από τους σπουδαστές ενός αντιγράφου (ή περισσότερων αντιγράφων) μιας κλάσης με διαφορετικό όνομα και η δημιουργία ενός αντικειμένου από κάθε μία κλάση για την επίλυση ενός προβλήματος, αντί για τον ορισμό μιας κλάσης και την δημιουργία του κατάλληλου αριθμού αντικειμένων από αυτή, είναι μια συχνή απόρροια της παρανόησης ότι κλάση και αντικείμενο είναι έννοιες ταυτόσημες. Για την αποφυγή της παρανόησης ότι οι έννοιες κλάση και αντικείμενο είναι έννοιες ταυτόσημες, πρέπει να χρησιμοποιούνται από τα πρώτα μαθήματα παραδείγματα και ασκήσεις, στις οποίες δημιουργούνται περισσότερα από ένα αντικείμενα από κάθε κλάση.

3.2 Δημιουργία αντικειμένων

Δυσκολίες και παρανοήσεις (Ragonis & Ben-Ari, 2005)

(**B1**) *Δυσκολία κατανόησης της διαδικασίας δημιουργίας ενός αντικειμένου.*
(**B2**) *Τα αντικείμενα δημιουργούνται από μόνα τους.*
(**B3**) *Η χρήση του κατασκευαστή είναι προαιρετική.* Η δημιουργία ενός αντικειμένου μπορεί να επιτευχθεί και με μια μέθοδο, έστω set_values, που δίνει αρχικές τιμές στις μεταβλητές στιγμιοτύπου του νέου αντικειμένου (Fleury, 2000).
(**B4**) *Η χρήση πολλαπλών κατασκευαστών προκαλεί σύγχυση* (Carter &Fowler, 1998). Αρκετοί *δυσκολεύονται να ορίσουν δύο ή και περισσότερους κατασκευαστές στην ίδια κλάση και ορίζουν ένα μόνο κατασκευαστή ή δεν δίνουν το ίδιο όνομα στους κατασκευαστές* όπως απαιτείται (Ξυνόγαλος, 2008).
(**B5**) Οι κατασκευαστές μπορούν να περιλαμβάνουν μόνο εντολές ανάθεσης για την αρχικοποίηση των πεδίων.
(**B6**) Η δημιουργία ενός στιγμιοτύπου (αντικειμένου) μιας κλάσης αφορά μόνο στην εκτέλεση του κατασκευαστή και όχι στην κατανομή μνήμης.
(**B7**) Δυσκολία κατανόησης ενός κενού κατασκευαστή.
(**B8**) Η αρχικοποίηση ενός πεδίου με μια σταθερά κατά τη δήλωσή του προκαλεί δυσκολία στο διαχωρισμό μεταξύ της κλάσης και του αντικειμένου.

(B9) Αν τα πεδία αρχικοποιούνται στη δήλωση της κλάσης τότε δεν υπάρχει ανάγκη να δημιουργηθούν αντικείμενα.

Διδακτικές παρεμβάσεις

Αφού οι σπουδαστές κατανοήσουν ότι μια κλάση αποτελεί το «καλούπι» για τη δημιουργία αντικειμένων, πρέπει να τους παρουσιαστεί ο τρόπος με τον οποίο επιτυγχάνεται η δημιουργία τους (B1). Όταν δημιουργηθεί ένα «καλούπι» (κλάση), αυτό παραμένει διαθέσιμο και μπορεί να χρησιμοποιηθεί σε διαφορετικές περιπτώσεις (προβλήματα) για τη δημιουργία του απαραίτητου κάθε φορά αριθμού αντικειμένων και την εκτέλεση από αυτά των κατάλληλων ενεργειών για την επίλυση του προβλήματος (A3). Είναι σημαντικό να κατανοήσουν οι σπουδαστές ότι η ύπαρξη μιας κλάσης δεν συνεπάγεται την ύπαρξη ενός ή περισσότερων αντικειμένων (B2). Για τη δημιουργία ενός αντικειμένου απαιτείται η κλήση της ειδικής μεθόδου που υπάρχει σε κάθε κλάση, έχει το ίδιο όνομα με αυτή και ονομάζεται κατασκευαστής. Πρέπει να γίνει σαφές ότι ο κατασκευαστής δεσμεύει την απαραίτητη ποσότητα μνήμης για την αποθήκευση των τιμών των ιδιοτήτων του αντικειμένου, οι οποίες καθορίζονται στην κλάση, και την ανάθεση των αρχικών τιμών τους (B6). Ο κατασκευαστής είναι η μοναδική μέθοδος μιας κλάσης που μπορεί να εκτελεστεί χωρίς προηγουμένως να έχει δημιουργηθεί κάποιο αντικείμενο από αυτή. Όλες οι υπόλοιπες μέθοδοι εκτελούνται από τα αντικείμενα που έχουν δημιουργηθεί. Συνεπώς, η χρήση του κατασκευαστή για τη δημιουργία αντικειμένων είναι απαραίτητη (B3). Μια μέθοδος που δίνει τιμή σε ένα ή περισσότερα πεδία ενός αντικειμένου μπορεί να κληθεί μετά τη δημιουργία του για να ενημερώσει την κατάστασή του, όχι όμως για τη δημιουργία του (B3).

