



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ & ΔΙΚΤΥΩΝ**

**Ανάπτυξη διδακτικού υλικού για την διδασκαλία του  
προγραμματισμού στο δημοτικό βασισμένο σε rubrics  
αξιολόγησης**

**SCRATCH**



**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ  
ΧΡΥΣΟΥΛΑ Ν. ΦΛΩΡΟΥ**

**Επιβλέποντες καθηγητές: κ. Χούστης Ηλίας, Καθηγητής ΤΜΗΥΤΔ  
κα. Χούστη Αικατερίνη, Καθηγήτρια ΤΜΗΥΤΔ**

**Ιούλιος 2009**

# Περίληψη

Στην παρούσα εργασία θα ασχοληθούμε με την διδασκαλία του προγραμματισμού σε μαθητές του δημοτικού σχολείου. Φυσικά, ακριβώς λόγω της ιδιαιτερότητας της παιδικής ηλικίας, θα πρέπει να αναζητήσουμε νέες μεθόδους διδακτικής προσέγγισης και αξιολόγησης, ξεφεύγοντας από τον κλασικό τρόπο διδασκαλίας του προγραμματισμού. Γι' αυτόν τον λόγο, θα χρησιμοποιήσουμε τους λεγόμενους «μικρόκοσμους προγραμματισμού» για την διδασκαλία του και την μέθοδο των ρουμπρίκων (rubrics) για την αξιολόγηση των μαθητών.

Στο πρώτο κεφάλαιο θα κάνουμε μια εισαγωγή για το πλαίσιο ένταξης της Πληροφορικής στην ελληνική εκπαίδευση, ξεκινώντας από το Νηπιαγωγείο και καταλήγοντας στο Ενιαίο Λύκειο. Θα σταθούμε, φυσικά, αρκετά στο δημοτικό και θα καλύψουμε τα ζητήματα του ρόλου της Πληροφορικής καθώς και των αξόνων υλοποίησης του σκοπού της Πληροφορικής στο δημοτικό.

Στο δεύτερο κεφάλαιο θα ασχοληθούμε με το αντικείμενο του προγραμματισμού Η/Υ. Συγκεκριμένα, θα παρουσιάσουμε τις απόψεις των παιδαγωγών και των ψυχολόγων για την ανάλυση της δραστηριότητας του προγραμματισμού, θα επιμείνουμε στο πεδίο της μάθησης του προγραμματισμού από παιδαγωγικής πλευράς και θα αναφέρουμε τις απαραίτητες γνωστικές δεξιότητες που συνιστούν προαπαιτούμενο για την μάθηση του προγραμματισμού. Στην συνέχεια, θα μιλήσουμε για τις εγγενείς δυσκολίες στην μάθηση του προγραμματισμού και τέλος, θα παρουσιάσουμε εν συντομία τους λόγους για τους οποίους η κλασική μέθοδος εισαγωγής στον προγραμματισμό κρίνεται αναποτελεσματική για παιδιά μικρών ηλικιών.

Στο τρίτο κεφάλαιο θα παρουσιάσουμε δυο εναλλακτικές προσεγγίσεις διδασκαλίας προγραμματισμού, τους διδακτικούς μικρόκοσμους και τον προγραμματισμό ρομποτικών εφαρμογών, που ανοίγουν μια νέα προοπτική στην εκπαιδευτική διαδικασία, όχι μόνο ως μέσο διδασκαλίας εισαγωγικών εννοιών του προγραμματισμού αλλά και ως μέσο αξιολόγησης των αντιλήψεων των μαθητών.

Στο τέταρτο κεφάλαιο θα γνωρίσουμε το εκπαιδευτικό περιβάλλον προγραμματισμού Scratch, που η αποδοχή του από την παγκόσμια εκπαιδευτική

κοινότητα είναι ιδιαίτερα θερμή. Αρχικά, θα γίνει μια περιγραφή των χαρακτηριστικών του, και στην συνέχεια θα ερευνησουμε την παιδαγωγική του χρήση καθώς και τα οφέλη από την αξιοποίησή του ως διδακτικό μέσο για την πρόσκτηση δεξιοτήτων προγραμματισμού.

Στο πέμπτο κεφάλαιο προσδιορίζεται η μεθοδολογία της αξιολόγησης ως ένα καθοριστικό μέσο για την απόκτηση και επεξεργασία σημαντικών πληροφοριών αναφορικά με τη συνεχή ή μη μαθησιακή πρόοδο των μαθητών. Συγκεκριμένα, θα παρουσιάσουμε την χρήση των ρουμπρικών (rubrics), οι οποίες αποτελούν εξαιρετικά χρήσιμα εργαλεία αξιολόγησης και ανατροφοδότησης τόσο για τους εκπαιδευτικούς όσο και για τους μαθητές, θα αναφερθούμε στην απροθυμία και αδυναμία των εκπαιδευτικών να εισάγουν τις ρούμπρικες (rubrics) στην διαδικασία της μάθησης και τέλος, θα δούμε κάποια χαρακτηριστικά υποδείγματα ρουμπρικών (rubrics).

Στο έκτο κεφάλαιο θα παρουσιάσουμε το διδακτικό υλικό που έχουμε αναπτύξει για την διδασκαλία του προγραμματισμού στο δημοτικό και το οποίο περιέχεται στο συνοδευτικό CD-ROM.

Τέλος, θα διατυπώσουμε κάποια συμπεράσματα από την εργασία μας και θα θέσουμε στόχους μελλοντικής εργασίας.

# Ευχαριστίες

Ύστερα από μια πορεία πέντε και πλέον χρόνων στο Τμήμα Μηχανικών Η/Υ, Τηλεπικοινωνιών & Δικτύων του Πανεπιστημίου Θεσσαλίας και με τις γνώσεις που αποκόμισα από τις σπουδές αυτές, ολοκληρώνω τις προπτυχιακές μου σπουδές με την εκπόνηση της παρούσας διπλωματικής εργασίας. Από την θέση αυτή θα ήθελα να ευχαριστήσω θερμά τον κ. Χούστη Ηλία, Καθηγητή του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών & Δικτύων, κύριο επιβλέποντα της εργασίας μου, για τις χρήσιμες παρατηρήσεις και υποδείξεις που μου παρείχε. Ευχαριστώ, επίσης, την συνεπιβλέποντα της εργασίας μου κα. Χούστη Αικατερίνη, Καθηγήτρια του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών & Δικτύων, για την καθοδήγηση της.

Ξεχωριστές ευχαριστίες θα ήθελα να αποδώσω στην κα. Μπαγιάτη Αικατερίνη, η οποία επέβλεψε στενά την προσπάθεια αυτή από την αρχή μέχρι το τέλος της καθώς και στους Pr. Ευαγγέλου Δήμητρα και Pr. Seam Brophy, Assistant Professors του Τμήματος Engineering Education του Purdue University, για τις πολύτιμες συμβουλές τους και το επιστημονικό υλικό το οποίο μου διέθεσαν.

Τέλος, ευχαριστώ θερμά την οικογένεια μου για την αμέριστη συμπαράσταση και ενθάρρυνση που μου παρείχε, όλα αυτά τα χρόνια για την ολοκλήρωση των σπουδών μου.

## **Συνοδευτικό CD-ROM**

Η παρούσα διπλωματική εργασία συνοδεύεται από ένα συμπληρωματικό CD-ROM το οποίο περιέχει σε ηλεκτρονική μορφή όλο το διδακτικό υλικό που σχεδιάσαμε και αναπτύξαμε για την διδασκαλία του προγραμματισμού στο δημοτικό.

# Περιεχόμενα

Περίληψη.....	2
Ευχαριστίες.....	4
Συνοδευτικό CD-ROM.....	5

## **1 Το πλαίσιο ένταξης της Πληροφορικής στην ελληνική εκπαίδευση ..... 13**

1.1 Η Πληροφορική στην προσχολική ελληνική εκπαίδευση.....	17
--	----

1.1.1 Το Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών (ΔΕΠΠΣ) για το νηπιαγωγείο.....	17
--	----

1.1.2 Η Πληροφορική στο ελληνικό νηπιαγωγείο.....	17
---	----

1.2 Η Πληροφορική στην πρωτοβάθμια ελληνική εκπαίδευση.....	19
---	----

1.2.1 Το ΕΠΠΣ και το ΔΕΠΠΣ για το δημοτικό .....	19
--	----

1.2.2 Ο σκοπός της Πληροφορικής στο δημοτικό .....	20
--	----

1.2.3 Άξονες υλοποίησης του σκοπού της Πληροφορικής στο δημοτικό.....	22
---	----

1.2.3.1 Ο υπολογιστής ως γνωστικό – διερευνητικό εργαλείο.....	23
--	----

1.2.3.2 Ο υπολογιστής ως «διανοητικό εργαλείο» (Mindtool).....	23
1.2.3.3 Ο υπολογιστής ως εποπτικό μέσο διδασκαλίας σε βασικά γνωστικά αντικείμενα.....	24
1.2.3.4 Ο υπολογιστής ως εργαλείο επικοινωνίας και αναζήτησης πληροφοριών.....	26
1.2.3.5 Ο πληροφορικός αλφαριθμητισμός .....	26
1.2.4 Χρήση και ένταξη του υπολογιστή στην μαθητική κοινότητα του δημοτικού σχολείου .....	27
1.2.5 Πιθανές χρήσεις των υπολογιστών από μαθητές δημοτικού .....	29
1.3 Η Πληροφορική στο Γυμνάσιο.....	31
1.3.1 Η Πληροφορική ως γνωστικό αντικείμενο στο ελληνικό γυμνάσιο.....	31
1.3.2 Βασικοί άξονες για τη διδασκαλία της Πληροφορικής στο γυμνάσιο .....	33
1.3.3 Γνώσεις & δεξιότητες μαθητών Γυμνασίου.....	34
1.3.4 Η διδασκαλία του προγραμματισμού στο Γυμνάσιο.....	34
1.4 Η Πληροφορική στο Ενιαίο Λύκειο .....	36
1.4.1 Η Πληροφορική ως μάθημα επιλογής και κύκλος σπουδών .....	36

1.4.2 Η Πληροφορική ως μάθημα γενικής παιδείας στο λύκειο .....	36
1.4.2.1 Γενικός σκοπός της Πληροφορικής στο λύκειο .....	37
1.4.2.2 Γνώσεις και δεξιότητες που πρέπει να αποκτηθούν .....	38
1.4.3 Η Πληροφορική στο κύκλο «Πληροφορικής και Υπηρεσιών» της τεχνολογικής κατεύθυνσης του λυκείου .....	39
1.4.3.1 Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον .....	40
1.4.3.1.1 Άξονες υλοποίησης του γενικού σκοπού .....	40
<b>2 Προγραμματισμός Η/Υ .....</b>	<b>43</b>
2.1 Πληροφορική, προγραμματισμός, γλώσσες προγραμματισμού .....	43
2.2 Προγραμματισμός και ψυχολογία: επίλυση προβλημάτων .....	45
2.3 Πλαίσιο ανάλυσης της δραστηριότητας του προγραμματισμού .....	47



2.4 Η μάθηση του Προγραμματισμού .....	52
2.4.1 Το σύστημα Αναπαράστασης και Επεξεργασίας.....	52
2.4.2 Τύποι γνώσεων στον προγραμματισμό.....	53
2.4.3 Εποικοδόμηση των εννοιών του προγραμματισμού.....	54
2.4.4 Τι μαθαίνει αυτός που μαθαίνει να προγραμματίζει;.....	56
2.4.5 Απαραίτητες δεξιότητες για τη μάθηση του προγραμματισμού .....	57
2.5 Η δυσκολία εκμάθησης του προγραμματισμού .....	59
2.6 Η κλασσική μέθοδος εισαγωγής στον προγραμματισμό .....	60

### **3 Εναλλακτικές Προσεγγίσεις διδασκαλίας του προγραμματισμού ..... 62**

3.1 Η μέθοδος των Μικρόκοσμων Προγραμματισμού.....	63
3.1.1 Λογισμικά δημιουργίας μικρόκοσμων.....	66
3.2 Η μέθοδος του προγραμματισμού ρομποτικών κατασκευών .....	73
3.2.1 Εκπαιδευτική Ρομποτική .....	73
3.2.2 Ο προγραμματισμός ρομποτικών κατασκευών .....	73
3.2.3 LEGO Mindstorms™: Η πιο δημοφιλής ρομποτική κατασκευή.....	74
3.2.3.1 Το λογισμικό ρομποτικής Lego Robolab .....	75

3.2.4 Τα ρομποτάκια Karel και CeeBot.....	76
<b>4 Το εκπαιδευτικό περιβάλλον προγραμματισμού</b>	
<b>Scratch .....</b>	<b>80</b>
4.1 Γενική περιγραφή του Scratch.....	80
4.2 Η αποδοχή του Scratch .....	81
4.2.1 Εκδηλώσεις-Γιορτές Scratch Day.....	82
4.3 Χαρακτηριστικά του Scratch .....	83
4.4 Αρχές εύρους και χρήσης της γλώσσας.....	84
4.5 Η Παιδαγωγική χρήση του Scratch.....	85
4.6 Σχέση με άλλα συστήματα.....	86
4.7 Περιβάλλον ανάπτυξης, ιστότοπος και Scratch Player .....	87
4.8 Το Scratchboard.....	87
4.9 Οφέλη από την αξιοποίηση του Scratch.....	89
4.9.1 Δεξιότητες διαχείρισης πληροφοριών και πολυμέσων .....	89
4.9.2 Κριτική Σκέψη .....	90
4.9.3 Αναγνώριση, διαμόρφωση και επίλυση προβλημάτων .....	90
4.9.4 Δημιουργικότητα .....	90
4.9.5 Διαπροσωπικές και συνεργατικές δεξιότητες.....	91
4.9.6 Υπευθυνότητα.....	91

<b>5</b>	<b>Η αξιολόγηση ως μέρος της διαδικασίας διδασκαλίας-μάθησης.....</b>	<b>92</b>
5.1	Οι τύποι αξιολόγησης του Bloom.....	93
5.2	Performance-Based Assessment (Αξιολόγηση επίδοσης).....	95
5.3	Rubrics (Ρούμπρικες) ως εργαλεία αξιολόγησης.....	97
5.3.1	Τύποι rubrics (ρουμπρικών).....	99
5.3.2	Πλεονεκτήματα των rubrics (ρουμπρικών) .....	101
5.3.3	Απροθυμία & Αδυναμία των εκπαιδευτικών να εισάγουν τα rubrics (ρούμπρικες) στην διαδικασία μάθησης .....	102
5.3.4	Χαρακτηριστικά υποδείγματα rubrics (ρουμπρικών).....	103
<b>6</b>	<b>Διδακτικό υλικό για τον μαθητή και τον εκπαιδευτικό.....</b>	<b>109</b>
6.1	Για μια ουσιαστική διδασκαλία του προγραμματισμού .....	109
6.2	Εργαλεία ανάπτυξης διδακτικού υλικού.....	110
6.2.1	Scratch: Ισχυρό προγραμματιστικό περιβάλλον.....	110
6.2.2	Rubrics (Ρούμπρικες): Ισχυρά εργαλεία αυτό-αξιολόγησης και ανατροφοδότησης για τη διδασκαλία του προγραμματισμού .....	111

6.2.2.1 Ερωτήσεις που θέσαμε πριν την κατασκευή των rubrics (ρουμπρίκων) .....	113
6.3 Παρουσίαση διδακτικού υλικού .....	113
6.3.1 Ο φάκελος του μαθητή .....	115
6.3.2 Ο φάκελος του εκπαιδευτικού .....	120
6.4 Συνοδευτικό CD-ROM .....	121
<b>7 Επίλογος.....</b>	<b>122</b>
7.1 Συμπεράσματα .....	122
7.2 Μελλοντική εργασία .....	123
<b>Βιβλιογραφία .....</b>	<b>125</b>

# 1 Το πλαίσιο ένταξης της Πληροφορικής στην ελληνική εκπαίδευση

---

Η ραγδαία εξέλιξη των Τεχνολογιών Πληροφορίας και Επικοινωνίας (ΤΠΕ) που χαρακτηρίζει την εποχή μας, οι νέες δυνατότητες διαχείρισης των πληροφοριών και η εξάπλωση των υπολογιστικών και δικτυακών τεχνολογιών σε όλους σχεδόν τους τομείς της ανθρώπινης δραστηριότητας καθιστούν αναπόφευκτη την εισαγωγή των υπολογιστών στην εκπαίδευση, δεδομένου ότι προσφέρουν στους σημερινούς μαθητές, στο πλαίσιο της γενικής τους εκπαίδευσης, τις απαιτούμενες δεξιότητες και τεχνολογικές γνώσεις που θα τους επιτρέψει να ανταποκριθούν στις απαιτήσεις της σύγχρονης κοινωνίας.

