

# Δομημένος Προγραμματισμός στη Δευτεροβάθμια Εκπαίδευση. Απάντηση Ερευνητικών Ερωτημάτων

Πέτρος Σαλαβασίδης

Εκπαιδευτικός Πληροφορικής, Μεταπτυχιακός φοιτητής Δ.Π.Μ.Σ. στα Πληροφοριακά Συστήματα Πανεπιστημίου Μακεδονίας  
petros\_salavasidis@sch.gr

## Περίληψη

Στην ελληνική δημόσια Δευτεροβάθμια Εκπαίδευση στη Γ' Τάξη του Ενιαίου Λυκείου και του Επαγγελματικού Λυκείου διδάσκονται μαθήματα Δομημένου Προγραμματισμού. Η παρούσα έρευνα επιχειρεί να απαντήσει επιλεγμένα ερευνητικά ερωτήματα που σχετίζονται με την χρησιμότητα-αναγκαιότητα ύπαρξης μαθήματος Δομημένου Προγραμματισμού στη Γ' Τάξη του Ενιαίου και του Επαγγελματικού Λυκείου. Η έρευνα αξιοποιεί την υπάρχουσα βιβλιογραφία.

**Λέξεις κλειδιά:** Δομημένος Προγραμματισμός, αναλυτική σκέψη, συνθετική σκέψη.

## Abstract

In the Third Grade of Lyceum and Vocational High School in the Greek public Secondary Education courses of structured programming are being taught. This article attempts, using related bibliography, to answer selected research questions related to the usefulness-necessity of a Structured Programming course in the Third Grade of Lyceum and Vocational High School.

**Keywords:** *Structured Programming, analytical thinking, synthesis thinking.*

## 1. Εισαγωγή

Η Επιστήμη των Υπολογιστών ή Επιστήμη της Πληροφορικής (Computer Science, CS) είναι μια νέα, σε ηλικία, επιστήμη (Dale & Lewis, 2002). Κατά τους Newell και Simon (1976) είναι η επιστήμη που μελετά τις έννοιες (phenomena) τις σχετικές με τους υπολογιστές. Επίσης οι Dale και Lewis (2002) υποστηρίζουν ότι η Επιστήμη των Υπολογιστών είναι μια επιστήμη η οποία ασχολείται με την κατασκευή και δοκιμή μοντέλων των φυσικών φαινομένων. Σύμφωνα με τους Daintith και Wright (2008) η Επιστήμη των Υπολογιστών είναι η επιστήμη της μελέτης των ηλεκτρονικών υπολογιστών, των βασικών αρχών που τους διέπουν αλλά και της χρήσης τους. Ενδεικτικά αντικείμενα της Επιστήμης των Υπολογιστών είναι: αλγόριθμοι και δομές δεδομένων, γλώσσες προγραμματισμού, γραφικά, λειτουργικά συστήματα, αριθμητικοί και συμβολικοί υπολογισμοί, μεθοδολογία και μηχανική λογισμικού κ.α. (Dale & Lewis, 2002).

Ο diSessa (2000) αναφερόμενος στο ρόλο που διαδραμάτισε η Επιστήμη των Υπολογιστών και που συνεχίζει καθημερινά να διαδραματίζει στην κοινωνία, αλλά και το πολιτισμό, σχολιάζει ότι είναι τόσο δραστικός που κυριολεκτικά την μεταμορφώνει. Η καθημερινότητα μας, στους περισσότερους εάν όχι σε όλους τους τομείς, κατακλύζεται από την «επαφή» με τους Υπολογιστές τόσο στην Εκπαίδευση, στην Ιατρική, στις Τηλεπικοινωνίες, στη Βιομηχανία, στη Διακυβέρνηση, στο Στρατό και σε άλλους τομείς (Dale & Lewis, 2002). Μάλιστα συμφωνά με τον diSessa (2000) οι εξελίξεις στην Επιστήμη των Υπολογιστών έχουν δημιουργήσει την «Επανάσταση της Πληροφορίας» η οποία μπορεί κάλλιστα να συγκριθεί με τη Βιομηχανική Επανάσταση.