Επίσης, πρέπει να αποφεύγεται η χρήση παραδειγμάτων όπου τα πεδία μιας κλάσης αρχικοποιούνται κατά τη δήλωσή τους και όχι στα πλαίσια του κατασκευαστή. Ακόμα και στην περίπτωση που κάποια από τα πεδία αρχικοποιούνται σε μια σταθερή τιμή, αυτό πρέπει να γίνεται στα πλαίσια του κατασκευαστή τόσο γιατί αυτό είναι και εννοιολογικά προτιμότερο όσο και για την αποφυγή των παρανοήσεων που ευνοεί (A1, B8, B9).

Τέλος, πρέπει να χρησιμοποιηθεί κάποιο παράδειγμα κλάσης στην οποία θα ορίζονται περισσότεροι από ένας κατασκευαστές, προκειμένου να αντιμετωπιστούν οι σχετικές δυσκολίες (B4). Πρέπει να επισημανθεί ότι στην περίπτωση πολλαπλών κατασκευαστών όλοι έχουν το ίδιο όνομα με την κλάση και η διάκρισή τους γίνεται με βάση τον αριθμό των ορισμάτων που χρησιμοποιούνται κατά την κλήση τους. Ιδιαίτερη βαρύτητα πρέπει να δοθεί στην επιθεώρηση της αρχικής κατάστασης των αντικειμένων που δημιουργούνται με τους διαθέσιμους κατασκευαστές.

3.3 Η έννοια των ιδιοτήτων και της κατάστασης ενός αντικειμένου

Δυσκολίες και παρανοήσεις

Η δυσκολία κατανόησης του ρόλου των πεδίων μιας κλάσης έχει ως συνέπεια τη δημιουργία διάφορων παρανοήσεων:

(**C1**) *Κάθε αντικείμενο αποτελεί απλά ένα «περιτύλιγμα» μιας μεταβλητής* (Holland et al., 1997).

(**C2**) *Δυσκολία κατανόησης της επίδρασης που έχει η εκτέλεση μιας μεθόδου στην κατάσταση ενός αντικειμένου* (Ragonis & Ben-Ari, 2005). Σε αρκετές περιπτώσεις δεν γίνεται αντιληπτό ότι η αρχική κατάσταση ενός αντικειμένου καθορίζεται όταν αυτό δημιουργείται και στη συνέχεια ενημερώνεται με την εκτέλεση μεθόδων από το αντικείμενο.

(**C3**) Δύο αντικείμενα της ίδιας κλάσης δεν μπορούν να έχουν τις ίδιες τιμές πεδίων (Ragonis & Ben-Ari, 2005).

Διδακτικές παρεμβάσεις

Σε αρκετές περιπτώσεις οι διδάσκοντες για λόγους απλότητας παρουσιάζουν παραδείγματα στα οποία χρησιμοποιείται ένα μόνο πεδίο σε κάθε κλάση. Η χρήση τέτοιου είδους παραδειγμάτων είναι πολύ πιθανό να δημιουργήσει την παρανόηση ότι ένα αντικείμενο είναι απλά ένα «περιτύλιγμα» μιας μεταβλητής χωρίς κάποια λειτουργικότητα (C1), και γι' αυτό θα πρέπει να αποφεύγεται (Holland et al., 1997). Επίσης, οι σπουδαστές πρέπει μέσα από ειδικά σχεδιασμένες δραστηριότητες να κληθούν να προσδιορίσουν το αποτέλεσμα της εκτέλεσης διαφόρων μεθόδων στην κατάσταση των αντικειμένων μιας κλάσης (C2), καθώς και να διαπιστώσουν στην πράξη ότι μπορεί τα αντικείμενα μιας κλάσης σε διάφορες φάσεις της εκτέλεσης ενός προγράμματος να έχουν την ίδια κατάσταση (C3).