Στο πλαίσιο, λοιπόν, αυτό η αναγκαιότητα μιας συνολικής θεώρησης για την ένταξη των ΤΠΕ στην ελληνική σχολική πραγματικότητα έγινε κατανοητή και οδήγησε στη διατύπωση ενιαίας εκπαιδευτικής πολιτικής από το ΥΠΕΠΘ μόλις στο τέλος της δεκαετίας του 1990. Τότε διατυπώθηκε για πρώτη φορά ένα ενιαίο πλαίσιο προγράμματος σπουδών που ξεκινά από την πρωτοβάθμια εκπαίδευση και καλύπτει όλο το φάσμα της δευτεροβάθμιας εκπαίδευσης. Παράλληλα, στην ίδια χρονική περίοδο σχεδιάστηκε και το πρώτο πιλοτικό πρόγραμμα μεσαίας κλίμακας για την ένταξη των ΤΠΕ στο πρόγραμμα σπουδών των διαφόρων γνωστικών αντικειμένων. Πρόκειται για το πρόγραμμα «Οδύσσεια» που υλοποιήθηκε από το ΥΠΕΠΘ - Ινστιτούτο Τεχνολογίας Υπολογιστών (Κόμης, 2004· Χατζηλάκος, 2000· Χατζηλάκος, 2002).

Η εισαγωγή της Πληροφορικής στην ελληνική εκπαίδευση (Σχήμα 1) ξεκίνησε καταρχήν από τα τεχνικά - επαγγελματικά και τα πολυκλαδικά λύκεια κατά την περίοδο 1983-1985. Στη συνέχεια επεκτάθηκε στα γυμνάσια από το 1992, όπου και ολοκληρώθηκε μετά από μερικά χρόνια (Komis & Politis, 2001). Τέλος, προχώρησε στο γενικό λύκειο από το 1998 και ολοκληρώθηκε μετά από μερικά χρόνια. Πιο πρόσφατα επεκτάθηκε και στην πρωτοβάθμια εκπαίδευση, με την καθιέρωση ενός ενδεικτικού προγράμματος σπουδών και τον εξοπλισμό μέρους των σχολείων με υπολογιστές. Όπως φαίνεται και από το Σχήμα 1, η Πληροφορική ως αντικείμενο σπουδών ξεκίνησε πιο

νωρίς (το 1985) και κάλυψε σταδιακά όλο το φάσμα της ελληνικής εκπαίδευσης. Αντιθέτως, η Πληροφορική ως εκπαιδευτικό μέσο ξεκίνησε πολύ αργότερα (1997) και αφορά πρωτίστως τη δευτεροβάθμια εκπαίδευση και δευτερευόντως την πρωτοβάθμια εκπαίδευση.



**Σχήμα 1:** Χρονολογική εξέλιξη της πληροφορικής στην εκπαίδευση

Πρέπει να σημειωθεί, ότι το ερώτημα που αφορά το επίπεδο από το οποίο διδάσκεται η Πληροφορική, απαντήθηκε σε μεγάλο βαθμό όχι με κριτήρια παιδαγωγικού και διδακτικού προβληματισμού (Ράπτης & Ράπτη, 2002), αλλά με κριτήρια που ικανοποιούσαν κυρίως κοινωνικές και οικονομικές πιέσεις σχετικά με την *πληροφοριοποίηση* του σχολείου και του εκπαιδευτικού συστήματος γενικότερα στο πλαίσιο της λεγόμενης «Κοινωνίας της Πληροφορίας».

Σε αυτή την κατεύθυνση, και αφενός με στόχο την εκπαίδευση στην Πληροφορική και αφετέρου στα πλαίσια του πληροφορικού αλφαριθμητισμού (computer literacy), υιοθετήθηκε καταρχήν ως πιο εύκολη λύση η δημιουργία κλάδου Πληροφορικής στα τεχνικά επαγγελματικά λύκεια και στα ενιαία πολυκλαδικά λύκεια (από τα μέσα της δεκαετίας του 1980) και, στη συνέχεια, η ένταξη ενός μαθήματος Πληροφορικής στο γυμνάσιο (αρχές δεκαετίας του 1990), όπου το ωρολόγιο πρόγραμμα μπορούσε να επιβαρυνθεί σχετικά εύκολα με ένα νέο μάθημα. Αντίθετα, το λύκειο και το δημοτικό αντιμετωπίστηκαν αργότερα, κυρίως για λόγους που σχετίζονται με την επιμόρφωση των εκπαιδευτικών (Μιχαηλίδης, 2001) και το εξεταστικό σύστημα για την

εισαγωγή στα πανεπιστήμια.

Για πρώτη φορά στην ελληνική εκπαίδευση, ένα Ενιαίο Πλαίσιο Προγράμματος Σπουδών (ΕΠΠΣ) Πληροφορικής σχεδιάστηκε και ολοκληρώθηκε το Δεκέμβρη του 1997 και θεσμοθετήθηκε μέσα στο 1998. Το πλαίσιο αυτό επιδιώκει αφενός να οριοθετήσει έναν ενιαίο τρόπο θεώρησης της ένταξης των Τεχνολογιών της Πληροφορίας και των Επικοινωνιών στο ελληνικό εκπαιδευτικό σύστημα και αφετέρου να δώσει απαντήσεις με σφαιρικό τρόπο στα κύρια θέματα που αφορούν την ένταξη των ΤΠΕ σε όλο το φάσμα του ελληνικού σχολικού συστήματος (γενικό πλαίσιο, προγράμματα σπουδών, μεθοδολογία διδασκαλίας, προδιαγραφές σχολικών εργαστηρίων, κ.λπ.).

Το ΕΠΠΣ του 1997 τροποποιήθηκε μερικώς την περίοδο 2001-2003 με την εισαγωγή και τη σταδιακή εφαρμογή του Διαθεματικού Ενιαίου Πλαισίου Προγράμματος Σπουδών (ΔΕΠΠΣ), χωρίς ωστόσο να αλλάξει ούτε στη βασική του φιλοσοφία και προσανατολισμό, ούτε και ουσιαστικά ως προς τα περιεχόμενα. Με το ΔΕΠΠΣ, η Πληροφορική επεκτείνεται σε όλο το φάσμα της πρωτοβάθμιας και της δευτεροβάθμιας εκπαίδευσης. Οι αλλαγές που γίνονται στο πλαίσιο του ΔΕΠΠΣ αφορούν κυρίως στη μείωση της διδακτέας ύλης και στην αναπροσαρμογή των σχολικών βιβλίων.



**Σχήμα 2:** Πρότυπα ένταξης της Πληροφορικής στην εκπαίδευση

Η ένταξη των ΤΠΕ στα πλαίσια της πρωτοβάθμιας εκπαίδευσης, όπως προσδιορίζεται από το Ενιαίο Πλαίσιο Προγράμματος Σπουδών (Σχήμα 2), εμπνέεται από το *ολοκληρωμένο πρότυπο* ένταξης, ενώ δανείζεται (κυρίως λόγω των συνθηκών που επικρατούν στην ελληνική σχολική πραγματικότητα, όπως έλλειψη υπολογιστών στην τάξη, ελλιπής επιμόρφωση εκπαιδευτικών, και ανάπτυξη εργαστηρίου υπολογιστών στα μεγάλα δημοτικά σχολεία) ιδέες του *πραγματολογικού προτύπου* (Μακράκης & Κοντογιαννοπούλου, 1995· Κόμης, 2004).

Όσον αφορά στην εισαγωγή των ΤΠΕ στη δευτεροβάθμια εκπαίδευση, η προσέγγιση εμπνέεται, κυρίως, από το *πραγματολογικό πρότυπο* ένταξης, με εμφανή όμως και τα στοιχεία του *τεχνοκεντρικού προτύπου* (Μακράκης, 2000).

Αντιθέτως, στον κύκλο «Πληροφορικής και Υπηρεσιών» του ενιαίου λυκείου, καθώς και στον κλάδο Πληροφορικής των ΤΕΕ ακολουθείται ξεκάθαρα το *τεχνοκεντρικό πρότυπο* ένταξης των ΤΠΕ (Μακράκης, 2000).



## **1.1 Η Πληροφορική στην προσχολική ελληνική εκπαίδευση**

### **1.1.1 Το Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών (ΔΕΠΠΣ) για το νηπιαγωγείο**

Το Πλαίσιο Προγράμματος Σπουδών Πληροφορικής του 1997 (ΕΠΠΣ, 1997) δεν περιείχε πρόβλεψη για την προσχολική εκπαίδευση. Το πλαίσιο προγράμματος σπουδών εκείνης της περιόδου προέβλεπε προαιρετική εισαγωγή και χρήση της Πληροφορικής μόνο στις δύο τελευταίες τάξεις του δημοτικού. Για πρώτη φορά γίνεται πρόβλεψη ένταξης της Πληροφορικής σε όλη την πρωτοβάθμια και προ-δημοτική εκπαίδευση στο Διαθεματικό Πλαίσιο Προγράμματος Σπουδών (ΔΕΠΠΣ), το οποίο δημοσιεύτηκε το 2003 (ΔΕΠΠΣ, 2003). Το ΔΕΠΠΣ δημοσιεύτηκε στην Εφημερίδα της Κυβερνήσεως *Αρ. Φύλλου 1376, Τ. Β' 1810-2001, άρθρο 6* και εφαρμόζεται πιλοτικά σε μια σειρά από ολοήμερα νηπιαγωγεία από το σχολικό έτος 2003-2004.

Σε μεγάλο βαθμό, το πλαίσιο σπουδών Πληροφορικής του νηπιαγωγείου είναι στην ίδια κατεύθυνση με το πλαίσιο σπουδών της πρωτοβάθμιας εκπαίδευσης και, στο γενικό τους πλαίσιο, εμπνέονται από το ολοκληρωμένο πρότυπο ένταξης των ΤΠΕ στην εκπαιδευτική διαδικασία.

### **1.1.2 Η Πληροφορική στο ελληνικό νηπιαγωγείο**

Το ΔΕΠΠΣ, όπως δημοσιεύτηκε το Νοέμβριο του 2003 (ΔΕΠΠΣ, 2003), προβλέπει τον ακόλουθο σκοπό της εισαγωγής της Πληροφορικής για το νηπιαγωγείο (που είναι κοινός με αυτόν του δημοτικού, δεδομένου ότι νηπιαγωγείο και δημοτικό θεωρούνται ως ενιαία εκπαιδευτική βαθμίδα):

*«Σκοπός της εισαγωγής της Πληροφορικής στο νηπιαγωγείο και στο δημοτικό σχολείο είναι να εξοικειωθούν οι μαθητές και οι μαθήτριες με τις βασικές λειτουργίες του υπολογιστή και να έλθουν σε μια πρώτη επαφή με διάφορες χρήσεις του ως εποπτικού μέσου διδασκαλίας, ως γνωστικού - διερευνητικού εργαλείου και ως εργαλείου επικοινωνίας»*

νωνίας και αναζήτησης πληροφοριών στο πλαίσιο των καθημερινών σχολικών τους δραστηριοτήτων με τη χρήση κατάλληλου λογισμικού και, ιδιαίτερα, ανοικτού λογισμικού διερευνητικής μάθησης.» (Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής, Παιδαγωγικό Ινστιτούτο, Αθήνα, Νοέμβριος 2003).

Με άλλα λόγια, οι μαθητές του νηπιαγωγείου πρέπει αφενός να γνωρίσουν τις βασικές λειτουργίες του υπολογιστή (ο υπολογιστής ως γνωστικό αντικείμενο) και αφετέρου να τον χρησιμοποιούν σε διάφορες δραστηριότητες στο καθημερινό σχολικό τους πρόγραμμα (ο υπολογιστής ως εκπαιδευτικό εργαλείο και μέσο).

Όσον αφορά στις γνώσεις που πρέπει να αποκτήσουν τα παιδιά και στη μεθοδολογία που πρέπει να ακολουθηθεί ώστε να επιτευχθεί ο σκοπός αυτός, το ΔΕΠΠΣ προτείνει μια σειρά δραστηριοτήτων για την ανάπτυξη γνώσεων και μεθοδολογικών δεξιοτήτων, δεξιοτήτων συνεργασίας και επικοινωνίας, καθώς και στάσεων για τη θέση της επιστήμης, της τεχνολογίας και των υπολογιστών στη ζωή τους. Οι δραστηριότητες αυτές αφορούν γνώση, μεθοδολογία, συνεργασία, επικοινωνία, επιστήμη, και καθημερινή ζωή.

#### **α) Γνώση και μεθοδολογία**

Τα παιδιά ενθαρρύνονται:

- Να προσεγγίζουν ένα σύνολο βασικών απλών εννοιών που αφορούν τη γενική δομή των υπολογιστικών συστημάτων.
- Να αποκτούν στοιχειώδεις δεξιότητες και γνώσεις χειρισμού λογισμικού γενικής χρήσης, καθώς και ικανότητες μεθοδολογικού χαρακτήρα.
- Να απομυθοποιούν τον υπολογιστή και να τον χρησιμοποιούν ως εργαλείο ανακάλυψης, δημιουργίας και έκφρασης, αλλά και ως νοητικό εργαλείο και εργαλείο ανάπτυξης της σκέψης.
- Να χρησιμοποιούν εφαρμογές πολυμέσων εκπαιδευτικού περιεχομένου και να κατακτούν τις έννοιες της πλοήγησης και της αλληλεπίδρασης.

#### **β) Συνεργασία και επικοινωνία**

Τα παιδιά με τη βοήθεια της νηπιαγωγού ενθαρρύνονται:

- Να χρησιμοποιούν το λειτουργικό σύστημα, το Διαδίκτυο, το λογισμικό εφαρμογών (επεξεργασία κειμένου, ζωγραφική, εκπαιδευτικό λογισμικό, λογισμικό

πλοήγησης στο Διαδίκτυο, κ.λπ.).

- Να αναπτύσσουν δραστηριότητες στο πλαίσιο ποικίλων ομαδικών – συνθετικών εργασιών.

### γ) Επιστήμη και καθημερινή ζωή

Τα παιδιά ευαισθητοποιούνται και ενθαρρύνονται:

- Να αντιλαμβάνονται τις επιπτώσεις των νέων τεχνολογιών στους διάφορους τομείς της ανθρώπινης δραστηριότητας. (ΔΕΠΠΣ, 2003)

Απαραίτητη προϋπόθεση για την ένταξη και την ενσωμάτωση της Πληροφορικής και των ΤΠΕ στην τάξη του νηπιαγωγείου είναι η κατάλληλη διαρρύθμιση του χώρου της σχολικής τάξης ώστε ο υπολογιστής να αποτελέσει κομμάτι της καθημερινής πρακτικής και τα παιδιά να αντιληφθούν τη σημαντική συμβολή του στην πορεία της διδασκαλίας και της μάθησης.

Υπό το πρίσμα αυτό, ο υπολογιστής πρέπει να αποτελεί μέρος της τάξης και να βρίσκεται σε ένα από τα κεντρικά της σημεία και όχι σε ένα απομονωμένο σημείο της αίθουσας, έτσι ώστε όλα τα παιδιά να μπορούν να τον παρατηρούν και να διατυπώνουν σχόλια και, κατά συνέπεια, οι δραστηριότητες σε αυτή τη *γωνιά* (που αποκαλείται *γωνιά του υπολογιστή*) να είναι κοινωνικές και όχι ατομικές.

## 1.2 Η Πληροφορική στην πρωτοβάθμια ελληνική εκπαίδευση

### 1.2.1 Το ΕΠΠΣ και το ΔΕΠΠΣ για το δημοτικό

Στο ΕΠΠΣ Πληροφορικής, η πρώτη επαφή των μαθητών με τις ΤΠΕ προτείνεται να αρχίζει στο δημοτικό σχολείο, χωρίς όμως να προσδιορίζεται επακριβώς από ποια τάξη όσον αφορά τη χρήση τους στα διάφορα γνωστικά αντικείμενα. Παράλληλα στις δύο τελευταίες τάξεις του καθιερώνεται, κατεπιλογήν, εβδομαδιαία *Ωρα της Πληροφορικής*, που αφορά στο ένα τρίτο ή ένα τέταρτο του ημερήσιου σχολικού χρόνου (ΕΠΠΣ, 1997).

Με την *Ωρα της Πληροφορικής* δεν εννοείται η καθιέρωση ενός επιπλέον γνωστικού αντικειμένου στο Δημοτικό. Δεν είναι συνεπώς στόχος να εισαχθεί στο δημοτικό ένα μάθημα Πληροφορικής αντίστοιχο του γυμνασίου, ούτε πρόκειται για την απαρχή μιας εκπαίδευσης στην Πληροφορική. Το ΕΠΠΣ προτείνει να ενταχθεί η χρήση της Πληροφορικής καταρχάς στα πλαίσια της διδασκαλίας των επιστημών και της τεχνολογίας στην πρωτοβάθμια εκπαίδευση και, σταδιακά, σε όλο το εύρος του αναλυτικού προγράμματος.

Δεν πρόκειται με άλλα λόγια για ένα μάθημα που αποτελεί προαπαιτούμενο για τη χρήση του υπολογιστή, αλλά για μια ευκαιρία επαρκούς και ορθολογικής συνάντησης του παιδιού με τις ΤΠΕ, μιας συνάντησης που πρέπει να καλύπτει υπαρκτές ανάγκες της εκπαιδευτικής διαδικασίας. Η χρήση των υπολογιστών από τους μαθητές μπορεί επίσης να ενταχθεί και σε μια σειρά από δραστηριότητες που δεν εμπίπτουν στα στενά πλαίσια του αναλυτικού προγράμματος (Κόμης & Μικρόπουλος, 2001). Τέτοιου τύπου δραστηριότητες εφαρμόζονται στα πλαίσια του ολόημερου σχολείου (στη λεγόμενη ευέλικτη ζώνη - και στην ώρα της Πληροφορικής), όχι όμως πάντα με τον ενδεδειγμένο τρόπο.