Στην Εκπαίδευση η Επιστήμη των Υπολογιστών είναι ένα σημαντικό κεφάλαιο, ακαδημαϊκής και μη, μελέτης. Βέβαια υπάρχουν πολλές εφαρμογές και τεχνολογίες που σχετίζονται με αυτήν, όπως οι ονομαζόμενες «Τεχνολογίες της Πληροφορίας και Επικοινωνιών» (Information and Communication Technologies) γνωστές και ως ΤΠΕ (ICT). Ενδεικτικά αναφέρουμε παραδείγματα ΤΠΕ όπως τις πλατφόρμες λογισμικού για εξ' αποστάσεως εκπαίδευση, τους διαδραστικούς πίνακες κ.α. Είναι σημαντικό να αναφερθεί ότι συχνά συγχέονται οι δύο αυτοί όροι, η Επιστήμη των Υπολογιστών και οι Τεχνολογίες της Πληροφορίας και Επικοινωνιών. Η Wallace (2009) ορίζει τις ΤΠΕ ως μια περιοχή του εκπαιδευτικού προγράμματος που σχετίζεται με τη χρήση των εφαρμογών (applications) των Ηλεκτρονικών Υπολογιστών και των διάφορων εκπαιδευτικών τεχνολογιών και συμπληρώνει ότι ο όρος «Τεχνολογίες της Πληροφορίας και Επικοινωνιών» χρησιμοποιείται για να αναφερθούμε σε πόρους (resources) που χρησιμοποιούνται υποστηρικτικά στην εκπαίδευση, όπως οι υπολογιστές, οι τηλεοράσεις και οι συσκευές ήχου. Έτσι ξεκαθαρίζει το τοπίο όσον αφορά τη σύγχυση μεταξύ αυτών των δύο όρων.

## ***2. Ο Δομημένος Προγραμματισμός***

Σύμφωνα με τους Dahl, Dijkstra και Hoare (1972), ο Δομημένος Προγραμματισμός είναι ένα σύνολο αρχών που ακολουθούνται για τη «δημιουργία» (συγγραφή) καλά δομημένων προγραμμάτων. Ο Henderson (2009) συμπληρώνει ότι συνέπεια της καλής δομής που προτείνεται από τον Δομημένο Προγραμματισμό θα μπορούσε να είναι και η ορθότητα του.

### ***2.1 Ο Δομημένος Προγραμματισμός στην ελληνική δημόσια Δευτεροβάθμια Εκπαίδευση***

Όπως προαναφέραμε η Επιστήμη των Υπολογιστών δεν αφορά μόνο την ακαδημαϊκή-πανεπιστημιακή εκπαίδευση αλλά και άλλες μορφές εκπαίδευσης. Στην ελληνική δημόσια Δευτεροβάθμια Εκπαίδευση ένα από τα αντικείμενα που διδάσκονται στη Γ' Τάξη του Λυκείου, του Ενιαίου Λυκείου αλλά και του Επαγγελματικού Λυκείου, είναι ο Δομημένος Προγραμματισμός.

Το μάθημα του Δομημένου Προγραμματισμού στη Γ΄ Τάξη του Ενιαίου Λυκείου ονομάζεται «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» και συνοδεύεται από το ομώνυμο εγχειρίδιο (Βακάλη, Γιαννόπουλος, Ιωαννίδης, Κοίλιας, Μάλαμας, Μανωλόπουλος & Πολίτης, 1999). Στη Γ΄ Τάξη του Επαγγελματικού Λυκείου ονομάζεται «Δομημένος Προγραμματισμός» και συνοδεύεται από το εγχειρίδιο «Προγραμματισμός Υπολογιστών» (Σιδερίδης, Γιαλούρης, Μπακογιάννης & Σταθόπουλος, 2000). Αξίζει να σημειωθεί ότι και τα δύο μαθήματα εξετάζονται πανελλαδικά, το πρώτο στην Τεχνολογική Κατεύθυνση του Ενιαίου Λυκείου και το δεύτερο στον Τομέα Πληροφορικής και Δικτύων του Επαγγελματικού Λυκείου και ότι ακολουθούν παρόμοια διδακτική ύλη.

Τα μαθήματα αυτά στο αναλυτικό τους πρόγραμμα, που έχει καταρτιστεί από το Παιδαγωγικό Ινστιτούτο, θέτουν ως γενικό σκοπό οι μαθητές να αναπτύξουν αναλυτική και συνθετική σκέψη, να αποκτήσουν ικανότητες μεθοδολογικού χαρακτήρα και να μπορούν να επιλύσουν απλά σχετικά προβλήματα αλλά και να αναπτύξουν απλές εφαρμογές σε προγραμματιστικό περιβάλλον. Στο πλαίσιο των μαθημάτων αυτών οι μαθητές διδάσκονται στην αρχή αντικείμενα όπως ανάλυση προβλήματος, βασικές έννοιες αλγορίθμων, δομές δεδομένων, αλγόριθμους, τεχνικές σχεδίασης αλγορίθμων και τέλος την υλοποίηση αυτών σε «ψευδοκώδικα», διαγράμματα ροής και κάποια γλώσσα προγραμματισμού όπως «ΓΛΩΣΣΑ» στη Γ΄ Τάξη του Ενιαίου Λυκείου (Βακάλη κ.α., 1999) και PASCAL στο Γ΄ Τάξη του Επαγγελματικού Λυκείου (Σιδερίδης κ.α., 2000).