3.4 Η έννοια των λειτουργιών και της συμπεριφοράς ενός αντικειμένου

Δυσκολίες και παρανοήσεις

(**D1**) *Σε αρκετές περιπτώσεις δεν γίνεται αντιληπτό ότι η συμπεριφορά ενός αντικειμένου μπορεί να αλλάξει ουσιαστικά ανάλογα με την κατάστασή του* (Holland et al., 1997).

(**D2**) *Δεν αξιοποιείται η λειτουργικότητα των μεθόδων* και επαναλαμβάνεται ένα τμήμα κώδικα αντί να ορίζεται μια μέθοδος (Fleury, 2001).

(**D3**) Μπορούμε να ορίσουμε μια μέθοδο που προσθέτει μια ιδιότητα/πεδίο σε μια κλάση (Ragonis & Ben-Ari, 2005).

(**D4**) Δυσκολία κατανόησης της κατηγοριοποίησης των μεθόδων σε κατασκευαστές, μεθόδους πρόσβασης και μετάλλαξης (Ragonis & Ben-Ari, 2005).

(**D5**) Δυσκολία κατανόησης του γεγονότος ότι μια μέθοδος μπορεί να κληθεί για οποιοδήποτε αντικείμενο μιας κλάσης (Ragonis & Ben-Ari, 2005).

(**D6**) Οι μέθοδοι εκτελούνται σύμφωνα με τη σειρά που εμφανίζονται στον ορισμό της κλάσης (Ragonis & Ben-Ari, 2005).

(**D7**) Κάθε μέθοδος μπορεί να κληθεί μόνο μία φορά (Ragonis & Ben-Ari, 2005).

(**D8**) Δυσκολία κατανόησης της κλήσης μιας μεθόδου από άλλη μέθοδο (Ragonis & Ben-Ari, 2005).
(**D9**) Δυσκολία κατανόησης της προέλευσης των τιμών των παραμέτρων σε μια μέθοδο (Ragonis & Ben-Ari, 2005).
(**D10**) Δυσκολία κατανόησης σχετικά με το που καταλήγει η επιστρεφόμενη τιμή μιας μεθόδου (Ragonis & Ben-Ari, 2005).

Διδακτικές παρεμβάσεις

Αρκετές φορές χρησιμοποιούνται παραδείγματα και ασκήσεις στις οποίες τα αντικείμενα συμπεριφέρονται ως απλές εγγραφές δεδομένων χωρίς καμία απολύτως λειτουργικότητα (συμπεριφορά). Για παράδειγμα, χρησιμοποιείται μια κλάση VideoGame που αναπαριστά ηλεκτρονικά παιχνίδια και έχει ως πεδία τον τίτλο του παιχνιδιού, τον δημιουργό κτλ. Παρόλο που το θέμα αυτό προσελκύει (ενδεχομένως) το ενδιαφέρον των σπουδαστών, δεν αναδεικνύει το γεγονός ότι τα αντικείμενα έχουν συμπεριφορά, η οποία μάλιστα μπορεί να αλλάξει ανάλογα με την κατάστασή τους. Τα παραδείγματα που χρησιμοποιούνται και οι ασκήσεις που ανατίθενται στους σπουδαστές πρέπει να αναφέρονται σε αντικείμενα που περικλείουν δεδομένα, αλλά έχουν και συμπεριφορά που μπορεί να αλλάξει ουσιαστικά ανάλογα με την κατάστασή τους (D1) (Holland et al., 1997). Για παράδειγμα, μπορεί να χρησιμοποιηθεί μια κλάση που αναπαριστά ένα Λογαριασμό (ή τον Κουμπαρά) ενός σπουδαστή. Ένα αντικείμενο της κλάσης αυτής θα συμπεριφερθεί διαφορετικά σε μια αίτηση ανάληψης, ανάλογα με τον αν υπάρχει υπόλοιπο ή όχι στο λογαριασμό.