Βασική επιδίωξη της *Ωρας της Πληροφορικής* είναι μια αρχική, συγκροτημένη, και σφαιρική προσέγγιση των διάφορων χρήσεων των ΤΠΕ από όλους τους μαθητές του δημοτικού σχολείου στα πλαίσια των καθημερινών σχολικών τους δραστηριοτήτων, σε μια περίοδο που μαθαίνουν «οσμωτικά» και κατά συνέπεια η εξοικειώσή τους με τον υπολογιστή γίνεται χωρίς ιδιαίτερη προσπάθεια. Με τη βοήθεια των δασκάλων τους, οι μαθητές αναπτύσσουν δραστηριότητες με τον υπολογιστή και αντιλαμβάνονται βασικές αρχές που διέπουν τη χρήση της τεχνολογίας των υπολογιστών σε σημαντικές ανθρώπινες ασχολίες: η πληροφορία και η επεξεργασία της, η επικοινωνία, η ψυχαγωγία, οι νέες δυνατότητες προσέγγισης της γνώσης (Κόμης, 2004).

### **1.2.2 Ο σκοπός της Πληροφορικής στο δημοτικό**

Σύμφωνα με το ΕΠΠΣ (ΕΠΠΣ, 1997) του Παιδαγωγικού Ινστιτούτου, ο γενικός σκοπός της εισαγωγής των ΤΠΕ στην ελληνική πρωτοβάθμια εκπαίδευση είναι:

«...οι μαθητές να χρησιμοποιούν με (ή χωρίς) τη βοήθεια του εκπαιδευτικού τον υπολογιστή ως «γνωστικό-διερευνητικό εργαλείο», να αναζητούν πληροφορίες, να επικοινωνούν, και να προσεγγίζουν βασικές αρχές που διέπουν τη χρήση της υπολογιστικής τεχνολογίας.» (Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής, Π.Ι., Δεκέμβριος 1997).

Σύμφωνα με το ΔΕΠΠΣ (ΔΕΠΠΣ, 2003) του Παιδαγωγικού Ινστιτούτου ο ειδικός σκοπός της εισαγωγής των ΤΠΕ στην ελληνική πρωτοβάθμια εκπαίδευση διαμορφώνεται ως εξής:

«Ο ειδικός σκοπός της εισαγωγής της Πληροφορικής στο δημοτικό σχολείο είναι να εξοικειωθούν οι μαθητές και οι μαθήτριες με τις βασικές λειτουργίες του υπολογιστή και να έλθουν σε μια πρώτη επαφή με διάφορες χρήσεις του ως εποπτικού μέσου διδασκαλίας, ως γνωστικού-διερευνητικού εργαλείου, και ως εργαλείου επικοινωνίας και αναζήτησης πληροφοριών στο πλαίσιο των καθημερινών σχολικών τους δραστηριοτήτων, με τη χρήση κατάλληλου λογισμικού και, ιδιαίτερα, ανοικτού λογισμικού διερευνητικής μάθησης. Σε καμία περίπτωση δε νοείται η διδασκαλία της Πληροφορικής ως διδασκαλία γνωστικού αντικειμένου (λαμβανομένου, επιπλέον, υπόψη ότι δεν διατίθεται χρόνος στο αντίστοιχο ωρολόγιο πρόγραμμα). Σκοπός είναι ο μαθητής να μαθαίνει με τη χρήση των Τεχνολογιών της Πληροφορίας και της Επικοινωνίας (ΤΠΕ) παρά για τη χρήση τους.» (Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής, Π.Ι., Νοέμβριος 2003).

Και στις δύο περιπτώσεις διαφαίνεται συνεπώς ότι η εισαγωγή των Τεχνολογιών της Πληροφορίας και των Επικοινωνιών στο δημοτικό σχολείο στοχεύει σε μια σφαιρική προσέγγιση των διαφόρων χρήσεων των ΤΠΕ από όλους τους μαθητές στα πλαίσια των καθημερινών σχολικών τους δραστηριοτήτων. Στο νέο πλαίσιο προγράμματος σπουδών (ΔΕΠΠΣ, 2003) γίνεται ρητή αναφορά ότι δεν πρόκειται για εισαγωγή ενός μαθήματος Πληροφορικής.

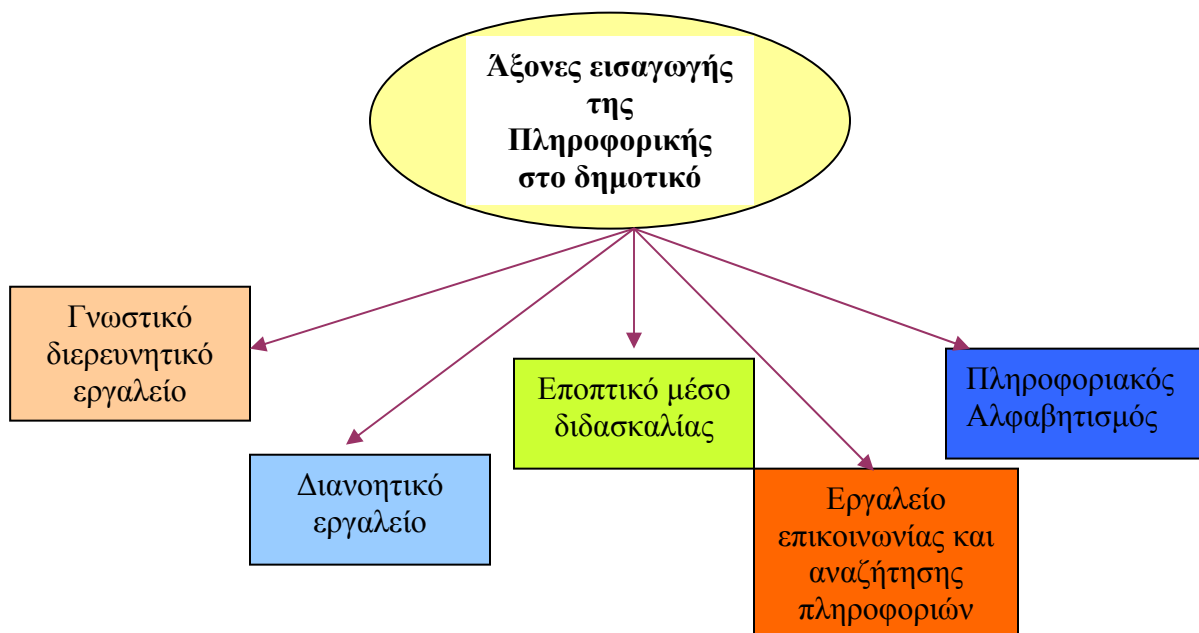
Η έμφαση δίνεται στο να εκτελέσουν οι μαθητές δραστηριότητες με τον υπολογιστή και να κατανοήσουν βασικές αρχές που διέπουν τη χρήση της υπολογιστικής τεχνολογίας σε σημαντικές ανθρώπινες ασχολίες: την **πληροφορία** και την **επεξεργασία** της, την **επικοινωνία**, την **ψυχαγωγία**, και τις **νέες δυνατότητες προσέγγισης της γνώσης**.

### 1.2.3 Άξονες υλοποίησης του σκοπού της Πληροφορικής στο δημοτικό

Για την υλοποίηση του γενικού σκοπού του προγράμματος σπουδών Πληροφορικής στο δημοτικό προτείνονται πέντε άξονες (οι οποίοι ταξινομούνται στη συνέχεια με σειρά σπουδαιότητας):

- ο υπολογιστής ως γνωστικό - διερευνητικό εργαλείο,
- ο υπολογιστής ως διανοητικό εργαλείο,
- ο υπολογιστής ως εποπτικό μέσο διδασκαλίας σε βασικά γνωστικά αντικείμενα,
- ο υπολογιστής ως εργαλείο επικοινωνίας και αναζήτησης πληροφοριών,
- ο πληροφορικός αλφαριθμητισμός.

Οι άξονες αυτοί καλύπτουν το εύρος του γενικού σκοπού, αλλά δεν είναι αναγκαίο να υλοποιηθούν στην ολότητά τους και δεν είναι δεσμευτικοί για τους εκπαιδευτικούς (ΕΠΠΣ, 1997). Ο κάθε εκπαιδευτικός επιλέγει με βάση τις γνώσεις του, την υπάρχουσα υποδομή και τις ανάγκες των μαθητών του, ποιον ή ποιους άξονες θα υλοποιήσει. Στο ΔΕΠΠΣ η έμφαση δίνεται σαφώς στους τρεις πρώτους άξονες.



Σχήμα 3: Άξονες υλοποίησης του σκοπού της Πληροφορικής στο δημοτικό

### **1.2.3.1 Ο υπολογιστής ως γνωστικό - διερευνητικό εργαλείο**

Ο υπολογιστής ως **γνωστικό - διερευνητικό εργαλείο** συνιστά τον κύριο άξονα ένταξης της Πληροφορικής και των ΤΠΕ στην ελληνική πρωτοβάθμια εκπαίδευση. Οι σύγχρονες διδακτικές και παιδαγωγικές αντιλήψεις, οι νέες θεωρήσεις της γνωστικής ψυχολογίας, καθώς και οι πρόσφατες εξελίξεις στο σχεδιασμό και την ανάπτυξη εκπαιδευτικού λογισμικού, καθιστούν απαραίτητη τη χρήση ανοικτού λογισμικού διερευνητικής μάθησης στο δημοτικό σχολείο.

Το λογισμικό αυτό μπορεί να είναι μορφής αλληλεπιδραστικών πολυμέσων, προσομοίωσης, εκπαιδευτικού παιχνιδιού, ή μοντελοποίησης και πρέπει να προσφέρει στους μαθητές τη δυνατότητα διερεύνησης πραγματικών ή φανταστικών καταστάσεων, αντίστοιχων του επιπέδου της ωριμότητάς τους, και να διευκολύνει την ανάπτυξη της δημιουργικής και ανακαλυπτικής μάθησης.

Ο υπολογιστής γίνεται μέσο για την ανάπτυξη δραστηριοτήτων από καταστάσεις που επιλέγονται από το άμεσο περιβάλλον του μαθητή και για την οργάνωση γνώσεων και δεξιοτήτων, έτσι ώστε ο μαθητής να είναι σε θέση να κατανοήσει σταδιακά τον κόσμο μέσα στον οποίο ζει και να δράσει σε αυτόν (Κόμης, 2005).

### **1.2.3.2 Ο υπολογιστής ως «διανοητικό εργαλείο» (Mindtool)**

Μια άλλη εκπαιδευτική διάσταση της πληροφορικής στην εκπαίδευση είναι ο ρόλος, που μπορεί να παίξει ο υπολογιστής ως «διανοητικό εργαλείο» (Mindtool). Τα εργαλεία αυτά έχουν γίνει πολύ δημοφιλή με τη δουλειά του Papert, συνεργάτη του Piaget, με τα γραφικά της χελώνας.

- Ο Papert θεωρεί ότι η χρήση του H/Y βάζει στα χέρια του παιδιού νέα νοητικά εργαλεία και μ' αυτό τον τρόπο μεταβάλλει τη μάθηση από αντικείμενο πειθαρχίας και ταλαιπωρίας σε αντικείμενο ανακάλυψης και ενθουσιασμού.

Πιο συγκεκριμένα, παρακολουθώντας ένα παιδί να προγραμματίζει, διαπιστώνουμε τα εξής πράγματα : Κατ' αρχήν το παιδί κατέχει το δικό του τρόπο σκέψης και επί πλέον μαθαίνει και το μηχανικό τρόπο σκέψης του υπολογιστή. Προγραμματίζοντας, τώρα, το παιδί μαθαίνει τον υπολογιστή πώς να «σκέπτεται». Αλλά

για να μάθει τη μηχανή να «σκέπτεται», πρώτα ανακαλύπτει (με κάποια υποβοήθηση από το δάσκαλο) πως το ίδιο σκέφτεται. Δηλαδή, έχουμε μια εμπλοκή δύο τρόπων σκέψης, όπου το παιδί έχει τον έλεγχο. Η προσπάθεια αυτή του παιδιού να μάθει κατ' αρχήν τους στοιχειώδεις κανόνες σκέψης του άλλου και στη συνέχεια να τον κατευθύνει κάθε φορά σε μια συγκεκριμένη διαδικασία σκέψης τον μετατρέπει σε επιστημολόγο, προσφέροντάς του μια εμπειρία που λίγοι ενήλικοι έχουν αποκτήσει.

Συμπερασματικά, θα λέγαμε πως τα παιδιά, προγραμματίζοντας στον υπολογιστή, διερευνούν διαδικασίες σκέψης και τρόπους μάθησης, έτσι ώστε να λειτουργούν, τηρουμένων των αναλογιών, ως ψυχολόγοι και επιστημολόγοι.

Διανοητικά εργαλεία ονομάζουμε επιμέρους μοντέλα γνωστικών δομών που μπορεί να γενικευθούν και μέσω των οποίων η ανακαλυπτική μάθηση μπορεί να διευκολυνθεί και να παραχθεί. Ένα διανοητικό εργαλείο, για να είναι αποτελεσματικό, πρέπει να μεσολαβεί μεταξύ του θεσμικού και πολιτισμικού πλαισίου από τη μια μεριά και της τεχνολογίας από την άλλη, μέσω της οποίας δομήθηκε, προσφέροντας έτσι, ένα φυσικό τρόπο προσέγγισης στο χρήστη.

Όπως αντιλαμβάνεται κανείς η χρήση αυτών των διανοητικών εργαλείων δεν είναι εύκολη και πλατιά, γιατί προϋποθέτει ένα ενιαίο «σύστημα» στις σχολικές τάξεις αποτελούμενο από το δάσκαλο, το μαθητή και τον υπολογιστή, στο πλαίσιο του αναλυτικού προγράμματος, που απαιτεί την ευαισθησία, δημιουργικότητα, φαντασία και αυτοπεποίθηση του δασκάλου αλλά και την ικανότητά του να προσαρμόζεται σε νέες αναδυόμενες μεθόδους διδασκαλίας με εγγυητές πάντα τις δημοκρατικές αξίες και την ελευθερία.

Όλα αυτά προϋποθέτουν, όμως, μια συνεχή επιμόρφωση των εκπαιδευτικών που διδάσκουν πληροφορική και ένα δημοκρατικό, αποκεντρωμένο εκπαιδευτικό σύστημα.

### **1.2.3.3 Ο υπολογιστής ως εποπτικό μέσο διδασκαλίας σε βασικά γνωστικά**

#### **αντικείμενα**

Ο υπολογιστής ως **εποπτικό μέσο διδασκαλίας σε βασικά γνωστικά αντικείμενα** συνιστά το τρίτο κύριο άξονα ένταξης. Η αποτελεσματική χρήση του υπολογιστή με λογισμικό ευρείας χρήσης (π.χ. ζωγραφική, επεξεργασία κειμένου,



λογιστικό φύλλο) εντάσσεται στα πλαίσια της διδασκαλίας μαθημάτων όπως η γλώσσα και η γραπτή έκφραση, τα μαθηματικά, και η δημιουργία και ανάπτυξη δεξιοτήτων στις καλλιτεχνικές και τις συλλογικές δραστηριότητες.

Έτσι (Κόμης, 2005):

- Το μάθημα γίνεται πιο κατανοητό, πιο ευχάριστο και δίνει ερεθίσματα για περισσότερη εμπάθунση – με την παρέμβαση πάντοτε του δασκάλου.
- Ο Η/Υ έχει απεριόριστη υπομονή και δεν «τραβάει ποτέ τα αυτιά των παιδιών» ούτε επηρεάζεται από υποσυνείδητες προκαταλήψεις. Το παιδί δεν έχει λόγο να τον ντρέπεται ούτε να φοβάται μην το περάσει για ηλίθιο, κάτι που οι δάσκαλοι δεν μπορούν πάντα να το αποφύγουν.
- Η ενίσχυση που δίνεται στο μαθητή από τη σωστή απάντηση είναι άμεση κι αυτό ενισχύει και την ίδια τη μάθηση.
- Επιτρέπει στο μαθητή να προχωρήσει στην εργασία του με ρυθμό ανάλογο με τις δικές του ικανότητες. Έτσι αν κάποιος μαθητής, για οποιοδήποτε λόγο, έχει διακόψει τη φοίτηση ενός μαθήματος, μπορεί να αρχίσει από εκεί που σταμάτησε την τελευταία φορά, χωρίς να δημιουργεί κενό που μπορεί να είναι καθοριστικό για την πρόοδό του. Αλλά και το αντίθετο : Αν κάποιος μαθητής τελειώσει την εργασία που του ανατέθηκε, μπορεί να συνεχίσει με άλλη εργασία ανώτερου επιπέδου και έτσι δεν υπάρχει αυτή η ισοπέδωση των ικανοτήτων και ενδιαφερόντων.
- Η ασχολία των μαθητών με τους Η/Υ επιτρέπει στο δάσκαλο να επιτηρεί όλη την τάξη και επί πλέον να καταγράφει τις αδυναμίες των μαθητών και να τα βοηθάει ατομικά. Ευνοείται έτσι η εξατομικευμένη και προγραμματισμένη διδασκαλία. Εξαιτίας αυτής της βαθμιαίας πορείας ο μαθητής ελέγχει την πρόοδό του και φθάνει σε σημείο να απαντά σχεδόν πάντα σωστά.
- Τα εκπαιδευτικά προγράμματα έχουν τη δυνατότητα να διαδίδονται πιο εύκολα ακόμα και στα πιο απομακρυσμένα χωριά. Έτσι η διάδοση της γνώσης γίνεται ευκολότερη και δημιουργούνται ίσες ευκαιρίες για κάθε παιδί του χωριού και της πόλης.
- Ο Η/Υ ως διδακτικό μέσο έχει αποδειχθεί εξαιρετικά κατάλληλο με εκπληκτικά αποτελέσματα – ιδιαίτερα σε ορισμένες τάξεις του δημοτικού –

για την πρόοδο μαθητών με μαθησιακές δυσκολίες ποικίλης αιτιολογίας (νοητική ή αισθητηριακή υστέρηση, συμπεριφορικές ιδιαιτερότητες, υπερκινητικότητα, απέχθεια προς το βιβλίο και τα μαθήματα κ.α.).