### **3. Ο σκοπός της έρευνας**

Ο σκοπός της παρούσας έρευνας μέσω βιβλιογραφικής ανασκόπησης είναι η διερεύνηση της χρησιμότητας-αναγκαιότητας ύπαρξης μαθήματος Δομημένου Προγραμματισμού στη Γ΄ Τάξη, του Ενιαίου Λυκείου και του Επαγγελματικού Λυκείου, ως μάθημα που αποσκοπεί στην ανάπτυξη της αναλυτικής και συνθετικής σκέψης αλλά παράλληλα και της ανάπτυξης της μεθοδικότητας στην κατανόηση και επίλυση προβλημάτων - ασκήσεων.

#### **3.1 Ερευνητικά ερωτήματα που εντοπίζονται**

Αρκετά ερωτήματα εντοπίζονται-δημιουργούνται αλλά παραθέτουμε μερικά από αυτά που θεωρούμε σημαντικότερα και πρέπει να σημειώσουμε ότι τελικά μόνο τα τέσσερα πρώτα εξ' αυτών θα απασχολήσουν το παρόν άρθρο, τα υπόλοιπα θα αποτελέσουν υλικό για μελλοντική εμβάθυνση στη μελέτη του θέματος. Τα ερευνητικά ερωτήματα είναι:

- Είναι κατάλληλη η ηλικία των μαθητών της Γ΄ Τάξης του Ενιαίου Λυκείου και του Επαγγελματικού Λυκείου ώστε να διδάσκονται μαθήματα Δομημένου Προγραμματισμού;

- Μαθαίνουν οι μαθητές μέσω του μαθήματος του Δομημένου Προγραμματισμού να προγραμματίζουν;
- Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την αναλυτική σκέψη των μαθητών;
- Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την συνθετική σκέψη των μαθητών;
- Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την μεθοδικότητα στην κατανόηση και την επίλυση ενός προβλήματος;
- Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την δημιουργικότητα των μαθητών;
- Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την αυτοπεποίθηση των μαθητών;
- Η μέχρι τώρα διδασκαλία μαθημάτων Δομημένου Προγραμματισμού ποια αποτελέσματα είχε;

#### **4. Ανάλυση των ερευνητικών ερωτημάτων**

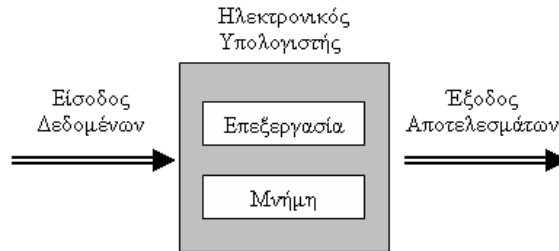
Παρακάτω παραθέτουμε την ανάλυση των ερευνητικών ερωτημάτων αξιοποιώντας την υπάρχουσα βιβλιογραφία.

##### **4.1 Είναι κατάλληλη η ηλικία των μαθητών της Γ' Τάξης του Ενιαίου Λυκείου και του Επαγγελματικού Λυκείου ώστε να διδάσκονται μαθήματα Δομημένου Προγραμματισμού;**

Οι γλωσσολόγοι διατυπώνουν μια γενική άποψη ότι οι άνθρωποι είναι «φτιαγμένοι» για τη «γλώσσα» (Guzdial, 2009). Ο εγκέφαλος μας στη διάρκεια των αιώνων έχει εξελιχθεί ώστε να προσλαμβάνει, να επεξεργάζεται τη «γλώσσα» και να αξιοποιεί τα αποτελέσματα αυτής της διεργασίας. Η «γλώσσα» αυτή είναι η «φυσική» γλώσσα (diSessa, 2000).

Ο Piaget (1950) με επίκεντρο τις διαδικασίες της ψυχοδιανοητικής ανάπτυξης του παιδιού διατυπώνει την εξελικτική πορεία του ατόμου σε τέσσερα στάδια-περιόδους, η περίοδος των τυπικών νοητικών πράξεων (4<sup>η</sup> Περίοδος), από τον 11ο-12ο χρόνο είναι η περίοδος που μας ενδιαφέρει. Σε αυτή τη περίοδο, η οποία συμπίπτει με την αρχή της εφηβείας και με την ένταξη του παιδιού στη Δευτεροβάθμια Εκπαίδευση, το παιδί πλέον αρχίζει να ωριμάζει ψυχοδιανοητικά, αρχίζει να αξιολογεί προηγούμενες και νέες γνώσεις και σκέφτεται χρησιμοποιώντας αφηρημένες έννοιες καθώς ταυτίζει αντικείμενα και έννοιες, διαφοροποιεί, αιτιολογεί και αντιδιαστέλει (OECD, 2007; Piaget, 1950; Πιπερόπουλος, 2007). Αξίζει να σημειώσουμε και την σύνδεση της εγκεφαλικής λειτουργίας με τις ανάγκες του οργανισμού σε κάθε περίοδο και ειδικότερα την αυξημένη εγκεφαλική λειτουργία τη συγκεκριμένη περίοδο (OECD, 2007).