Οι σπουδαστές συχνά επαναλαμβάνουν πανομοιότυπα τμήματα κώδικα σε ένα πρόγραμμα, αντί να ορίζουν μια μέθοδο για την υλοποίηση της συγκεκριμένης λειτουργίας και να την καλούν στα απαραίτητα σημεία. Ειδικά, όταν το συγκεκριμένο τμήμα κώδικα εκτελείται μέσω άλλων μεθόδων, υπάρχουν σπουδαστές που ακόμα και αν αντιληφθούν ότι αυτό το τμήμα κώδικα μπορεί να αποτελέσει μια μέθοδο προτιμούν αντί αυτού να επαναλαμβάνουν το ίδιο τμήμα κώδικα. Όπως έχουν αναφέρει και άλλοι ερευνητές (Ragonis & Ben-Ari, 2005) η κλήση μεθόδων μέσα από άλλες μεθόδους προκαλεί δυσκολίες σε αρκετούς σπουδαστές.

Ο διδάσκοντας θα πρέπει να ετοιμάσει από τα πρώτα μαθήματα κατάλληλες δραστηριότητες προκειμένου να κατανοήσουν οι σπουδαστές και να αξιοποιήσουν τη λειτουργικότητα των μεθόδων (D2). Ενδεικτικά, θα μπορούσε να δοθεί στους σπουδαστές ο ορισμός μιας κλάσης στην οποία επαναλαμβάνεται το ίδιο τμήμα κώδικα και να τεθούν σχετικές ερωτήσεις, να ακολουθήσει συζήτηση, να εντοπιστεί το πρόβλημα και να βελτιωθεί ο ορισμός της κλάσης με την αντικατάσταση των ίδιων τμημάτων κώδικα από μια κλήση σε μια νέα μέθοδο που θα προστεθεί.

4. Δραστηριότητες για το περιβάλλον BlueJ

Χρησιμοποιώντας μια έτοιμη κλάση, οι σπουδαστές μέσω διερεύνησης και πειραματισμού εξοικειώνονται με τις βασικές έννοιες: κλάση, αντικείμενο, δημιουργία αντικειμένου, ιδιότητες, κατάσταση αντικειμένου, λειτουργίες. Η κλάση που θα χρησιμοποιηθεί πρέπει να περιγράφει με απλό τρόπο μια οντότητα γνωστή στους σπουδαστές. Ωστόσο, πρέπει να περιλαμβάνει περισσότερες από μία ιδιότητες (C1), ενώ τα αντικείμενα που αναπαριστά πρέπει να έχουν κάποια συμπεριφορά που επηρεάζεται από την κατάστασή τους (D1).

Οι σπουδαστές **δημιουργούν ένα αντικείμενο** χρησιμοποιώντας τις δυνατότητες *άμεσης διαχείρισης* του BlueJ (κάνοντας δεξί κλικ πάνω στο ορθογώνιο που αναπαριστά την κλάση στο βασικό παράθυρο και επιλέγοντας τη δημιουργία αντικειμένου από το αναδυόμενο μενού). Η δημιουργία αντικειμένου απαιτεί:

- τον καθορισμό του ονόματος του νέου αντικειμένου που θα χρησιμοποιηθεί στη συνέχεια για την αποστολή μηνυμάτων σε αυτό, και
- τον καθορισμό των αρχικών τιμών των ιδιοτήτων του αντικειμένου (η αρχική κατάσταση του αντικειμένου) που δεν έχουν μια σταθερή αρχική τιμή.

Στη συνέχεια, **επιθεωρούν την αρχική κατάσταση του αντικειμένου** χρησιμοποιώντας τη σχετική δυνατότητα του περιβάλλοντος (κάνοντας δεξί κλικ πάνω στο αντικείμενο και επιλέγοντας Inspect από το αναδυόμενο μενού).