- Τα παιδιά αισθάνονται ότι η γνώση που αποκτάται με τη χρήση του υπολογιστή «ανήκει και σ' αυτά και όχι μόνο στους δασκάλους».

#### **1.2.3.4 Ο υπολογιστής ως εργαλείο επικοινωνίας και αναζήτησης**

##### **πληροφοριών**

Ο υπολογιστής ως εργαλείο επικοινωνίας και αναζήτησης πληροφοριών είναι ο τέταρτος άξονας ένταξης. Το πλαίσιο προγράμματος σπουδών συνιστά τη χρήση βάσεων δεδομένων για αναζήτηση στοιχείων και τη χρήση των δικτύων για επικοινωνία με άλλους μαθητές και για αναζήτηση πληροφοριών.

#### **1.2.3.5 Ο πληροφορικός αλφαριθμητισμός**

Ο πληροφορικός αλφαριθμητισμός αποτελεί τον τελευταίο άξονα ένταξης των ΤΠΕ στην ελληνική πρωτοβάθμια εκπαίδευση και αφορά (σύμφωνα με το ΔΕΠΠΣ) κυρίως τις δραστηριότητες που διεξάγονται στο πλαίσιο της «Ευέλικτης Ζώνης».

Το πλαίσιο σπουδών θεωρεί σε αυτό το στάδιο ως απαραίτητη την προσέγγιση των βασικών λειτουργιών του υπολογιστή-αποθήκευση πληροφοριών, επεξεργασία δεδομένων, επικοινωνία - μέσα σε μια προοπτική τεχνολογικού αλφαριθμητισμού και αναγνώρισης των δυνατοτήτων της υπολογιστικής τεχνολογίας.

Στα πλαίσια του δημοτικού, οι μαθητές εξοικειώνονται με τον πληροφορικό αλφαριθμητισμό έμμεσα και αβίαστα από τις εμπειρίες που αποκομίζουν χρησιμοποιώντας τον υπολογιστή ως εργαλείο, χωρίς να δίνεται ιδιαίτερη έμφαση στις διδακτικές ενέργειες που προϋποθέτει η υλοποίηση αυτού του άξονα.

Το ΔΕΠΠΣ υποστηρίζει ότι οι εκπαιδευτικοί πρέπει να ακολουθούν εντελώς διαφανείς διδακτικές προσεγγίσεις, εστιάζοντας δηλαδή στις σημαντικές και ουσιαστικές γνώσεις και δεξιότητες, χωρίς αναφορές σε περιττές λεπτομέρειες και συγκεκριμένα

εργαλεία.

Πρόκειται συνεπώς για εξοικείωση των μαθητών, από την πιο μικρή ηλικία, με την Πληροφορική και τις ΤΠΕ, κατά τρόπο ώστε να γίνουν ικανοί να ενεργούν στα πλαίσιά τους και να τα χρησιμοποιούν με σχετική ευχέρεια.

Η προσέγγιση αυτή εμπεριέχει δύο συμπληρωματικές πτυχές: τη διανοητική - γνωστική πτυχή, στα πλαίσια της οποίας ο μαθητής οφείλει να κατανοήσει αυτό το οποίο κάνει όταν χρησιμοποιεί πληροφορικά αντικείμενα, και την ηθική και πολιτισμική πτυχή, στα πλαίσια της οποίας είναι απαραίτητο ο μαθητής να κατανοήσει τα πληροφορικά εργαλεία μέσα από την προοπτική της κατάρτισης του αυριανού πολίτη, συνειδητού και αυτόνομου όντος σε ένα σύγχρονο κοινωνικό και τεχνολογικό περιβάλλον (Κόμης & Μικρόπουλος, 2001).

#### **1.2.4 Χρήση και ένταξη του υπολογιστή στην μαθητική κοινότητα του δημοτικού σχολείου**

Η χρήση του υπολογιστή στην εκπαιδευτική κοινότητα του δημοτικού σχολείου είναι δυνατό να βελτιώσει τη διδακτική τεχνική χρησιμοποιώντας:

- δυναμικά εποπτικά μέσα, με κινούμενη εικόνα και ήχο για τα διάφορα διδακτικά αντικείμενα.
- ποικίλα εκφραστικά μέσα για όλες τις σχολικές δραστηριότητες.
- μέσα πειραματισμού και εξερεύνησης, που επιτρέπουν την αναζήτηση και ανακάλυψη της γνώσης.
- την τεχνολογία του Διαδικτύου.

Με την ένταξη του υπολογιστή στην μαθητική ζωή και τη σωστή χρήση του:

- το μάθημα γίνεται πιο ευχάριστο και εποικοδομητικό για τους μαθητές και τους διδάσκοντες.
- οι μαθητές προοδεύουν και μαθαίνουν περισσότερα.

- οι μαθητές εμπεδώνουν καλύτερα διάφορα αντικείμενα γνώσης μέσω κατάλληλων παρουσιάσεων με τη χρήση του υπολογιστή ή λόγω της τριβής τους με κατάλληλο και επίκαιρο στο μάθημα λογισμικό.
- το μάθημα γίνεται πιο συναρπαστικό, εύκολο και αποτελεσματικό.
- οι μαθητές μαθαίνουν να συνεργάζονται, μέσω της ένταξής τους σε ομάδες εργασίας.

Όμως, οι κίνδυνοι από μια ανεξέλεγκτη εισαγωγή του υπολογιστή στο Δημοτικό είναι πολλοί, όπως για παράδειγμα:

- η δημιουργία παθητικών ανθρώπων, όταν η πρωτοβουλία περνά στη μηχανή.
- η υποβάθμιση των ανθρωπίνων σχέσεων.
- ο αποπροσανατολισμός των μαθητών.

Θα πρέπει να δοθεί ιδιαίτερη προσοχή στο γεγονός ότι σκοπός μας είναι ο υπολογιστής να χρησιμεύσει ως εργαλείο-μέσο για να πετύχουμε τους μαθησιακούς στόχους και όχι για κάθετη ειδίκευση (Δ.Ε.Ε.Π.Σ, 2001). Πρέπει να δοθεί ιδιαίτερη προσοχή διότι οι μαθητές είναι μικρής ηλικίας και εύκολα μπορεί να έχουμε τα αντίθετα αποτελέσματα. Οι διδάσκοντες πρέπει να έχουν εξειδικευτεί κατάλληλα στο αντικείμενο. Δεν αρκεί εδώ μια συνήθης εκπαίδευση σε υπολογιστές διότι η ημιμάθεια αποτελεί ένα μεγάλο κίνδυνο. Θα πρέπει επίσης να δοθεί ιδιαίτερη προσοχή και στην επιλογή του κατάλληλου εκπαιδευτικού λογισμικού καθώς και της ύλης που θα χρησιμοποιείται.

Έτσι, πρέπει ο επιμορφωμένος δάσκαλος να μπορεί να συνδυάζει τις Νέες Τεχνολογίες τηρώντας τις αρχές των κλασικών θεμάτων που άπτονται της μαθησιακής διαδικασίας (Ράπτης κ.ά., 1997) ή (Ράπτης κ.ά., 2001) ή (Piaget, 1969) ή (Piaget, 1977) ή (Papert, 1980) ή (Papert, 1993) –επίσης Bruner και Vygotsky- ή νεώτερες απόψεις και προτάσεις γύρω από αυτήν. Πρέπει να είναι ενημερωμένος γύρω από τη συμπεριφορά των μαθητών ως προς τη μάθηση (Felder, 1996) ή (Felder κ.ά., 2000), την αξιολόγηση (Κασσωτάκης κ.ά., 1986) ή (Κασσωτάκης κ.ά., 1998) ή (Δημητρόπουλος, 1998) καθώς και τις εκάστοτε επερχόμενες προτάσεις όπως είναι το νέο πλαίσιο του Παιδαγωγικού Ινστιτούτου σχετικά με την Διεπιστημονική Διδασκαλία και Μάθηση και τα νέα Προγράμματα Σπουδών για την Υποχρεωτική Εκπαίδευση (Δ.Ε.Π.Π.Σ., 2001) και γενικότερα κείμενα του ΥΠΕΠΘ (ΥΠΕΠΘ, 1998) ή (Ε.Ε.Θ., 2002).

## 1.2.5 Πιθανές χρήσεις των υπολογιστών από μαθητές δημοτικού

Στη συνέχεια αναφέρονται μερικές από τις πιθανές χρήσεις του υπολογιστή από μαθητές του Δημοτικού:

### 1. Γενική Χρήση

Ο υπολογιστής μπορεί να χρησιμοποιηθεί για:

- Εξοικείωση στη γραφή με υπολογιστή έτσι ώστε οι μαθητές να δημιουργούν αξιόλογες εργασίες ή εκθέσεις ή να εκδίδουν σχολικές εφημερίδες και περιοδικά.
- Αναζήτηση πληροφοριών στη σχολική βιβλιοθήκη για θέματα που ενδιαφέρουν τους μαθητές.
- Αναζήτηση πληροφοριών από ηλεκτρονικές εγκυκλοπαίδειες και ειδικές βάσεις δεδομένων.
- Γνωριμία με την Τεχνολογία.

### 2. Ξένες Γλώσσες

Ο υπολογιστής μπορεί να χρησιμοποιηθεί για:

- Την εύρεση λέξεων, λαθών και γενικά για τη λύση ασκήσεων με ευχάριστο τρόπο.
- Καλύτερη εκμάθηση της γλώσσας με τη χρήση βίντεο .

### 3. Μαθήματα Θετικών Επιστημών

Ο υπολογιστής μπορεί να χρησιμοποιηθεί:

- Σε επίλυση ερωτημάτων, ασκήσεων με ευχάριστο και ευφυή τρόπο.
- Σε παρουσίαση διαγραμμάτων, συγκριτικών μελετών και συνδυασμό πληροφοριών από δεδομένα που οι μαθητές επιλέγουν.
- Σε παρουσίαση πειραμάτων ή προβολές αντικειμένων -μέσα από ειδικό λογισμικό- με τη συμμετοχή των μαθητών στη διαδικασία, ώστε να υπάρχει εντονότερο ενδιαφέρον.

### 4. Μαθήματα Θεωρητικών Επιστημών

Με τη βοήθεια του υπολογιστή μπορεί να γίνει χρήση κατάλληλων, εκπαιδευτικού χαρακτήρα, εφαρμογών στην υποστηρικτική διδασκαλία των μαθημάτων.

### 5. Προγραμματισμός

*Οι μαθητές έρχονται σε επαφή με τον «προγραμματισμό» του υπολογιστή χρησιμοποιώντας κατάλληλες γλώσσες.*

### **6. Ζωγραφική**

Οι μαθητές μαθαίνουν να ζωγραφίζουν και να σχεδιάζουν με τη βοήθεια κατάλληλων εφαρμογών

### **7. Μουσική**

Οι μαθητές μπορούν να εισαχθούν στο μαγικό κόσμο της μουσικής όχι μόνο ως ακροατές ή μαθητευόμενοι, αλλά και ως συνθέτες μουσικών φράσεων

### **8. Επικοινωνία.**

Οι μαθητές, μέσα από τις Υπηρεσίες του Διαδικτύου επικοινωνούν, και ανταλλάσσουν ιδέες και εργασίες.

Όσον αφορά δε το λογισμικό, μπορούμε να δούμε τις ακόλουθες κατηγορίες:

#### **A. Το Λογισμικό γενικής χρησιμότητας**

Σ' αυτή την κατηγορία λογισμικού ανήκουν:

- τα παραθυρικά περιβάλλοντα.
- λογισμικό πλοήγησης στο Διαδίκτυο.
- οι επεξεργαστές κειμένου.
- τα προγράμματα ζωγραφικής.
- λεξικά, εγκυκλοπαίδειες, κ.ά..

Όλα αυτά μπορούν να λειτουργήσουν ως εργαλεία ενδιάμεσα σε μαθήματα και κατάλληλα χρησιμοποιούμενα να χρησιμεύσουν σε ενότητες όπως:

- γράφω
- ζωγραφίζω
- επικοινωνώ και ερευνώ
- γνωρίζω την Τεχνολογία
- δημιουργώ, προγραμματίζοντας.

## **B. Το Εκπαιδευτικό Λογισμικό**

Το λογισμικό αυτής της κατηγορίας βοηθά στην διδασκαλία ενός μαθήματος, ακόμα και στην διδασκαλία των αντικειμένων του υπολογιστή. Εδώ όμως πρέπει να τονιστεί ότι η δημιουργία και η αξιολόγηση εκπαιδευτικού λογισμικού είναι πολύ σύνθετη εργασία. Πρέπει να τονιστεί η διαφοροποίηση του απλού εκπαιδευτικού λογισμικού από το έξυπνο. Το έξυπνο εκπαιδευτικό λογισμικό προσαρμόζεται στις ιδιαιτερότητες του μαθητή και στις μαθησιακές του δυνατότητες. Για παράδειγμα, αφήνει τον μαθητή να λύσει μια άσκηση και παρεμβαίνει αυτόματα για να παρέχει βοήθεια, αφού πρώτα διαγνώσει την ύπαρξη προβλήματος (Βίρβου, Καμπάση, 2003). Μόνο έξυπνα ή έμπειρα συστήματα μπορούν να επιτελέσουν τέτοιες λειτουργίες και το κόστος ανάπτυξης τους είναι απαγορευτικό, ειδικά για τα ελληνικά δεδομένα.

### **1.3 Η Πληροφορική στο Γυμνάσιο**

Το ελληνικό γυμνάσιο υπήρξε η πρώτη, μετά τους κλάδους Πληροφορικής των τεχνικών λυκείων και των τεχνικών επαγγελματικών σχολών, σχολική βαθμίδα μαζικής εισαγωγής ενός αυτοδύναμου μαθήματος Πληροφορικής στην ελληνική προ-πανεπιστημιακή εκπαίδευση. Η βαθμιαία εισαγωγή άρχισε το 1992 και ολοκληρώθηκε σταδιακά την επόμενη δεκαετία. Σήμερα, το σύνολο σχεδόν των ελληνικών γυμνασίων διαθέτει σχολικό εργαστήριο Πληροφορικής (συνήθως 10 με 20 υπολογιστές, κατά κανόνα σε περιβάλλον Microsoft Windows).

Το μάθημα Πληροφορικής διδάσκεται σε όλες τις τάξεις μία ώρα εβδομαδιαίως από καθηγητές Πληροφορικής των κλάδων ΠΕ19 (απόφοιτοι ΑΕΙ) και ΠΕ20 (απόφοιτοι ΤΕΙ).

#### **1.3.1 Η Πληροφορική ως γνωστικό αντικείμενο στο ελληνικό γυμνάσιο**

Ο γενικός σκοπός του μαθήματος Πληροφορικής στο γυμνάσιο, σύμφωνα με το ενιαίο πλαίσιο προγράμματος σπουδών του 1997 (ΕΠΠΣ, 1997) και το διαθεματικό

πλαίσιο προγράμματος σπουδών του 2003 (ΔΕΠΠΣ, 2003), προσδιορίζει και οριοθετεί μια συνολική θεώρηση της Πληροφορικής σε αυτό το επίπεδο εκπαίδευσης τόσο με όρους γνώσεων (βασικές έννοιες και όροι της επιστήμης της Πληροφορικής), όσο και με όρους ανάπτυξης δεξιοτήτων (απόκτηση αυτονομίας χειρισμού ενός απλού υπολογιστικού συστήματος).