Ο Προγραμματισμός αναφέρεται στη διαχείριση μιας «τεχνητής» γλώσσας, η οποία δημιουργήθηκε για ένα συγκεκριμένο σκοπό, την επικοινωνία με μη-ανθρώπινο παράγοντα, τον Υπολογιστή, και με απώτερο σκοπό την εκτέλεση από τον Υπολογιστή ανθρώπινων εντολών (Guzdial, 2009). Πρέπει επίσης να σημειώσουμε ότι συμφωνά με τον Τσίρο (2009) το μοντέλο λειτουργίας του Υπολογιστή και το ανθρώπινο νοητικό σύστημα είναι το ίδιο χρησιμοποιώντας «εισόδους», «επεξεργασίες», «μνήμη» ή «αναπαραστάσεις» και «εξόδους» όπως φαίνεται και στις εικόνες παρακάτω, Σχήμα 1 και Σχήμα 2.



**Σχήμα 1:** Μοντέλο Λειτουργίας του Υπολογιστή (Dale & Lewis, 2002)



**Σχήμα 2:** Μοντέλο Λειτουργίας του ανθρώπινου νοητικού συστήματος (Τσίρος, 2009)

Με βάση τα παραπάνω, δηλαδή τη φυσική εξέλιξη του ανθρώπινου εγκεφάλου ώστε να κατανοεί τη «φυσική» γλώσσα (Guzdial, 2009), την ηλικιακή ψυχοδιανοητική ωρίμανση των μαθητών αυτής της ηλικίας (OECD, 2007; Piaget, 1950; Πιπερόπουλος, 2007) και επιπλέον λαμβάνοντας υπόψη τις ομοιότητες λειτουργίας του ανθρώπινου νοητικού συστήματος (Τσίρος, 2009) και του μοντέλου λειτουργίας του Υπολογιστή (Dale & Lewis, 2002) καταλήγουμε στο συμπέρασμα ότι οι μαθητές της Γ' Τάξης του Λυκείου είναι σε κατάλληλη ηλικία ώστε να διδαχθούν Δομημένο Προγραμματισμό. Τέλος, πρέπει να σημειωθεί και το γεγονός ότι αρκετοί από τους μαθητές θα επιλέξουν να ακολουθήσουν μετά τις πανελλήνιες εξετάσεις, τις Θετικές

Επιστήμες σε ακαδημαϊκό επίπεδο και είναι μεγάλη η πιθανότητα να έρθουν ξανά σε επαφή με κάποια μορφή Προγραμματισμού (ΥΠΔΒΜΘ, 2010).

#### ***4.2 Μαθαίνουν οι μαθητές μέσω του μαθήματος του Δομημένου Προγραμματισμού να προγραμματίζουν;***

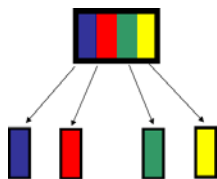
Παρά το γεγονός ότι ο σκοπός των σχολικών μαθημάτων Δομημένου Προγραμματισμού δεν είναι να γίνουν οι μαθητές προγραμματιστές αλλά να αναπτύξουν αναλυτική και συνθετική σκέψη (Βακάλη κ.α., 1999; Σιδερίδης κ.α., 2000), τελικά οι μαθητές μέσω των μαθημάτων έρχονται σε επαφή με τον Δομημένο Προγραμματισμό και τον χρησιμοποιούν ως εργαλείο στην επίλυση των προβλημάτων/ασκήσεων μέσω της χρήσης γλωσσών προγραμματισμού που διδάσκονται, η «ΓΛΩΣΣΑ» στο Ενιαίο Λύκειο και η PASCAL στο Επαγγελματικό Λύκειο. Πρέπει επίσης να σημειώσουμε ότι η ανάπτυξη αλγοριθμικής σκέψης (Piater, 2009; Solomon, 2007) είναι μία δεξιότητα που σχετίζεται ιδιαίτερα με την Επιστήμη των Υπολογιστών και η οποία σίγουρα είναι σημαντική (Hromkovic, 2006; Solomon, 2007) αλλά η οποία είναι ανεξάρτητη του Προγραμματισμού (Futschek, 2006; Solomon, 2007). Βέβαια αρχικά η εκμάθηση, έστω και βασικών, προγραμματιστικών δεξιοτήτων είναι επίπονη (Holvikivi, 2010, Weinberg, 1971) και οι μαθητές συνήθως αντιμετωπίζουν δυσκολίες στην ανάπτυξη προγραμματιστικών δεξιοτήτων (Guzdial, 2009; Weinberg, 1971) και τείνουν να έχουν αρνητική στάση προς τα μαθήματα Προγραμματισμού (Rivka, Ben-Ari & Armoni, 2009). Όμως παρά την αρχική δυσκολία που αντιμετωπίζουν οι μαθητές τα αποτελέσματα σε γνωστικό επίπεδο, σε μεταγνωστικό επίπεδο αλλά και σε επίπεδο δεξιοτήτων που αποκτούν είναι ιδιαίτερα σημαντικά (diSessa, 2000; Linn, 1985; Solomon, 2007, Weinberg, 1971).