Έχοντας το παράθυρο επιθεώρησης της κατάστασης του αντικειμένου στο προσκήνιο οι σπουδαστές στέλνουν σε αυτό μηνύματα, ή αλλιώς **καλούν τις διαθέσιμες μεθόδους** χρησιμοποιώντας τη δυνατότητα άμεσης διαχείρισης του BlueJ (επιλέγοντας τις μεθόδους από το αναδυόμενο μενού που εμφανίζεται κάνοντας δεξί κλικ πάνω στο αντικείμενο και παρέχοντας τις απαραίτητες πληροφορίες μέσω πλαισίων διαλόγου). Οι σπουδαστές καλούνται μέσω ενός φύλλου δραστηριοτήτων να:

- Καλέσουν τις μεθόδους του αντικειμένου με οποιαδήποτε σειρά επιθυμούν (D6) και όσες φορές χρειάζεται (D7) προκειμένου να κατανοήσουν τη λειτουργία τους.
- Παρατηρήσουν, με τη βοήθεια του παραθύρου επιθεώρησης της κατάστασης του αντικειμένου, ποιες μέθοδοι αλλάζουν την κατάστασή του και ποιες όχι (C2, D4) .
- Παρατηρήσουν, αν ο τρόπος ανταπόκρισης του αντικειμένου (συμπεριφορά) σε κάποιες μεθόδους διαφέρει ανάλογα με την κατάστασή του (D1).
- Διαβάζουν προσεκτικά τις πληροφορίες που παρέχονται στα πλαίσια διαλόγου που εμφανίζονται κατά την κλήση μεθόδων και παρέχουν τα κατάλληλα δεδομένα (μέθοδοι με παραμέτρους) προκειμένου να εκτελέσουν τη λειτουργία τους (D9).

Ολοκληρώνοντας τις δραστηριότητες οι σπουδαστές καταγράφουν σε πίνακες τα παρακάτω στοιχεία, χρησιμοποιώντας αποκλειστικά τις πληροφορίες που παρέχονται από το παράθυρο επιθεώρησης κατάστασης αντικειμένου, τα αναδυόμενα μενού και

τα πλαίσια διαλόγου και όχι από τον κώδικα της κλάσης: (1) **κλάση**: όνομα και είδος αντικειμένων που αναπαριστά, (2) **ιδιότητες**: όνομα και ρόλος της κάθε ιδιότητας, (3) **δημιουργία αντικειμένου**: όνομα της λειτουργίας που εκτελείται για τη δημιουργία του αντικειμένου (κατασκευαστής), ιδιότητες που παίρνουν αρχική τιμή από τον χρήστη και ιδιότητες που παίρνουν κάποια προεπιλεγμένη (σταθερή) τιμή, και (4) **λειτουργίες**: όνομα, ρόλος, δεδομένα που απαιτούνται από τον χρήστη για την εκτέλεση της λειτουργίας, τιμή που επιστρέφεται ως αποτέλεσμα της εκτέλεσής της, ιδιότητα/ες που επηρεάζονται από την εκτέλεσή της.

Πρόσθετες δραστηριότητες που μπορεί να χρησιμοποιηθούν για την αποφυγή δημιουργίας συνηθισμένων παρανοήσεων είναι οι εξής:

- **δημιουργία πολλαπλών στιγμιοτύπων από μια κλάση**, προκειμένου να αποσαφηνιστεί η σχέση μεταξύ κλάσης και αντικειμένου (A1) και να διαπιστώσουν οι σπουδαστές ότι για τη δημιουργία κάθε αντικειμένου απαιτείται η εκτέλεση της ίδιας κάθε φορά ειδικής λειτουργίας (κλήση κατασκευαστή) που ορίζεται στην κλάση (B1, B2)
- κατανόηση του γεγονότος ότι μια κλάση αποτελεί το πρότυπο βάσει του οποίου δημιουργούνται αντικείμενα που έχουν όλα τις ίδιες ιδιότητες και τις ίδιες λειτουργίες (A2), χρησιμοποιώντας τη δυνατότητα επιθεώρησης της κατάστασης των αντικειμένων και κλήσης μεθόδων από το αναδυόμενο μενού για κάθε αντικείμενο
- κλήση των λειτουργιών που ορίζονται στην κλάση για κάθε αντικείμενο που έχει δημιουργηθεί (D5, D6, D7) δίνοντας έμφαση στη στατική φύση της κλάσης και τη δυναμική φύση των αντικειμένων (A3)
- κλήση μεθόδων με ταυτόχρονη παρακολούθηση της κατάστασής τους και διαπίστωση ότι είναι πιθανό διαφορετικά αντικείμενα να έχουν κάποια στιγμή την ίδια ακριβώς κατάσταση (C3)