Παράλληλα, ο σκοπός προσεγγίζει το ζήτημα της ευαισθητοποίησης στις κοινωνικές επιπτώσεις των ΤΠΕ (με έμφαση στην καλλιέργεια στάσεων και αξιών). Ο σκοπός φαίνεται να υιοθετεί την άποψη που υποστηρίζει ότι η σύγχρονη γενική κουλτούρα οφείλει να έχει ένα ισχυρό τεχνικό και επιστημονικό συστατικό. Σε αυτό το συστατικό, η Πληροφορική έχει τη δική της θέση:

*«Ειδικός σκοπός του μαθήματος της Πληροφορικής στο γυμνάσιο είναι να δώσει στους μαθητές όλα τα απαιτούμενα εφόδια ώστε να εντρυφήσουν στις βασικές έννοιες και όρους της Τεχνολογίας της Πληροφορίας και της Επικοινωνίας (ΤΠΕ), δηλαδή των μέσων και των τεχνικών που χρησιμοποιούνται για την επεξεργασία, τη μετάδοση, και τη λήψη κάθε πληροφορίας που μπορεί να παρουσιαστεί σε ψηφιακή μορφή. Να προσεγγίσουν το σύνολο των βασικών απλών εννοιών που αφορούν τη γενική δομή των υπολογιστικών συστημάτων και τις διαχρονικές αρχές που τα διέπουν (αρχιτεκτονική υπολογιστών, διαφορετικότητα υπολογιστικών συστημάτων, πρόγραμμα, οργάνωση και διαχείριση αρχείων κ.λπ.).*

*Να αποκτήσουν τις απαραίτητες δεξιότητες χειρισμού και κριτικής επεξεργασίας, καθώς και δεξιότητες μεθοδολογικού χαρακτήρα, ασκούμενοι σε ένα σύστημα υπολογιστών και στα βασικά εργαλεία που το συνοδεύουν. Να αποκτήσουν γνώσεις σχετικά με τη διαδικασία επίλυσης απλών προβλημάτων με τη χρήση του υπολογιστή. Να διαπιστώσουν και να αντιληφθούν ότι μια απλή μηχανή ελέγχεται και προγραμματίζεται από τον άνθρωπο.*

*Να χρησιμοποιήσουν εφαρμογές πολυμέσων, να κατακτήσουν τις έννοιες της πλοήγησης και της αλληλεπίδρασης, να περιηγηθούν στο Διαδίκτυο, να εκπαιδευτούν στη χρήση κατάλληλου λογισμικού ώστε να αξιοποιήσουν τον υπολογιστή - αρχικά στο πλαίσιο διαφόρων μαθημάτων τους, αλλά και στις μετέπειτα δραστηριότητές τους. Να ανακαλύψουν, να επιλέξουν, να αναλύσουν, και να αξιολογήσουν πληροφορίες για να τις αξιοποιήσουν στις εκπαιδευτικές τους δραστηριότητες αλλά και στην καθημερινή τους ζωή γενικότερα.*



*Να αναπτύξουν κώδικες δεοντολογίας στο πλαίσιο της συνεργασίας με άλλους, του σεβασμού της εργασίας τους και της διαφορετικότητάς τους. Να γνωρίσουν και να κρίνουν τις τρέχουσες και τις μελλοντικές επιπτώσεις των ΤΠΕ σε ατομικό και κοινωνικό επίπεδο, αλλά και στους διάφορους τομείς της ανθρώπινης δραστηριότητας». (Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής, Π.Ι., Νοέμβριος 2003).*

### **1.3.2 Βασικοί άξονες για τη διδασκαλία της Πληροφορικής στο γυμνάσιο.**

Σύμφωνα με το ΔΕΠΠΣ, η επίτευξη του γενικού σκοπού αναφέρεται ότι μπορεί να επιτευχθεί με την προσέγγιση των εννοιών της Πληροφορικής με βάση τους παρακάτω τέσσερις άξονες:

**Γνωρίζω-επικοινωνώ με τον υπολογιστή:** Γνωριμία με τα βασικά στοιχεία ενός υπολογιστή (στοιχεία αρχιτεκτονικής, υπολογιστικά συστήματα, πρόγραμμα, διαχείριση αρχείων) (Γρηγοριάδου, 2003· ΔΕΠΠΣ, 2003 ).

**Διερευνώ-δημιουργώ-ανακαλύπτω:** Χρήση ενός βασικού λειτουργικού συστήματος, λογισμικού ευρείας χρήσης (επεξεργασία κειμένου, λογιστικά φύλλα, ζωγραφική, πλοήγηση στο Διαδίκτυο) και κατάλληλου εκπαιδευτικού λογισμικού (γλώσσα προγραμματισμού Logo). Η χρήση αυτών των μορφών λογισμικού προβλέπεται να γίνεται μέσα από την ανάπτυξη μεγάλου αριθμού συνθετικών εργασιών ώστε οι μαθητές να μπορούν να αναγνωρίζουν τα χαρακτηριστικά των διάφορων κατηγοριών λογισμικού (Γρηγοριάδου, 2003· Κόμης & Μικρόπουλος, 2001· ΕΠΠΣ, 1997).

**Ελέγχο-προγραμματίζω τον υπολογιστή:** Επίλυση απλών προβλημάτων σε υπολογιστικό περιβάλλον. Κατανόηση εννοιών του αλγορίθμου, του κύκλου ανάπτυξης ενός προγράμματος, του περιβάλλοντος μιας γλώσσας προγραμματισμού και των βασικών δομών της (ΕΠΠΣ, 1997).

**Ο υπολογιστής στη ζωή μας:** Ευαισθητοποίηση των μαθητών στις επιπτώσεις των τεχνολογιών της Πληροφορίας και της Επικοινωνίας στους διάφορους τομείς της ανθρώπινης δραστηριότητας όπως και σε θέματα ηθικής (προστασία ανθρώπινων

δικαιωμάτων, συμπεριφορά στο Διαδίκτυο, ασφάλεια της πληροφορίας) (Κόμης & Μικρόπουλος, 2001).

### **1.3.3 Γνώσεις & δεξιότητες μαθητών Γυμνασίου**

Οι γνώσεις και οι δεξιότητες οι οποίες θα πρέπει να καλλιεργηθούν στο Γυμνάσιο στους μαθητές αναφέρονται παρακάτω (Κόμης, 2004):

- Κατανόηση βασικών εννοιών ή όρων Πληροφορικής (δεδομένα, πληροφορίες, κωδικοποίηση, πρόγραμμα, επεξεργασία δεδομένων, αρχείο, αποθήκευση, λογισμικό συστήματος κλπ).
- Περιγραφή της λειτουργίας των κυριότερων μονάδων του υπολογιστή.
- Κατανόηση βασικών εννοιών και όρων των δικτυακών τεχνολογιών και των τεχνολογιών πολυμέσων.
- Άνεση στη χρήση βασικών γραφικών περιβαλλόντων βασικών προγραμμάτων εφαρμογών και του Διαδικτύου.
- Επίλυση απλών προβλημάτων σε υπολογιστικό περιβάλλον.
- Συζήτηση και ενημέρωση στις τεχνολογικές εξελίξεις και στις επιπτώσεις του στους διάφορους τομείς της ανθρώπινης δραστηριότητας.

### **1.3.4 Η διδασκαλία του προγραμματισμού στο Γυμνάσιο**

Θα μπορούσαμε να ισχυριστούμε ότι η διδασκαλία του Προγραμματισμού στην Γ΄ Γυμνασίου έχει τύχει μιας πολύ πρόχειρης αντιμετώπισης από το ΠΙ. Αντιγράφουμε από τις πιο πρόσφατες οδηγίες: *«Βασικός στόχος της ενότητας αυτής είναι η ανάπτυξη της ικανότητας των μαθητών-μαθητριών στην αλγοριθμική προσέγγιση και επίλυση πολύ απλών προβλημάτων. Το προγραμματιστικό περιβάλλον χρησιμοποιείται ως μέσο για την επίτευξη αυτού του στόχου. Μέσα από την υλοποίηση των αλγορίθμων οι μαθητές-μαθήτριες γνωρίζουν ότι είναι απαραίτητο από τα χαρακτηριστικά του προγραμματιστικού περιβάλλοντος, εξοικειώνονται μόνο με τα βασικά και απαραίτητα δομικά του στοιχεία και τις εντελώς απαραίτητες έννοιες της γλώσσας Προγραμματισμού που θα χρησιμοποιηθεί. Ο Προγραμματισμός δεν είναι αυτοσκοπός».*

Και μόνο το γεγονός ότι όλα τα παραπάνω θα πρέπει να υλοποιηθούν μέσα σε 10-15 διδακτικές ώρες (ακόμη και η διάρκεια δεν είναι ξεκάθαρη), αρκεί για να καταδείξει τη ματαιοπονία του εγχειρήματος. Ας εξετάσουμε ωστόσο λεπτομερέστερα τι ακριβώς ζητά το ΠΙ.

Η προτεινόμενη οπτική γωνία για την εισαγωγή στον Προγραμματισμό είναι ελάχιστα συμβατή, τόσο με την ηλικία, όσο κυρίως με το γνωστικό υπόβαθρο των μαθητών της Γ΄ τάξης. Σε κανένα από τα μαθήματα που έχουν διδαχθεί έως τη στιγμή αυτή δεν έχουν ακούσει την έννοια του αλγορίθμου, ασχέτως εάν χρησιμοποιούν ήδη αρκετούς από το δημοτικό σχολείο. Τούτο δεν αποτελεί οξύμωρο σχήμα, αλλά οφείλεται στο γεγονός της αδυναμίας να ορισθούν και να γίνουν κατανοητές έννοιες τόσο αφηρημένες, όσο η έννοια του αλγορίθμου.

Η ίδια η επίλυση προβλημάτων είναι -δυστυχώς- κάτι το οποίο ελάχιστα έχει διδαχθεί ο μαθητής, μια και σε μαθήματα όπως τα μαθηματικά και η φυσική έχει εθιστεί στην επίλυση τυποποιημένων ασκήσεων μάλλον, όπου απλά εφαρμόζει κάποιους τύπους, παρά στην επίλυση προβλημάτων, όπου θα έπρεπε να αναζητήσει ο ίδιος τους τύπους που πρέπει να χρησιμοποιηθούν. Είναι, λοιπόν, υπερβολικά μεγαλόπνοος ο στόχος της εκμάθησης επίλυσης προβλημάτων με αλγοριθμική προσέγγιση σε αυτό το στάδιο και σε τόσο ελάχιστο χρόνο.

Εξαιρετικά δύσκολος στην εφαρμογή του δείχνει και ο διττός στόχος μέσα σε λίγα σαραντάλεπτα να διδαχθούν τόσο τα «απαραίτητα χαρακτηριστικά του προγραμματιστικού περιβάλλοντος», όσο και τα «βασικά δομικά στοιχεία και απαραίτητες έννοιες της γλώσσας Προγραμματισμού». Ο διδάσκων καλείται κατ' αρχάς να επιλέξει ποια ακριβώς είναι τα «απαραίτητα» και τα «βασικά» και από την άλλη μεριά να ισορροπήσει ανάμεσα στην εξήγηση του περιβάλλοντος εργασίας προς τους μαθητές και την ίδια τη διδασκαλία του αντικειμένου του Προγραμματισμού.

Η δε κατακλείδα των οδηγιών πως «ο Προγραμματισμός δεν είναι αυτοσκοπός» έρχεται να ολοκληρώσει τη νεφελώδη εικόνα που καλείται να αποκρυπτογραφήσει ο διδάσκων, λαμβάνοντας τις αποφάσεις για το τι ακριβώς πρέπει να διδαχθεί, πως και γιατί (Γεωργιάδης, 2006).

## **1.4 Η Πληροφορική στο Ενιαίο Λύκειο**

### **1.4.1 Η Πληροφορική ως μάθημα επιλογής και κύκλος σπουδών**

Με τη θέσπιση του ενιαίου λυκείου (το οποίο τέθηκε σε λειτουργία από το σχολικό έτος 1998-1999 σύμφωνα με το Νόμο 2525/97 [ΦΕΚ 188 ΑΙ]) η Πληροφορική εντάσσεται ως μάθημα γενικής παιδείας (με τη μορφή μαθήματος επιλογής) και στις τρεις τάξεις (Α', Β', και Γ' Λυκείου) και ως κύκλος μαθημάτων (υποχρεωτικά και επιλογής) της τεχνολογικής κατεύθυνσης στη Γ' Λυκείου (ΕΠΠΣ, 1997).

Οι μαθητές του ενιαίου λυκείου μπορούν να επιλέξουν στην Α' Λυκείου το κατεπιλογήν μάθημα της Πληροφορικής, ενώ μπορούν να το επιλέξουν ξανά στη Β' ή στη Γ' Λυκείου. Συχνά όμως, η επιλογή του μαθήματος εξαρτάται και από εξωγενείς παράγοντες, όπως η ύπαρξη και η διαθεσιμότητα εργαστηρίου Πληροφορικής στο σχολείο, καθώς και η ώρα που τοποθετείται το μάθημα μέσα στο ωρολόγιο πρόγραμμα (συνήθως προς το τέλος του ημερήσιου προγράμματος).

Παράλληλα, ένας από τους κύκλους μαθημάτων της τεχνολογικής κατεύθυνσης του ενιαίου λυκείου αφορά στην Πληροφορική. Πιο συγκεκριμένα, στην τεχνολογική κατεύθυνση του ενιαίου λυκείου καθιερώνεται κύκλος «Πληροφορικής και Υπηρεσιών» που περιλαμβάνει πέντε μαθήματα Πληροφορικής και οδηγεί στην πανεπιστημιακή εκπαίδευση των θετικών και των τεχνολογικών επιστημών.

Από τα πέντε μαθήματα του κύκλου ένα είναι υποχρεωτικό και τέσσερα επιλογής, τα οποία καλύπτουν μεγάλο εύρος του γνωστικού αντικείμενου της Πληροφορικής. Στις πανελλήνιες εξετάσεις που καθορίζουν την εισαγωγή στην τριτοβάθμια εκπαίδευση εξετάζεται σε πανελλήνιο επίπεδο μόνο το υποχρεωτικό μάθημα, με τίτλο «Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον».

### **1.4.2 Η Πληροφορική ως μάθημα γενικής παιδείας στο λύκειο**

Είναι αξιοσημείωτο το γεγονός ότι το μάθημα Πληροφορικής στο ενιαίο λύκειο που αφορά όλους τους μαθητές (Εφαρμογές Πληροφορικής και Υπολογιστών) δεν θεσμοθετείται ως υποχρεωτικό αλλά ως μάθημα επιλογής. Στην Α' Λυκείου, καθώς και στο μάθημα κορμού της Β' ή Γ' Λυκείου, οι μαθητές συναντούν για τελευταία φορά την

Πληροφορική και τις ΤΠΕ στα πλαίσια ενός γνωστικού αντικειμένου που δεν σχετίζεται όμως άμεσα με το μελλοντικό επαγγελματικό τους προσανατολισμό (Κόμης, 2001).

Κάτω από το πρίσμα αυτό, η Πληροφορική ως μάθημα γενικής παιδείας του ενιαίου λυκείου δεν έχει ως σκοπό την επαγγελματική κατάρτιση των μαθητών στα επαγγέλματα της Πληροφορικής, αλλά τη συνέχιση και εμπάθυνση των γνώσεων που έχουν αποκτηθεί στις προηγούμενες βαθμίδες της εκπαίδευσης καθώς και την προσαρμογή τους στις νέες εξελίξεις των ΤΠΕ.

Η σφαιρική αυτή εκπαίδευση στη χρήση και στις βασικές και διαχρονικές έννοιες της Πληροφορικής θα μπορούσε να διασφαλίσει με τον πιο ευκρινή τρόπο την αποτελεσματικότερη διάχυση των ΤΠΕ σε όλο το φάσμα του αναλυτικού προγράμματος (Κόμης & Μικρόπουλος, 2001).

Στο ενιαίο λύκειο δεν έχουν επέλθει αλλαγές στο πρόγραμμα σπουδών Πληροφορικής μετά την καθιέρωσή του το 1998 - κάτι που έγινε στο γυμνάσιο με το ΔΕΠΠΣ. Στην ενότητα αυτή θα μελετήσουμε πρωτίστως την Πληροφορική ως μάθημα γενικής παιδείας στο ενιαίο λύκειο. Στη συνέχεια, θα ασχοληθούμε με τα μαθήματα του «κύκλου πληροφορική και υπηρεσιών» του λυκείου.

#### **1.4.2.1 Γενικός σκοπός της Πληροφορικής στο λύκειο**

Σύμφωνα με το ΕΠΠΣ (ΕΠΠΣ, 1997), ο γενικός σκοπός του μαθήματος Πληροφορικής στο ενιαίο λύκειο είναι ο ακόλουθος:

*«Τα μαθήματα επιλογής Εφαρμογές Πληροφορικής και Εφαρμογές Υπολογιστών εντάσσονται στο ωρολόγιο πρόγραμμα, των Α' και Β' / Γ' τάξεων αντίστοιχα, του ενιαίου λυκείου και έχουν γενικό σκοπό:*

*την επέκταση της γενικής πληροφορικής παιδείας των μαθητών με έμφαση στην ανάπτυξη ικανοτήτων και δεξιοτήτων στη χρήση και αξιοποίηση των υπολογιστικών και δικτυακών τεχνολογιών ως εργαλείων μάθησης και σκέψης*

*την ενημέρωση των μαθητών για τις εφαρμογές της Πληροφορικής στο σύγχρονο κόσμο και, ειδικότερα, για τις δυνατότητες που προσφέρει και τις προοπτικές που δημιουργεί στον κλάδο / κατεύθυνση που επέλεξαν (ή πρόκειται να επιλέξουν) για να σπουδάσουν*

*την ευαισθητοποίηση, τον προβληματισμό, και την ανάπτυξη κριτικής ικανότητας εκ μέρους των μαθητών στα κοινωνικά, ηθικά, πολιτισμικά, κ.ά. ζητήματα που τίθενται με την «εισβολή» των υπολογιστικών και δικτυακών τεχνολογιών σε όλους τους τομείς της ανθρώπινης δραστηριότητας.» (Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής, Π.Ι, Δεκέμβριος 1997).*

Στο πλαίσιο αυτό, οι μαθητές που τελειώνουν το ενιαίο λύκειο πρέπει να αποκτήσουν συγκροτημένη γνώση στις τεχνολογίες της Πληροφορικής, να αναπτύξουν κλίσεις και δεξιότητες σε αυτές, και να είναι σε θέση να τις αξιοποιήσουν στη συνέχεια των σπουδών τους ή στην επαγγελματική τους δραστηριότητα. Υπό το πρίσμα αυτό, ο σκοπός των μαθημάτων Πληροφορικής γενικής παιδείας του ενιαίου λυκείου ακολουθεί τόσο το πραγματολογικό, όσο και το τεχνοκεντρικό πρότυπο εισαγωγής των υπολογιστών στην εκπαίδευση.