Πρέπει να τονιστεί ότι ο Δομημένος Προγραμματισμός, στη Δευτεροβάθμια Εκπαίδευση, πρέπει να λειτουργεί όχι ως ένα στείρο εκπαιδευτικό εργαλείο εκμάθησης μίας ή περισσότερων προγραμματιστικών γλωσσών χωρίς σύνδεση με τα υπόλοιπα μαθήματα (diSessa, 2000) αλλά ως ένα δημιουργικό (Kleiman, 2008) εκπαιδευτικό εργαλείο δίνοντας, όπως άλλωστε τονίζεται και από τη Βακάλη (1999) και τον Σιδερίδη (2000), έμφαση στην ανάπτυξη της αναλυτικής και συνθετικής σκέψης των μαθητών και των ικανοτήτων τους μεθοδολογικού χαρακτήρα για την κατανόηση και επίλυση διαφόρων προβλημάτων στα πλαίσια του σχολείου και εκτός αυτών (Linn, 1985).

#### ***4.3 Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την αναλυτική σκέψη των μαθητών;***

Κατά τον Amer (2005) η αναλυτική σκέψη είναι το κομμάτι της σκέψης το οποίο σχετίζεται με την κατανόηση των «μερών» ή «κομματιών» μιας «κατάστασης», βλέπε Σχήμα 3, «σπάει» τα «αντικείμενα» στα «μέρη» από τα οποία αποτελούνται και επίσης βρίσκει τις όποιες διαφορές υπάρχουν μεταξύ των αντικειμένων. Η

αναλυτική σκέψη είναι ευκολότερη της συνθετικής διότι μαθαίνουμε να σκεφτόμαστε έτσι από τη γέννηση μας (Amer, 2005).



**Σχήμα 3:** Ανάλυση του «αντικειμένου» σε «κομμάτια» (Amer, 2005)

Στον Δομημένο Προγραμματισμό μια από τις έννοιες κλειδιά είναι η λέξη Πρόβλημα, με τον όρο αυτό εννοείται μια κατάσταση η οποία χρήζει αντιμετώπισης ενώ η λύση της δεν είναι γνωστή και ούτε προφανής (Βακάλη κ.α. 1999; Σιδερίδης κ.α. 2000; Dale & Lewis, 2002). Καθημερινά ο άνθρωπος ανεξάρτητα από ηλικιακό, κοινωνικό, οικονομικό ή άλλο επίπεδο, έρχεται αντιμέτωπος με προβλήματα τα οποία των απασχολούν είτε σε ατομικό είτε σε κοινωνικό επίπεδο (Βακάλη κ.α., 1999; Σιδερίδης κ.α., 2000). Ιδιαίτερα σημαντική για την «επίλυση» του προβλήματος είναι αρχικά η κατανόηση του (Βακάλη κ.α., 1999; Dale & Lewis, 2002; Fee & Holland-Minkley, 2010; Σιδερίδης κ.α., 2000). Και στην εκπαιδευτική διαδικασία η κατανόηση του προβλήματος παίζει σημαντικό ρόλο και θέτει τις βάσεις για τη βελτίωση της διανοητικής ικανότητας, της δημιουργικότητας αλλά και της ενίσχυσης της αυτοπεποίθησης του μαθητή (diSessa, 2000) δημιουργώντας την κατάλληλη υποκίνηση για «περισσότερη» και «ποιοτικότερη» μάθηση (ChanLin, 2010).