Μετά από την εκτέλεση δραστηριοτήτων για την κατανόηση των βασικών εννοιών μέσω διερεύνησης μιας έτοιμης κλάσης, οι σπουδαστές καλούνται να λύσουν ένα **συγκεκριμένο πρόβλημα**, το οποίο απαιτεί τη δημιουργία κάποιων στιγμιοτύπων (τουλάχιστον δύο) και την κλήση κάποιων μεθόδων. Οι σπουδαστές αξιοποιώντας τις πληροφορίες των πλαισίων διαλόγου που εμφανίζονται κατά την κλήση μεθόδων (μεταξύ των οποίων και η σύνταξη των αντίστοιχων εντολών στη Java) καταγράφουν σε ένα πίνακα (διαφορετικό για κάθε αντικείμενο) τις εντολές και τις μεταβολές των τιμών των ιδιοτήτων του. Οι σπουδαστές μπορούν στη συνέχεια αξιοποιώντας τους πίνακες που έχουν συμπληρώσει να λύσουν το ίδιο πρόβλημα δημιουργώντας τα αντικείμενα με το γνωστό τρόπο και εκτελώντας τις εντολές μία-μία γράφοντας τον αντίστοιχο κώδικα στο εργαλείο Code pad του περιβάλλοντος BlueJ.

5. Συμπεράσματα

Η διδασκαλία του αντικειμενοστρεφούς προγραμματισμού συνοδεύεται από ποικίλες δυσκολίες και παρανοήσεις και για αρκετά χρόνια θεωρούνταν ιδιαίτερα δύσκολη. Ωστόσο, αρκετά εκπαιδευτικά εργαλεία και περιβάλλοντα δημιουργήθηκαν, ενώ πληθώρα μελετών διεξήχθησαν που ανέδειξαν τις σημαντικότερες δυσκολίες και παρανοήσεις. Η αξιοποίηση των διαθέσιμων εργαλείων σε συνδυασμό με τη συσσωρευμένη εμπειρία μπορεί να συντελέσει σε μια πιο επιτυχημένη διδασκαλία και εκμάθηση του αντικειμενοστρεφούς προγραμματισμού. Στην παρούσα εργασία έγινε μια προσπάθεια ανάδειξης των σημαντικότερων εννοιών στις οποίες πρέπει να επικεντρώνεται η διδασκαλία του αντικειμενοστρεφούς προγραμματισμού και σύνοψης των σημαντικότερων δυσκολιών και παρανοήσεων που έχουν καταγραφεί για τις έννοιες αυτές. Η συνεισφορά της εργασίας εντοπίζεται κυρίως στις προτεινόμενες διδακτικές παρεμβάσεις που είναι ανεξάρτητες της γλώσσας και του περιβάλλοντος που θα επιλεγεί για τη διδασκαλία του αντικειμενοστρεφούς προγραμματισμού και στην επίδειξη του τρόπου εφαρμογής τους στο BlueJ.

Βιβλιογραφία

Carter, J. & Fowler, A. (1998), Object Oriented Students? *SIGCSE Bulletin*, 30(3), 271.

Fleury, A. (2000). Programming in Java: student-constructed rules. *SIGCSE Bulletin*, 32(1), 197-201.

Holland, S. Griffiths, R., Woodman, M. (1997), Avoiding object misconceptions. *SIGCSE Bulletin*, 29(1), 1997, 131-134.

Ragonis, N. & Ben-Ari, M. A. (2005), Long-Term Investigation of the Comprehension of OOP Concepts by Novices, *International Journal of Computer Science Education*, 15(3), 2005, 203-221.

Sanders, K. & Thomas, L. (2007), Checklists for grading object-oriented CS1 programs: concepts and misconceptions, *SIGCSE Bull.* 39(3), 166-170.

Thomasson, B., Ratcliffe, M. and Thomas, L. (2006), Identifying novice difficulties in object oriented design. *SIGCSE Bulletin,* 38(3), 28-32.

Xinogalos, S. & Satratzemi, M. (2009), A Long-Term Evaluation and Reformation of an Object Oriented Design and Programming Course, *Proceedings of the 9th IEEE International Conference on Advanced Learning Technologies*, July 2009, Riga, Latvia, IEEE Computer Society Press, 64-66.

Ξυνόγαλος, Σ. (2008), Μελέτη των Δυσκολιών των Φοιτητών για την Έννοια του «Αντικειμένου» στον Αντικειμενοστραφή Προγραμματισμό, *Πρακτικά 4ου Πανελλήνιου Συνεδρίου «Διδακτική της Πληροφορικής»*, 91-100.