#### **1.4.2.2 Γνώσεις και δεξιότητες που πρέπει να αποκτηθούν**

Σύμφωνα με το ΕΠΠΣ (ΕΠΠΣ, 1997), οι μαθητές που θα έχουν παρακολουθήσει με επιτυχία τα μαθήματα Εφαρμογές Πληροφορικής και Εφαρμογές Υπολογιστών στο ενιαίο λύκειο πρέπει:

- ✚ Να μπορούν να περιγράφουν την έννοια, το σκοπό, και τα στάδια ανάπτυξης των πληροφοριακών συστημάτων.
- ✚ Να μπορούν να διακρίνουν και να αναγνωρίζουν τις εφαρμογές της πληροφορικής στο σύγχρονο κόσμο.
- ✚ Να γνωρίζουν τις βασικές κατηγορίες υπολογιστικών συστημάτων και να περιγράφουν τα βασικά χαρακτηριστικά της λειτουργίας και των δυνατοτήτων τους.
- ✚ Να μπορούν να επιλέγουν το κατάλληλο λογισμικό κάθε φορά που θα το χρειάζονται.
- ✚ Να μπορούν να περιγράφουν τα χαρακτηριστικά και τις δυνατότητες των σύγχρονων προγραμματιστικών εργαλείων.
- ✚ Να μπορούν να επιλύουν απλά προβλήματα με χρήση προγραμματιστικών εργαλείων.

- ✚ Να μπορούν να αναπτύσσουν απλές εφαρμογές πολυμέσων.
- ✚ Να κατανοούν και να μπορούν να εξηγήσουν βασικές έννοιες και όρους της σύγχρονης δικτυακής τεχνολογίας.
- ✚ Να μπορούν να χρησιμοποιούν τις υπηρεσίες του Internet και να δημιουργούν τις δικές τους σελίδες στον Παγκόσμιο Ιστό πληροφοριών.
- ✚ Να μπορούν να κρίνουν τις επιπτώσεις της Πληροφορικής στη ζωή των ανθρώπων.
- ✚ Να έχουν αποκτήσει επαρκή εικόνα για τις εφαρμογές και τις δυνατότητες που προσφέρουν οι υπολογιστικές και δικτυακές τεχνολογίες στην κατεύθυνση /κλάδο που επέλεξαν για να σπουδάσουν (Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορική, Δεκέμβριος 1997).

### **1.4.3 Η Πληροφορική στο κύκλο «Πληροφορικής και Υπηρεσιών» της τεχνολογικής κατεύθυνσης του λυκείου**

Όπως έχουμε ήδη αναφέρει στην αρχή του κεφαλαίου, εκτός από μάθημα γενικής παιδείας, η Πληροφορική καθιερώνεται επίσης στο ενιαίο λύκειο ως πεδίο βασικών γνώσεων και κλάδου ειδίκευσης.

Στο πλαίσιο αυτό και με βάση το νόμο για το ενιαίο λύκειο (2525/97 [ΦΕΚ 188 ΑΙ]), στην τεχνολογική κατεύθυνση του ενιαίου λυκείου καθιερώνεται κύκλος «Πληροφορικής και Υπηρεσιών» που περιλαμβάνει πέντε μαθήματα Πληροφορικής. Τόσο ο τύπος, όσο και η διάρκεια αυτών των μαθημάτων έχουν μάλλον ενδεικτική μορφή αφού έχουν προκύψει κάποιες αλλαγές κατά τα χρόνια εφαρμογής του προγράμματος αυτού. Ως βασικό μάθημα του κύκλου θεωρείται πάντα το μάθημα με τίτλο «Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον», το οποίο εξετάζεται και πανελλαδικά για την εισαγωγή στο Πανεπιστήμιο.

Σύμφωνα με το ΕΠΠΣ (ΕΠΠΣ, 1997), τα μαθήματα Πληροφορικής του κύκλου Πληροφορικής και Υπηρεσιών της τεχνολογικής κατεύθυνσης του ενιαίου λυκείου:

*«έχουν ως γενικό σκοπό να δώσουν στους μαθητές όλα τα απαιτούμενα εφόδια ώστε να είναι ικανοί να ανταποκριθούν στις απαιτήσεις της τριτοβάθμιας εκπαίδευσης, στις απαιτήσεις της μετά-δευτεροβάθμιας επαγγελματικής εκπαίδευσης και κατάρτισης, στο*

ρόλο τους ως ενεργών πολιτών στην κοινωνία των πληροφοριών και στη διαμορφούμενη κοινωνία της μάθησης.

*Με τα μαθήματα αυτά, το πρόγραμμα σπουδών του ενιαίου λυκείου δίνει τη δυνατότητα στους μαθητές να ανιχνεύσουν και να καλλιεργήσουν τις κλίσεις και τα ταλέντα τους στις υπολογιστικές και δικτυακές τεχνολογίες, ανταποκρινόμενο έτσι στη διαρκώς αυξανόμενη κοινωνική απαίτηση για επαρκή εκπαίδευση στις τεχνολογίες αιχμής.»*

Υπό το πρίσμα αυτό, ο σκοπός των μαθημάτων Πληροφορικής της τεχνολογικής κατεύθυνσης εντάσσεται στα πλαίσια του τεχνοκεντρικού προτύπου εισαγωγής των υπολογιστών στην εκπαίδευση.

#### **1.4.3.1 Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον**

Το μάθημα «Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον» συνιστά το βασικό μάθημα του κύκλου Πληροφορικής και Υπηρεσιών του ενιαίου λυκείου. Ο γενικός του σκοπός περιγράφεται ως εξής:

*«Ο γενικός σκοπός του μαθήματος είναι να αναπτύξουν οι μαθητές αναλυτική και συνθετική σκέψη, να αποκτήσουν ικανότητες μεθοδολογικού χαρακτήρα, και να μπορούν να επιλύουν απλά προβλήματα σε προγραμματιστικό περιβάλλον.»*

##### **1.4.3.1.1 Άξονες υλοποίησης του γενικού σκοπού**

Η προσέγγιση των εννοιών και η καλλιέργεια δεξιοτήτων που απαιτούνται για την υλοποίηση του γενικού σκοπού του μαθήματος «Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον» ταξινομούνται σε τρεις άξονες:

**Ανάλυση-Σχεδίαση:** Οι μαθητές κατανοούν το πρόβλημα, το αναλύουν, προσεγγίζουν με αυστηρότητα την έννοια του αλγορίθμου και περιγράφουν την αλγοριθμική διαδικασία επίλυσής του.

**Υλοποίηση σε προγραμματιστικό περιβάλλον:** Οι μαθητές μαθαίνουν να χρησιμοποιούν προγραμματιστικά εργαλεία, να εφαρμόζουν προγραμματιστικές τεχνικές, να γράφουν το πρόγραμμα, να το εκτελούν, να το διορθώνουν, και να το βελτιώνουν.

**Τεκμηρίωση-Αξιολόγηση:** Οι μαθητές τεκμηριώνουν την εργασία τους και αξιο-



λογούν την ποιότητά της.

Στον επόμενο πίνακα (Πίνακας 1) περιγράφονται συνοπτικά οι κύριες ενότητες και το περιεχόμενό τους (οι ώρες διδασκαλίας προτείνονται ενδεικτικά αλλά δεν επιτρέπεται να ξεπεραστεί το σύνολο των 75 διδακτικών ωρών).

Ενότητα	Περιεχόμενο
<b>1. Ανάλυση του προβλήματος</b>	Καθορισμός και κατανόηση του προβλήματος. Μεθοδολογίες ανάλυσης. Καθορισμός απαιτήσεων. Κύκλος ζωής λογισμικού. <b><u>Διδακτικές ώρες: 15</u></b>
<b>2. Σχεδίαση του αλγορίθμου</b>	Προδιαγραφές σχεδιασμού. Μεθοδολογίες σχεδιασμού. Σύγχρονες τεχνικές σχεδίασης προγραμμάτων. Αλγοριθμική γλώσσα. Ανάπτυξη αλγορίθμου. Έλεγχος αλγορίθμου. Σχεδιασμός περιβάλλοντος διεπαφής (διασύνδεσης). <b><u>Διδακτικές ώρες: 24</u></b>
<b>3. Υλοποίηση σε προγραμματιστικό περιβάλλον</b>	Δομημένος προγραμματισμός. Αντικειμενοστραφής προγραμματισμός. Δομικά στοιχεία προγραμματισμού. Σύγχρονα προγραμματιστικά εργαλεία.

	Αποσφαλμάτωση προγράμματος. Επικοινωνία με άλλες εφαρμογές. <u><b>Διδακτικές ώρες: 30</b></u>
<b>4. Τεκμηρίωση - Αξιολόγηση</b>	Τεκμηρίωση του προγράμματος. Αξιολόγηση της απόδοσης του προγράμματος. Περιγραφή πιθανών εναλλακτικών λύσεων. Δυνατότητες επέκτασης και όρια χρήσης. <u><b>Διδακτικές ώρες: 6</b></u>

**Πίνακας 1:** Άξονες περιεχομένου «Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον».

Στο σημείο αυτό είναι σκόπιμο να παρατηρηθεί ότι το γενικό περίγραμμα του μαθήματος δίνει έμφαση στην ανάλυση και στη σχεδίαση, ενώ η υλοποίηση του προγράμματος δεν σχετίζεται με κάποιο συγκεκριμένο προγραμματιστικό περιβάλλον.

Οι αναφορές στις έννοιες του δομημένου και του αντικειμενοστραφούς προγραμματισμού σχετίζονται με τις μεθόδους και τις τεχνικές μέσω των οποίων επιλύεται ένα πρόβλημα προγραμματισμού. Αντίθετα, δεν γίνεται λόγος για συγκεκριμένα εργαλεία (γλώσσες προγραμματισμού), η επιλογή των οποίων επαφίεται σε μεγάλο βαθμό στην κρίση του εκπαιδευτικού που διδάσκει το μάθημα.

Πιο συγκεκριμένα, όπως αναφέρεται στο πρόγραμμα σπουδών, «σκοπός του μαθήματος δεν είναι η εκμάθηση μιας γλώσσας προγραμματισμού. Έμφαση δίνεται στις ενότητες *Ανάλυση προβλήματος και Σχεδίαση αλγορίθμου*, ώστε οι μαθητές, να αναπτύξουν δεξιότητες αλγοριθμικής προσέγγισης, δημιουργικότητα, φαντασία, αναλυτικό πνεύμα και αυστηρότητα στην έκφραση, και να μπορούν να διακρίνουν ποια προβλήματα αντιμετωπίζονται σε προγραμματιστικό περιβάλλον» (ΕΠΠΣ, 1997).

## 2 Προγραμματισμός Η/Υ

---

### 2.1 Πληροφορική, προγραμματισμός, γλώσσες προγραμματισμού

Για μια μεγάλη περίοδο, η διδασκαλία της Πληροφορικής ταυτιζόταν με τη διδασκαλία του προγραμματισμού ενώ παράλληλα, για πολλά χρόνια, η διδασκαλία του προγραμματισμού ήταν συνδεδεμένη με τη διδασκαλία μιας γλώσσας προγραμματισμού. Το στάδιο αυτό ξεπεράστηκε κατά τη δεκαετία του 1970, κυρίως μετά τη δουλειά των Knuth (Knuth, 1968) και Dahl, Dijkstra, Hoare (Dahl, Dijkstra, Hoare, 1972, Dijkstra, 1976), η οποία ανέδειξε τα αδιέξοδα που εμπεριείχε αυτή η προσέγγιση και από τότε παρατηρείται μια νέα κοινή βάση για τη διδασκαλία του προγραμματισμού.

Το ενδιαφέρον πλέον εστιάζεται περισσότερο στις μορφές συλλογισμού που χρησιμοποιούν οι αρχάριοι και οι έμπειροι προγραμματιστές (Lieberman, 1986 Pennington, 1987) και στις μεθόδους εργασίας με σκοπό την καλή σύλληψη προγραμμάτων (Gueraud et Peyrin, 1988).

Στο πλαίσιο αυτό, ο προγραμματισμός μελετάται τόσο από τους ψυχολόγους όσο και από τους επιστήμονες της Διδακτικής ως μια ανθρώπινη δραστηριότητα που εμπερικλείει το σχεδιασμό της συμπεριφοράς του υπολογιστή με στόχο να υποβοηθά. και κάποιες φορές να υποκαθιστά, τους ανθρώπους σε νοητικές εργασίες (Rogalski. 1988).

Ο προγραμματισμός συνιστά μια ιδιαίτερη δραστηριότητα, που δεν προϋπήρχε της Πληροφορικής και δεν μπορεί να παρομοιαστεί με καμία άλλη ανθρώπινη δραστηριότητα. Κάτω από αυτό το πρίσμα, ενδιαφέρει τους παιδαγωγούς και τους ψυχολόγους.

Οι παιδαγωγοί βρίσκουν στον προγραμματισμό ένα αδιαμφισβήτητο εκπαιδευτικό ενδιαφέρον, το οποίο συνίσταται κυρίως στην ανάλυση ενός προβλήματος ή μιας κατάστασης, που προηγείται της συγγραφής ενός προγράμματος, και συνακόλουθα ενός τρόπου διδασκαλίας της λογικής σκέψης και της αλγοριθμικής επίλυσης προβλημάτων, εφαρμοσμένης σε οικείες περιστάσεις.

Οι ψυχολόγοι θεωρούν ότι το ζεύγος ανάλυση - προγραμματισμός συνιστά μία νοητική δραστηριότητα η οποία αποτελεί αντικείμενο μελέτης όπως και κάθε άλλη ανθρώπινη δραστηριότητα. Η δραστηριότητα αυτή ανήκει σε μία πιο εκτεταμένη κατηγορία έργων, την οποία ονομάζουν επίλυση προβλήματος (problem solving) (Kahney, 1993).

Συνεπώς, ο προγραμματισμός είναι μια τυπική δραστηριότητα μιας πιο εκτεταμένης κλάσης δραστηριοτήτων που αποκαλούνται επίλυση προβλημάτων στην ψυχολογία. Η κλάση αυτή συνίσταται κατά κύριο λόγο στη σύλληψη και στην αποσαφήνιση των διαδικασιών επεξεργασίας (processing).

Μέσα στο ρεπερτόριο των ανθρώπινων ικανοτήτων, μία από τις πιο χαρακτηριστικές είναι αυτή που επιτρέπει στο άτομο να επεξεργάζεται καταστάσεις και προβλήματα. Μια **κατάσταση προβλήματος** μπορεί να χαρακτηριστεί από τρία στοιχεία:

**α) την κατάσταση εκκίνησής του, δηλαδή την αρχική κατάσταση,**

**β) μια κατάσταση - σκοπό, δηλαδή την κατάσταση στην οποία οφείλουμε να φθάσουμε, και**

**γ) τις επιτρεπτές πράξεις, που παρέχουν τη δυνατότητα να τροποποιήσουμε την κατάσταση με τρόπο ώστε να συνδέσουμε την αρχική κατάσταση με την κατάσταση - σκοπό.**

Στον προγραμματισμό, με την ευρεία έννοια, οι δραστηριότητες αυτές αφορούν: την οικοδόμηση μεθόδων και τεχνικών, την ανακάλυψη ή τη βελτιστοποίηση αλγορίθμων, τη σύνταξη οδηγιών χρήσης και συμβουλών, κ.λπ. Στο επίκεντρο αυτής της δραστηριότητας τοποθετείται το ερώτημα της μετάβασης από μια διαδικασιακή γνώση (αυτό που κάνουμε) σε μια δηλωτική γνώση (διατύπωση των ιδιοτήτων αυτού που κάνουμε).

## 2.2 Προγραμματισμός και ψυχολογία: επίλυση προβλημάτων

Η δραστηριότητα του προγραμματισμού στην ψυχολογία θεωρείται ως διαδικασία *επίλυσης προβλήματος* (problem solving). Πολλές έρευνες έγιναν κυρίως στις δεκαετίες του 1970 και του 1980. Οι έρευνες αυτές (Soloway & Spohrer, 1989· Hoc et al., 1991) μελέτησαν τις ακόλουθες πτυχές:

- Νοητικές δραστηριότητες του προγραμματιστή ως διαδικασίες μέσω των οποίων τα υποκείμενα (επαγγελματίες προγραμματιστές ή αρχάριοι) συλλαμβάνουν προγράμματα, οργανώνουν το σχεδιασμό και τις στρατηγικές, αναπτύσσουν αλγορίθμους.
- Δραστηριότητα συγγραφής εντολών (σε επίπεδο σύνταξης και σημασιολογίας).
- Ανάλυση και κατανόηση προγραμμάτων γραμμένων από το ίδιο άτομο ή από άλλα άτομα, και στρατηγικές αναζήτησης λαθών.
- Μάθηση γλωσσών προγραμματισμού από ενήλικες και παιδιά.
- Σύγκριση στρατηγικών ανάμεσα σε αρχάριους και προχωρημένους προγραμματιστές.
- Δυσκολίες, λανθασμένες παραστάσεις, και σφάλματα (bugs) που εμφανίζονται κατά την προγραμματιστική δραστηριότητα.
- Σχέση μαθηματικών και προγραμματισμού.