Στο μάθημα του Δομημένου Προγραμματισμού ο μαθητής διδάσκεται τη σημασία της κατανόησης του Προβλήματος και το «σπάσιμο» του Προβλήματος στα συστατικά του μέρη, δηλαδή δεδομένα, ζητούμενα, συνθήκες και σχέσεις (Βακάλη κ.α., 1999; Dale & Lewis, 2002; Romeike, 2008; Σιδερίδης κ.α., 2000). Αξίζει να σημειώσουμε ότι έρευνες αναδεικνύουν τη σπουδαιότητα της Προβληματο-Κεντρικής Μάθησης αλλά και ειδικότερα την θετική αξιοποίηση της στη Δευτεροβάθμια Εκπαίδευση (Boyer, Phillips, Wallis, Vouk, & Lester, 2009; Fee & Holland-Minkley, 2010, ChanLin, 2010). Η Προβληματο-Κεντρική Μάθηση ως μεθοδολογία συνδέεται άμεσα με την Επιστήμη των Υπολογιστών και ιδιαίτερα με τον Προγραμματισμό (Dale & Lewis, 2002), επικεντρώνεται στην εκπαίδευση του μαθητή μέσω της παρουσίασης, ανάλυσης και επίλυσης προβλημάτων/ασκήσεων με τη συνδρομή του εκπαιδευτικού για την επίτευξη των εκπαιδευτικών στόχων που έχουν τεθεί και δίνει τη δυνατότητα στο μαθητή να αναπτύξει δεξιότητες και παράλληλα ενισχύει την αυτοπεποίθηση, την ανάγκη για δημιουργία και τη διάθεση για προσωπική αυτό-εκπαίδευση (ChanLin, 2010, Fee & Holland-Minkley, 2010, Hallinger & Bridges, 2007, Wurdinger & Rudolph, 2009). Στη συνέχεια παρουσιάζεται η έννοια του αλγόριθμου, δηλαδή μιας πεπερασμένης σειράς βημάτων τα οποία είναι αυστηρά καθορισμένα και εκτελέσιμα σε πεπερασμένο χρόνο και ο στόχος τους είναι η επίλυση ενός προβλήματος (Βακάλη κ.α., 1999; Dale & Lewis,

2002; Σιδερίδης κ.α., 2000). Ο μαθητής μαθαίνει και χρησιμοποιεί τεχνικές αναπαράστασης αλγορίθμων (ελεύθερο κείμενο, διαγραμματικές τεχνικές, φυσική γλώσσα και κωδικοποίηση είτε σε «ΓΛΩΣΣΑ» είτε σε PASCAL), τις δομές (ακολουθίας, επιλογής, επανάληψης, σταθερές, μεταβλητές, τελεστές και εκφράσεις) που είναι απαραίτητες για την αναπαράσταση αλγορίθμων και τις δομές δεδομένων (πίνακες, στοίβες και ουρές) με έμφαση στους πίνακες. Μελετά και λύνει μια σειρά από διαβαθμισμένα σε δυσκολία προβλήματα (Βακάλη κ.α. 1999; Σιδερίδης κ.α., 2000).

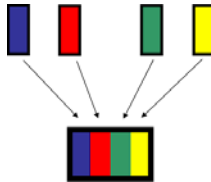
Ακολουθούν ενδεικτικά προβλήματα διαβαθμιζομένης δυσκολίας που καλείτε να λύσει ο μαθητής αρχικά αναλύοντας το πρόβλημα, στη συνέχεια χρησιμοποιώντας «ψευδοκώδικα» και τέλος δημιουργώντας το αντίστοιχο διάγραμμα ροής:

- Να διαβάσετε την παρακάτω συνταγή και να διακρίνεται α. τα συστατικά που χρειάζεται η νοικοκυρά για την παρασκευή του γλυκίσματος και β. τη διαδικασία που θα πρέπει να ακολουθηθεί (Βακάλη κ.α., 1999).
- Να διαβαστούν δύο αριθμοί, να υπολογισθεί και να εκτυπωθεί το άθροισμα τους (Βακάλη κ.α., 1999).

Η επίλυση των παραπάνω προβλημάτων, που κάθε άλλο παρά εξειδικευμένα είναι (Romeikié, 2008), σε συνάρτηση με την καθημερινή τριβή (OECD, 2007), ενισχύουν την αναλυτική σκέψη η οποία όπως έχουμε ήδη σημειώσει είναι ο τρόπος σκέψης τον οποίο μαθαίνουμε μόλις γεννηθούμε (Amer, 2005). Είναι σημαντικό να σημειωθεί ότι το γεγονός στα πλαίσια των δύο μαθημάτων το κάθε πρόβλημα προσεγγίζεται με ανάλυση, με «ψευδοκώδικα» και με διάγραμμα ροής δημιουργώντας μια καλύτερη εικόνα στο μαθητή της λύσης του προβλήματος, ακόμα και του προβλήματος της συνταγής που δόθηκε παραπάνω (Βακάλη κ.α., 1999; Σιδερίδης κ.α., 2000).

#### **4.4 Ο Δομημένος Προγραμματισμός μπορεί να ενισχύσει την συνθετική σκέψη των μαθητών;**

Κατά τον Amer (2005) η συνθετική σκέψη είναι το κομμάτι της σκέψης το οποίο σχετίζεται με την κατανόηση του τρόπου με τον οποίο συσχετίζονται τα «κομμάτια» μιας «κατάστασης», μας επιτρέπει να βρούμε τα πρότυπα (patterns) που σχετίζονται με τα «μέρη» των «αντικειμένων» που προηγουμένως έχουμε «σπάσει» σε μικρότερα και τέλος να βρούμε ομοιότητες μεταξύ των αντικειμένων, βλέπε Σχήμα 4.



**Σχήμα 4:** Σύνθεση «κομματιών» στο «αντικείμενο» (Amer, 2005)



Η συνθετική σκέψη είναι δυσκολότερη της αναλυτικής διότι δεν μαθαίνουμε να σκεφτόμαστε με αυτό τον τρόπο, βέβαια αξίζει να σημειώσουμε ότι παρόλα αυτά η συνθετική σκέψη λειτουργεί ασυναίσθητα (Amer, 2005).

Στα μαθήματα του Δομημένου Προγραμματισμού της Γ' Τάξης του Ενιαίου Λυκείου και του Επαγγελματικού Λυκείου έχοντας μάθει ο μαθητής όσα παρουσιάσαμε στη προηγούμενη παράγραφο στη συνέχεια διδάσκεται μια σειρά από αλγόριθμους οι οποίοι παρουσιάζονται με διαβαθμισμένη δυσκολία. Ενδεικτικά αναφέρουμε τους αλγόριθμους Πολλαπλασιασμό αλά Ρώσικα, Εύρεση στοιχείου σε μονοδιάστατο ή δυσδιάστατο πίνακα, αλγόριθμους Ταξινόμησης, τις λειτουργίες στοιβάς και ουράς κ.α. (Βακάλη κ.α., 1999; Σιδερίδης κ.α. 2000).

Η σύνθεση είναι πιο επίπονη (Amer, 2005) αλλά είναι δυνατό να επιτευχθεί εφόσον οι έννοιες που αφορούν την ανάλυση του προβλήματος έχουν γίνει κατανοητές και συνδεθούν σταδιακά με τους αλγόριθμους οι οποίοι παρουσιάζουν έτσι και αλλιώς μια αυξημένη δυσκολία (Βακάλη κ.α., 1999; Σιδερίδης κ.α. 2000). Τυπικό παράδειγμα σύνθεσης είναι να δοθεί διάγραμμα ροής και ζητηθεί από το μαθητή να διακρίνει τυχόν λάθη στη λογική του διαγράμματος ροής αλλά και να αναδημιουργήσει το πρόβλημα (Βακάλη κ.α., 1999; Σιδερίδης κ.α. 2000).

## 5. Συμπεράσματα

Ο άνθρωπος καθημερινά καλείται να «προγραμματίσει» τη ζωή του σε επίπεδο ατομικό, κοινωνικό, οικονομικό κ.α. Ο Δομημένος Προγραμματισμός είναι από τα βασικά στοιχεία της Επιστήμης των Υπολογιστών, μιας επιστήμης η οποία διαδραματίζει σημαντικό ρόλο στην καθημερινότητα μας (Dahl, Dijkstra, & Hoare, 1972; Dale & Lewis, 2002; diSessa, 2000). Ο Δομημένος Προγραμματισμός παρέχει μια οργανωμένη προσέγγιση στην επίλυση προβλημάτων και είναι σημαντική η συνεισφορά του στην εκπαίδευση των μαθητών της Γ' Τάξης του Ενιαίου Λυκείου και του Επαγγελματικού Λυκείου, ως ένα μάθημα που αναπτύσσει την αναλυτική και συνθετική σκέψη των μαθητών και παράλληλα τους παρουσιάζει βασικές προγραμματιστικές έννοιες που μπορούν να αξιοποιηθούν εντός και εκτός του στενού σχολικού περιβάλλοντος.

## Βιβλιογραφία

- Amer, A. (2005). *Analytical Thinking*. Cairo: Center for Advancement of Postgraduate Studies and Research in Engineering Sciences, Faculty of Engineering – Cairo University (CAPSCU).
- Boyer, K. E., Phillips, R., Wallis, M. D., Vouk, M. A. & Lester, J. C. (2009). Investigating the Role of Student Motivation in Computer Science Education Through One-to-One Tutoring. *Computer Science Education*, 19(2), 111-135.
- ChanLin, L. (2008). Technology Integration Applied to Project-Based Learning in Science. *Innovations in Education and Teaching International*, 45(1), 55-65.