Ο χώρος της επίλυσης προβλημάτων συνιστά ένα ιδιαίτερα ευρύ πεδίο μελέτης τόσο στη διδακτική όσο και στην ψυχολογία. Πρόβλημα θεωρείται ότι υπάρχει όταν υπάρχει διάσταση ανάμεσα στην παρούσα κατάσταση και στον προς επίτευξη στόχο.

Με την έννοια *πρόβλημα* εννοείται στα πλαίσια της ψυχολογίας κάθε κατάσταση που χαρακτηρίζεται από (Smith, 1991· Kahney, 1993):

1. Ένα σύνολο δεδομένων (υλικά αντικείμενα, δράσεις, συμβάντα, συμβολικές και γλωσσικές αναπαραστάσεις, γραφικά, κ.λπ.) που προσδιορίζουν την *αρχική κατάσταση* του προβλήματος,
2. Ένα σύνολο ερωτημάτων που προσδιορίζουν τον προς επίτευξη σκοπό,

δηλαδή την *τελική κατάσταση* (κατάσταση - στόχος),

3. Ένα σύνολο δυσχερειών που περιορίζουν τις πράξεις του υποκειμένου, και
4. Ένα σύνολο από δυνατούς χειρισμούς που επιτρέπουν τη λύση του προβλήματος.

Η επίλυση ενός προβλήματος συνίσταται στην ανάπτυξη στρατηγικών και στην εγκαθίδρυση διαδικασιών που επιτρέπουν τη σύνδεση ανάμεσα στην αρχική και την τελική κατάσταση (κατάσταση - στόχος).

Στο πλαίσιο αυτό, βασικός στόχος της διδασκαλίας του προγραμματισμού πρέπει να είναι η μεταφορά δεξιοτήτων επίλυσης προβλημάτων δηλαδή, η ανάπτυξη της ικανότητας των μαθητών να εφαρμόζουν τις γνώσεις τους για την επίλυση προβλημάτων που δεν έχουν διδαχθεί πιο πριν (Τζιμογιάννης, 2003). Τα προβλήματα αυτά ονομάζονται επίσης «αυθεντικά» προβλήματα.

Η επίλυση προβλήματος συνιστά μια σύνθετη νοητική διαδικασία που εμπερικλείει το συντονισμό ενός συνόλου από απαιτητικές και αλληλοσυνδεόμενες δεξιότητες. Οι δεξιότητες αυτές περιλαμβάνουν:

- Κατανόηση και αναπαράσταση της αρχικής κατάστασης του προβλήματος, συμπεριλαμβανομένου και του προσδιορισμού των ειδών της πληροφορίας που απαιτείται για τη λύση του.
- Συλλογή και οργάνωση της κατάλληλης και ουσιώδους πληροφορίας.
- Κατασκευή και διαχείριση ενός σχεδίου δράσης ή μιας στρατηγικής και αναζήτηση ευρετικών τεχνικών (heuristics).
- Χρήση διαφόρων εργαλείων επίλυσης προβλήματος.
- Διαχωρισμός ενός σύνθετου προβλήματος σε απλούστερα προβλήματα, η λύση των οποίων είναι ενδεχομένως γνωστή.
- Συλλογισμός, έλεγχος υποθέσεων, και λήψη απόφασης.

Στο πλαίσιο αυτό, όπως αναφέρει ο Hoc (Hoc, 1987), ένα προγραμματιστικό πρόβλημα είναι άλλοτε ένα πρόβλημα *μετασχηματισμού καταστάσεων* και άλλοτε ένα πρόβλημα *σχεδιασμού*. Ο στόχος του προγραμματιστή όταν χρησιμοποιεί ένα συμβατικό προγραμματιστικό περιβάλλον είναι να αναπτύξει μια εκτελέσιμη διαδικασία από μια

καλώς ορισμένη μηχανή.

Στην περίπτωση αυτή δεν πρόκειται για τον υπολογιστή ως φυσική συσκευή, αλλά για την υποκείμενη φορμαλιστική μηχανή που σχετίζεται άμεσα με το χρησιμοποιούμενο προγραμματιστικό περιβάλλον.

## **2.3 Πλαίσιο ανάλυσης της δραστηριότητας του προγραμματισμού**

Πώς αναλύεται η προγραμματιστική δραστηριότητα και ποια είναι τα βασικά συνθετικά της κάτω από το πρίσμα μιας εποικοδομιστικής διδακτικής προσέγγισης (Ben-Ari, 2001); Η Linn, για παράδειγμα (Linn, 1985), μελετώντας τις γνωστικές συνέπειες της διδασκαλίας του προγραμματισμού, κάνει τη διάκριση ανάμεσα σε δεξιότητες σχεδιασμού λύσεων σε παρόμοια προβλήματα και σε μεταγνωστικές δεξιότητες επίλυσης σύνθετων και πρωτότυπων (δηλαδή αυθεντικών) προβλημάτων.

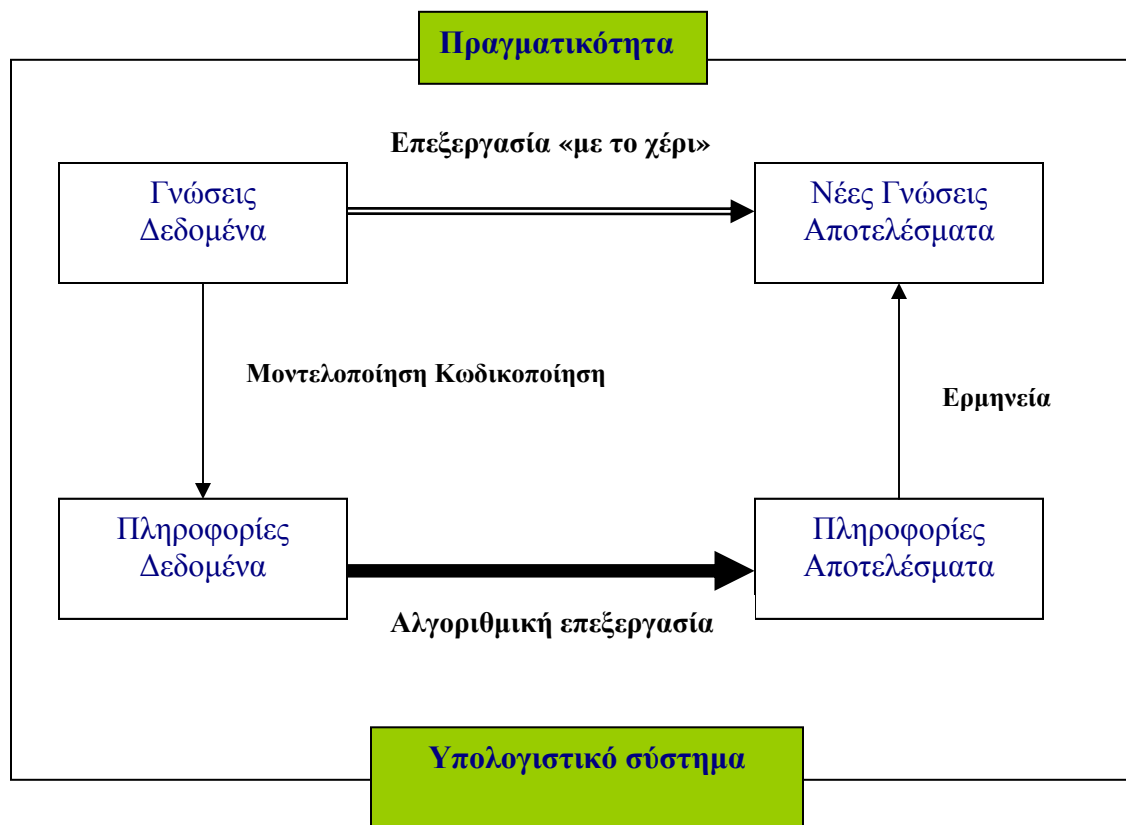
Από τη μεριά του, ο J.-M. Hoc (Hoc, 1987), εντάσσει τη δραστηριότητα του προγραμματισμού στο πλαίσιο μιας ειδικής κατηγορίας επίλυσης προβλημάτων, τα οποία ασχολούνται με τη σύλληψη και τη διασαφήνιση των διαδικασιών επεξεργασίας.

Γενικότερα, η προγραμματιστική δραστηριότητα συνίσταται στην κωδικοποίηση γνώσεων του πραγματικού κόσμου και στην αναζήτηση μιας επεξεργασίας πάνω σε αφηρημένα αντικείμενα (Lagrange, 1990). Στο πλαίσιο αυτό, η επίλυση προβλήματος σε ένα κλασικό προγραμματιστικό περιβάλλον υποθέτει την κωδικοποίηση οντοτήτων (αντικείμενα, σχέσεις, ...), τις οποίες αφορά το πρόβλημα με τη μορφή πληροφορικών δεδομένων και την οικοδόμηση ενός αλγορίθμου επεξεργασίας αυτών των δεδομένων. Η διαδικασία αυτή μπορεί να αναπαρασταθεί από το επόμενο σχήμα (Arsac, 1987 ·Duchateau, 1991) (Σχήμα 1).

Κατά την προγραμματιστική δραστηριότητα, ο αρχάριος προγραμματιστής που προσπαθεί να επιλύσει ένα πρόβλημα είναι συνήθως σε θέση να κάνει την επεξεργασία των δεδομένων «με το χέρι» (εάν η κατάσταση είναι γνωστή ή οικεία).

Αντιθέτως, η σχεδίαση και η ανάπτυξη ενός προγράμματος απαιτεί από το υποκείμενο που την εξασκεί μια ορισμένη γνώση των αφηρημένων αντικειμένων που

συνίστανται από τις κωδικοποιημένες πληροφορίες (για να πραγματοποιήσει την κωδικοποίηση), καθώς και του τρόπου με τον οποίο το υπολογιστικό σύστημα μπορεί να τις επεξεργαστεί (ώστε να φανταστεί ή και να οικοδομήσει την αλγοριθμική επεξεργασία) (Lagrange, 1990). Η γνώση αυτή σχετίζεται με τις ιδιαιτερότητες του υπολογιστικού συστήματος και τις δυσχέρειες που επιβάλλει η χρήση του.

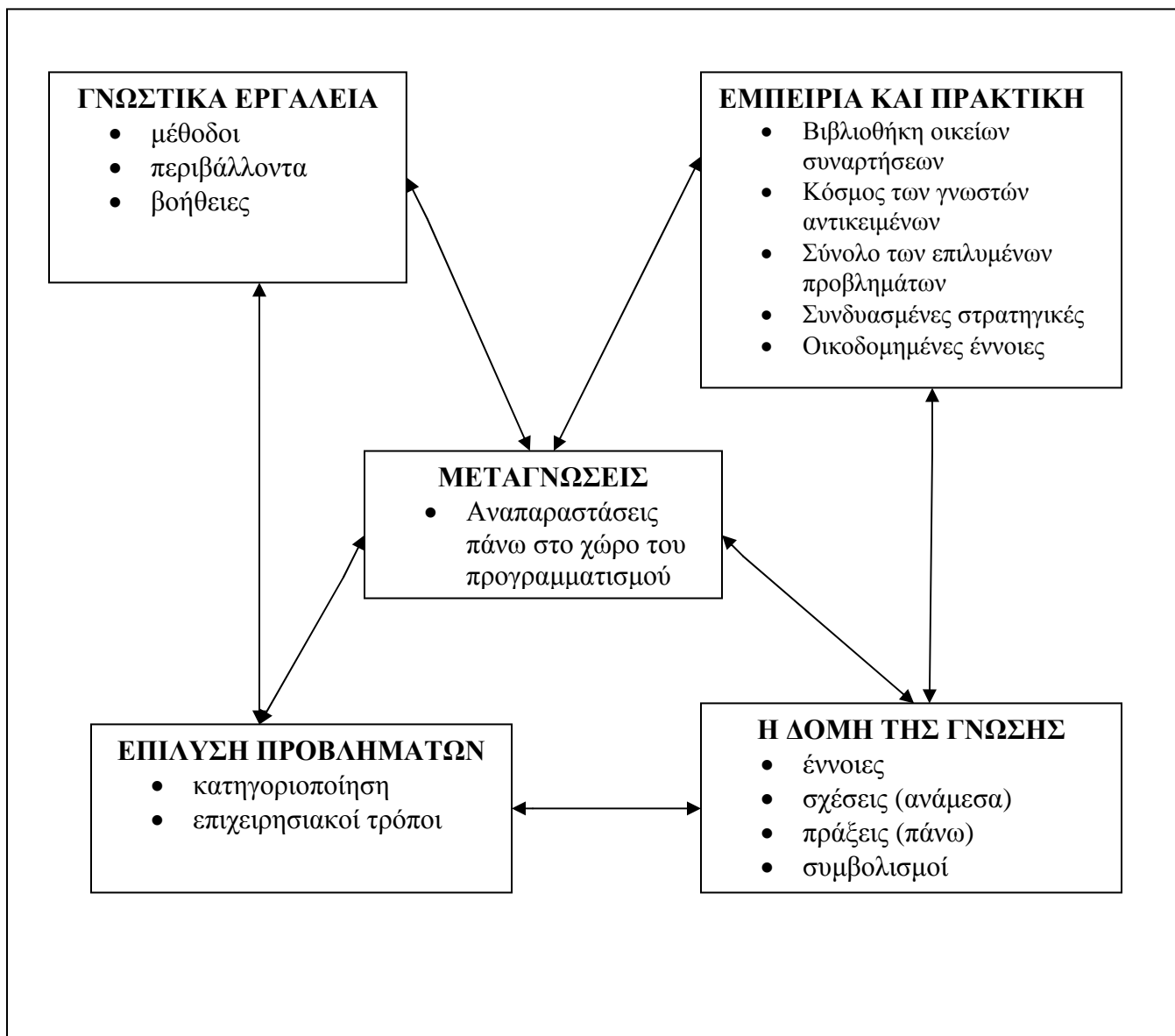


Σχήμα 1: Η δραστηριότητα του προγραμματισμού

Ένα γενικό θεωρητικό πλαίσιο για την ανάλυση της δραστηριότητας του προγραμματισμού παρουσιάζεται συνοπτικά στο Σχήμα 1. Το πλαίσιο αυτό παρουσιάζει πολλούς «χώρους» σε αλληλεξάρτηση (Green, Hoc, Samurcay & Gilmore, 1990) : δομή των γνώσεων, πεδίο επίλυσης προβλημάτων, χώρο της πρακτικής, χώρο των «γνωστικών εργαλείων» και, τέλος, το χώρο της μεταγνώσης που αφορά στο χώρο του προγραμματισμού.



## Συνθήκες για τη γνωστική δραστηριότητα του προγραμματισμού



### ΑΠΑΙΤΗΣΕΙΣ ΤΟΥ ΧΩΡΟΥ

Σχήμα 2: Πλαίσιο ανάλυσης της δραστηριότητας του προγραμματισμού

Η συνολική αυτή δομή αντανακλά αφενός την υπόθεση ότι η γνώση οικοδομείται και αποδεικνύεται με την επίλυση προβλημάτων (εποικοδομισμός) και δεν ταυτίζεται με ένα κείμενο γνώσεων (ακόμα και αν είναι δομημένο) και, αφετέρου, μία αντίληψη της δραστηριότητας του προγραμματισμού η οποία εντάσσει στο σκοπό της (δηλαδή το προς επίλυση προγραμματιστικό πρόβλημα) τα γνωστικά εργαλεία και τις εν δυνάμει

εμπειρικές αναφορές που διαθέτει ο προγραμματιστής.

Η *δομή των γνώσεων* συνίσταται σε ένα σύστημα εννοιών, πράξεων πάνω στις έννοιες, σχέσεων ανάμεσά τους, και συμβολισμών για την αναπαράσταση των εννοιών, με μια ιεραρχική οργάνωση σε επίπεδα (όπως, για παράδειγμα, σε ποιο επίπεδο μπορούν να οικοδομηθούν η πληροφορική μεταβλητή, οι βασικές λογικές δομές, η έννοια της διαδικασίας, κ.λπ.).

Κάθε πρόβλημα προγραμματισμού αναφέρεται σε ένα σύνολο εννοιών σε αλληλεπίδραση όπως, για παράδειγμα, η επεξεργασία λιστών συνεπάγεται τις έννοιες της μεταβλητής, της επανάληψης, της αναδρομικότητας, των συναρτήσεων που επιδρούν στις λίστες, των ελέγχων, των εισόδων και των εξόδων δεδομένων, και αυτό σε οποιαδήποτε γλώσσα προγραμματισμού και με οποιοσδήποτε συντακτικές και σημασιολογικές ιδιαιτερότητες.

Για να περάσουμε στη φάση της εκτέλεσης πρέπει επίσης να επιλεγούν οι κατάλληλες αναπαραστάσεις στο επίπεδο της συντακτικής μορφής, στο εσωτερικό της δεδομένης γλώσσας. Άλλωστε, τύποι προβλημάτων και τρόποι επίλυσης συνδέονται. Για παράδειγμα, υπάρχουν κλασικοί αλγόριθμοι για τα προβλήματα άθροισης, ταξινόμησης, και επεξεργασίας λιστών ή γράφων.