- Dahl, O. J., Dijkstra, E. W. & Hoare, C. A. R. (1972). *Structured Programming*. London: Academic Press LTD.
- Dale, N. & Lewis, J. (2002). *Computer Science Illuminated*. Sudbury: Jones and Bartlett Publishers, Inc.
- Daintith, E. J., & Wright, E. (2008). "Computer Science". A Dictionary of Computing. *Oxford Reference Online*. Oxford University Press. Retrieved 24 December 2010 from <http://www.oxfordreference.com/views/ENTRY.html?subview=Main&entry=t11.e963>
- diSessa, A. A. (2000). *Changing Minds - Computers, Learning and Literacy*. Massachusetts Institute of Technology Press.
- Fee, S. B. & Holland-Minkley, A. M. (2010). Teaching Computer Science Through Problems, Not Solutions. *Computer Science Education*, 20(2), 129-144.
- Futschek, G. (2006). Algorithmic Thinking: The Key for Understanding Computer Science. *Informatics Education – The Bridge between Using and Understanding Computers. Lecture Notes in Computer Science*, 4226, 159-168.
- Guzdial, M. (2009). Why Is It So Hard to Learn to Program? In Oram, A. & Wilson, G. (eds). *Making Software, What Really Works and Why We Believe It*. USA: O' Reilly.
- Hallinger, P. & Bridges, E. (2007). *A Problem-based Approach for Management Education*. The Netherlands: Springer.
- Henderson, H. (2009). *Encyclopedia of Computer Science and Technology, Revised Edition*. New York: Facts On File, Inc.
- Holvikivi, J. (2010). Conditions for Successful Learning of Programming Skills. *IFIP Advances in Information and Communication Technology*, 324, 155-164.
- Hromkovic, J. (2006). Contributing to General Education by Teaching Informatics. *Informatics Education – The Bridge between Using and Understanding Computers. Lecture Notes in Computer Science*, 4226, 25-37.
- Kleiman, P. (2008). Towards Transformation Conceptions of Creativity in Higher Education. *Innovations in Education and Teaching International*, 45(3), 209-217.
- Linn, M. C. (1985). The Cognitive Consequences of Programming Instruction in Classrooms. *Educational Researcher*, 14(5), 14-29.
- Newell, A. & Simon, H. A. (1976). Computer Science as Empirical inquiry: symbols and search. *Communications of the ACM*, 13(3), 113-126.
- OECD (2007). *Understanding the Brain: The Birth of a Learning Science*. Paris: CERI Publications.
- Piaget, J. (1950). *The Psychology of Intelligence*. New York: Harcourt-Brace.
- Piater, J. H. (2009). Planning Readings: A Comparative Exploration of Basic Algorithms. *Computer Science Education*, 19(3), 179-192.
- Rivka, T., Ben-Ari, M. & Armoni, M. (2009). The Effect of CS Unplugged on Middle School Students' Views of CS. *Innovation and Technology in Computer Science Education* (Proceedings of the 14th Annual ACM's SIGCSE ITiCSE, 99-103). New York: Association for Computing Machinery.

- Romeike, R. (2008). What's My Challenge? The Forgotten Part of Problem Solving in Computer Science Education. In Mittermeir, R. T. & Syslo, M. M. (eds) *Lecture Notes in Computer Science 5090* (pp.122-135). Berlin: Springer
- Solomon, J. (2007). Putting the Science into Computer Science: Treating Introductory Computer Science as the Study of Algorithms. *ACM - Special Interest Group on Computer Science Education*, 39(2), 46-49.
- Wallace, E. S. (2009). "Information Communication Technology". A Dictionary of Education. *Oxford Online Reference*. Oxford University Press.
- Weinberg, G. M. (1971). *The Psychology of Computer Programming*. New York: Van Nostrand Reinhold Company.
- Wurdinger, S. & Rudolph, J. (2009). A different type of success: teaching important life skills through project based learning. *Improving Schools*, 12(2), 115-129.
- Βακάλη, Α., Γιαννόπουλος, Κ., Ιωαννίδης, Ν., Κοίλιας, Χ., Μάλαμας, Κ., Μανωλόπουλος, Ι. & Πολίτης, Π. (1999). *Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον*. Αθήνα: ΟΕΔΒ.
- Πιπερόπουλος, Γ. (2007). *Ψυχολογία*. Θεσσαλονίκη: Αυτοέκδοση.
- Σιδερίδης, Α., Γιαλούρης, Κ., Μπακογιάννης, Σ. & Σταθόπουλος, Κ. (2000). *Προγραμματισμός Υπολογιστών*. Αθήνα: ΟΕΔΒ.
- Τσίρος, Χ. (2009). Ο Κυρίαρχος Αισθητηριακός Τύπος ως Εκπαιδευτικό Μέσο Ανάπτυξης των Δεξιοτήτων Σκέψης και του Μαθησιακού Ύφους Διδάσκοντος και Διδασόμενου. *Το Μαθησιακό Ύφος*. 8-42. Τρίπολη: Ακαδημαϊκό.
- ΥΠΔΒΜΘ, (2010). *Αναζήτηση Βάσεων*. Ανακτήθηκε στις 24 Δεκεμβρίου 2010 από <http://www.ypepth.gr/efarmoges/baseis.php>