Οι γνωστικές δραστηριότητες στον προγραμματισμό εξαρτώνται από την *πρότερη εμπειρία* και τα ατομικά. *γνωστικά εργαλεία* του μαθητή. Μέσα σε μια διαδικασία μάθησης, οι διάφοροι «χώροι» εξελίσσονται. Νέες γνώσεις οικοδομούνται μέσω νέων αλληλεπιδράσεων. Για παράδειγμα, η διαβίβαση των παραμέτρων σε διαδικασίες και συναρτήσεις συνεπάγεται όχι μόνο την πρόσκτηση της έννοιας της διαδικασίας, αλλά επίσης έναν εκ νέου ορισμό των μεταβλητών ως τοπικών ή καθολικών. Επομένως, αναφέρεται στην οικοδόμηση ενός νέου επιπέδου κατανόησης των σχέσεων ανάμεσα στη μεταβλητή, το όνομα, και την τιμή της (Κόμης, 2005).

Στην αρχή της μαθησιακής διαδικασίας ο μαθητής αναφέρεται βασικά στην προηγούμενη εμπειρία του. Σύμφωνα με την οργάνωση της διδασκαλίας, μπορεί να σχηματίσει ένα μοντέλο Συστήματος Αναπαράστασης και Επεξεργασίας του δεδομένου πληροφορικού μέσου (ειδικότερα, της γλώσσας και του περιβάλλοντός της) είτε μέσω της πράξης, είτε μέσω αναλογιών με άλλα γνωστά μέσα, είτε με βάση άλλους χώρους και κυρίως με βάση τα μαθηματικά (τα οποία έχουν κοινή με την Πληροφορική τη

διαδικασία μοντελοποίησης των «προβλημάτων του κόσμου»), είτε με βάση τις ήδη γνωστές γλώσσες προγραμματισμού.

Στη συνέχεια της μαθησιακής διαδικασίας, το σύνολο των ήδη επιλυμένων προβλημάτων αυξάνει, και γίνεται πλέον εφικτή μια, έστω ασυνείδητη, κατηγοριοποίηση. Διάφορες στρατηγικές γίνονται αυτόματα διαθέσιμες και συνδέουν τύπους προβλημάτων με τρόπους επίλυσης για παράδειγμα, τα προβλήματα που λύνονται με αναδρομή. Συνεπώς, αυξάνει και η πολυπλοκότητα των προβλημάτων τα οποία μπορεί να προσεγγίσει ο μαθητής. Επιπρόσθετα, σταδιακά γίνονται οικεία αντικείμενα οι δομές δεδομένων, που είναι πολύπλοκες κατά την αρχική φάση της διδασκαλίας (όπως για παράδειγμα οι πίνακες στις διαδικαστικές γλώσσες).

Τα νοητικά σχήματα που δημιουργούνται από πρότερες γνώσεις μπορούν να προκαλέσουν παραγωγικά αποτελέσματα, σαν ένα πλαίσιο υποστήριξης (scaffolding)-για παράδειγμα, δίνοντας νόημα σε νέες έννοιες και πράξεις. Μπορούν όμως να παράγουν και μειωτικά αποτελέσματα (να γίνουν δηλαδή διδακτικά εμπόδια), λόγω των διαφορών των δομών των γνώσεων μέσα στο χώρο προέλευσης και το χώρο - «στόχο» προσκλήσεων.

Η ύπαρξη μειωτικών και παραγωγικών αποτελεσμάτων διαφαίνεται και από τα πειραματικά αποτελέσματα των ερευνών. Αρκετές έρευνες (Soloway & Spohrer, 1989) δείχνουν την ύπαρξη λανθασμένων αντιλήψεων που προέρχονται από πρότερες, προγραμματιστικές γνώσεις (preprogramming knowledge).

Μια οικοδομημένη γνώση προγραμματισμού μπορεί να αποδειχθεί διδακτικό ή γνωστικό εμπόδιο και να απαιτήσει μια διαδικασία «αποδόμησης», η οποία συνήθως λαμβάνει χώρα μέσω διαδικασιών γνωστικής ή κοινωνικογνωστικής σύγκρουσης.

Επιπρόσθετα, σε διάφορες έρευνες παρατηρείται ότι οι φοιτητές με μαθηματικό υπόβαθρο δεν έχουν ανάγκη να παρακολουθήσουν τόση διδασκαλία όσο οι «μη προερχόμενοι από επιστημονικούς χώρους» μαθητές (Soloway & Spohrer, 1989, Green, Hoc, Samurcay & Gilmore, 1990).

Όσον αφορά το πεδίο των μεταγνώσεων (δηλαδή των σχετικών με συγκεκριμένη γνώση αναπαραστάσεων) φαίνεται ότι οι αναπαραστάσεις του μαθητή πάνω σε αυτό που είναι ο προγραμματισμός μπορούν να επηρεάσουν σημαντικά τη μάθησή του.

## 2.4 Η μάθηση του Προγραμματισμού

### 2.4.1 Το σύστημα Αναπαράστασης και Επεξεργασίας

Η μάθηση του προγραμματισμού (με την ευρεία έννοια - που υπονοεί ακόμη και τη χρήση υπολογιστικών εργαλείων όπως λογιστικά φύλλα και βάσεις δεδομένων) μπορεί να οδηγήσει στην εγκαθίδρυση νέων τρόπων αντίληψης, μοντελοποίησης, και επεξεργασίας προβλημάτων και, για το λόγο αυτό, συνιστά ένα εξαιρετικά ενδιαφέρον πεδίο στο πρόγραμμα σπουδών, που δεν προσεγγίζεται από κανένα άλλο γνωστικό αντικείμενο στο χώρο της δευτεροβάθμιας εκπαίδευσης.

Οι διάφορες έρευνες αναφέρουν ότι, κατά τα πρώτα στάδια της μάθησης του προγραμματισμού, οι μαθητές συμπεριφέρονται χρησιμοποιώντας γνωστικές διεργασίες που τους είναι ήδη οικείες (Mayer, 1981). Προσπαθούν να αντιμετωπίσουν νέες καταστάσεις χρησιμοποιώντας καταρχήν γνωστά νοητικά σχήματα (κυρίως από τις βιοματικές εμπειρίες τους, αλλά και από τις μαθηματικές γνώσεις τους). Ωστόσο, απέναντι στα γεγονότα και την πραγματικότητα, οι νοητικές δομές αυτές εξανάγκης τροποποιούνται. Η προσαρμογή αυτή οδηγεί το μαθητή να οικοδομήσει σταδιακά νέες δομές αναπαράστασεων, δηλαδή να προχωρήσει στην εννοιολογική αλλαγή.

Παρατηρώντας αρχάριους προγραμματιστές βλέπουμε ότι δεν ικανοποιούν δύο θεμελιώδεις απαιτήσεις του προγραμματισμού: την αναγκαιότητα της αναπαράστασης της δομής του προγράμματος και την αναγκαιότητα της σύλληψης μιας διαδικασίας συμβατής με τη λειτουργία της μηχανής. Οι απαιτήσεις αυτές ικανοποιούνται μόνο όταν ο προγραμματιστής οικοδομήσει αυτό που οι ψυχολόγοι, οι οποίοι έχουν μελετήσει την προγραμματιστική δραστηριότητα, ονομάζουν ένα «**Σύστημα Αναπαράστασης και Επεξεργασίας**» (Green, Hoc, Samurçay & Gilmore, 1990).

Πριν όμως οικοδομηθεί αυτό το σύστημα, το οποίο συνδέεται άμεσα με την χρησιμοποιούμενη γλώσσα προγραμματισμού, οι μαθητές χρησιμοποιούν άλλα συστήματα πιο οικεία σε αυτούς. Για παράδειγμα, η πιο συνηθισμένη παρατήρηση που αφορά στους αρχάριους προγραμματιστές είναι ότι προγραμματίζουν όπως σκέφτονται (δηλαδή συλλογίζονται με τη λογική της επεξεργασίας «με το χέρι» που αναφέραμε στην προηγούμενη ενότητα), χωρίς να λαμβάνουν υπόψη τους τις λειτουργίες της μηχανής.

## 2.4.2 Τύποι γνώσεων στον προγραμματισμό

Οι Bayman και Mayer έχουν αναπτύξει ένα πλαίσιο για τις ειδικές γνώσεις που απαιτούνται στον προγραμματισμό. Σύμφωνα με το πλαίσιο αυτό, ταξινομούν τους διάφορους τύπους προγραμματιστικών γνώσεων που εμφανίζουν οι αρχάριοι και οι έμπειροι προγραμματιστές στις παρακάτω κατηγορίες (Bayman & Mayer, 1988 Mayer, 1988):

### *α) Συντακτική γνώση*

Είναι η γνώση των ειδικών χαρακτηριστικών μιας γλώσσας προγραμματισμού και των κανόνων χρήσης της. Η συντακτική γνώση εξαρτάται στενά από τη χρησιμοποιούμενη γλώσσα προγραμματισμού (Shneiderman & Mayer, 1979).

### *β) Εννοιολογική γνώση*

Η εννοιολογική γνώση αφορά στην πλήρη κατανόηση των προγραμματιστικών δομών και αρχών. Διακρίνεται στη σημασιολογική (semantic knowledge) και στη σχηματική γνώση (schematic knowledge). Η σημασιολογική γνώση βασίζεται σε ολοκληρωμένα εννοιολογικά μοντέλα για το σύστημα του υπολογιστή, το τι συμβαίνει στο εσωτερικό του κατά την εκτέλεση των εντολών του προγράμματος, την έννοια της μεταβλητής, τις δομές επιλογής και επανάληψης κ.λ.π. Η σχηματική γνώση συνίσταται στο ρεπερτόριο ρουτινών και αλγορίθμων που διαθέτει ο μαθητής. Σε αντίθεση με τους μαθητές, οι έμπειροι προγραμματιστές έχουν ένα μεγάλο, δομημένο σύνολο ρουτινών που είναι σε θέση εύκολα να ανακαλέσουν και να εφαρμόσουν σε νέες καταστάσεις (Τζιμογιάννης, 2002).

### *γ) Στρατηγική γνώση (μεταγνώση)*

Είναι η ικανότητα εφαρμογής των συντακτικών και εννοιολογικών γνώσεων για την επίλυση αυθεντικών προβλημάτων προγραμματισμού. Βασίζεται στην ανάπτυξη δεξιοτήτων υψηλού επιπέδου (ανάλυση-σύνθεση, διατύπωση συνθηκών και αιτιακών συσχετισμών κ.λ.π.) για το σχεδιασμό προγραμμάτων και στην ικανότητα μεταφοράς δεξιοτήτων επίλυσης προβλημάτων σε άλλα γνωστικά αντικείμενα (McGill & Violet, 1997).

### 2.4.3 Εποικοδόμηση των εννοιών του προγραμματισμού

Το σύγχρονο παιδαγωγικό και μαθησιακό πλαίσιο βασίζεται στη θεωρία του **εποικοδομισμού ή κονστρουκτιβισμού (constructivism)**. Βασική αρχή του εποικοδομισμού είναι ότι η γνώση οικοδομείται από τους ίδιους τους μαθητές και δεν μεταφέρεται παθητικά από τις διαλέξεις του διδάσκοντα ή τα βιβλία. Οι μαθητές οικοδομούν τις νέες γνώσεις συνδυάζοντας τον εμπειρικό κόσμο με τις προϋπάρχουσες γνωστικές δομές και τα νοητικά μοντέλα. Κύρια εργαλεία του εποικοδομισμού είναι η θεώρηση των προϋπαρχουσών γνώσεων και αντιλήψεων των μαθητών και η οργάνωση διδακτικών-μαθησιακών δραστηριοτήτων που ευνοούν τη **ανακαλυπτική μάθηση (discovery learning)** και τη **συνεργατική μάθηση (cooperative learning)**.

Η διδασκαλία του προγραμματισμού στο πλαίσιο των αρχών του εποικοδομισμού έχει αποτελέσει αντικείμενο ενδιαφέροντος τα τελευταία, κυρίως, χρόνια (Ben-Ari, 2001· Κόμης, 2001). Υπάρχουν όμως ιδιαίτερα χαρακτηριστικά στη διδασκαλία του προγραμματισμού, τα οποία σχετίζονται με το ρόλο της μηχανής.

Ο υπολογιστής και το περιβάλλον προγραμματισμού έχουν διπλό ρόλο: αποτελούν **ένα μηχανισμό** ο οποίος μπορεί να χρησιμοποιηθεί για **την ανάπτυξη και εκτέλεση άλλων μηχανισμών** (προγραμμάτων). Από την άλλη μεριά, ο μαθητής έρχεται σε άμεση επαφή με τη μηχανή, από την οποία παίρνει ανάδραση ως συνέπεια των δικών του νοητικών μοντέλων. Οι παρανοήσεις και τα εναλλακτικά νοητικά μοντέλα των μαθητών προκαλούν σφάλματα (bugs), άμεσα παρατηρήσιμα στην οθόνη του υπολογιστή κατά την εκτέλεση των δικών τους προγραμμάτων. Ο υπολογιστής παίζει σημαντικό ρόλο στη διδασκαλία του προγραμματισμού. Δεν αποτελεί απλά τη μηχανή που εκτελεί το πρόγραμμα. Η μεγάλη παιδαγωγική του αξία οφείλεται στο ότι επιτρέπει τη σύνταξη, τον έλεγχο και τη διόρθωση του προγράμματος από τους ίδιους τους μαθητές. Με άλλα λόγια, ο υπολογιστής παρέχει σε κάθε μαθητή δυνατότητες εξάσκησης, πειραματισμού, ελέγχου των νοητικών μοντέλων για τα υπολογιστικά αντικείμενα και τις προτεινόμενες λύσεις, συνεργασίας και αλληλεπίδρασης με τους συμμαθητές του.

Ο Du Boulay (1989) έχει εισάγει τον όρο εννοιολογική μηχανή (notional machine) για να περιγράψει το ρόλο του υπολογιστή στον προγραμματισμό και τις αντιλήψεις που έχουν οι αρχάριοι γι' αυτόν. Ένα πρόγραμμα που εκτελείται στον

υπολογιστή αποτελεί ένα μηχανισμό, ο οποίος δεν είναι εύκολα προσιτός στους μαθητές. Οι δεξιότητες που είναι απαραίτητες, ώστε οι μαθητές να εμπεδώσουν το πρόγραμμα ως μια ενιαία οντότητα, να κατανοήσουν τα κύρια μέρη του και τις σχέσεις μεταξύ τους, απαιτούν χρόνο και αναπτύσσονται σταδιακά.

Η έρευνα έχει αποδείξει ότι οι μαθητές δεν έχουν αναπτύξει αποτελεσματικά μοντέλα για τον υπολογιστή και τη λειτουργία του κατά την εκτέλεση του προγράμματος (Bonar & Soloway 1985, Rogalski & Vergnaud 1987). Η διδακτική εμπειρία δείχνει ότι, σε πολλές περιπτώσεις, η μηχανή αντιμετωπίζεται από τους μαθητές ως μαύρο κουτί (black box). Τα μοντέλα και οι αντιλήψεις των μαθητών εισάγουν πρόσθετα γνωστικά εμπόδια στο να κατανοήσουν το επίπεδο λεπτομέρειας, το οποίο απαιτείται για να περιγραφεί μια διαδικασία σε μια γλώσσα προγραμματισμού. Συχνά οι μαθητές αποδίδουν ανθρωπομορφικά χαρακτηριστικά στον υπολογιστή έχοντας την αντίληψη ότι είναι "νοητικός γίγαντας" ή ότι έχει "κρυμμένη νοημοσύνη" (Pea 1986, Taylor 1990). Ο Du Boulay (1989) αναφέρει ότι «... ακόμη και αν δεν γίνει καμία παρουσίαση τού τι συμβαίνει στο εσωτερικό του υπολογιστή, οι μαθητές δημιουργούν το δικό τους μοντέλο».

Σε πολλές περιπτώσεις οι ίδιες οι γλώσσες προγραμματισμού ενσωματώνουν ανθρωπομορφικά χαρακτηριστικά, τα οποία εισάγουν πρόσθετες δυσκολίες και παρανοήσεις στους μαθητές. Συχνά, η γλώσσα που χρησιμοποιούν οι εκπαιδευτικοί ενισχύει τις αντιλήψεις αυτές (για παράδειγμα, «Έτσι που έγραψες την εντολή ο υπολογιστής θεωρεί ότι ...»). Τα πρώτα διδακτικά παραδείγματα αλγορίθμων βασίζονται σε μια σειρά οδηγιών (βημάτων) που εκτελεί ο άνθρωπος για την υλοποίηση εργασιών της καθημερινής ζωής (π.χ. πώς παρασκευάζεται ένα γλυκό, πώς υλοποιείται μια κατασκευή κ.λ.π.). Η χρήση παραδειγμάτων του τύπου αυτού για την εισαγωγή των μαθητών στην αλγοριθμική επίλυση προβλημάτων οδηγεί, συνήθως, σε μια σειρά λανθασμένων αντιλήψεων, όπως ότι

- η γλώσσα σύνταξης των εντολών είναι ελεύθερη
- υπάρχουν εντολές που μπορούν να υποστηρίξουν με απλό τρόπο κάθε περίπλοκη υπολογιστική διαδικασία.

#### 2.4.4 Τι μαθαίνει αυτός που μαθαίνει να προγραμματίζει;

Οι ερευνητές και οι παιδαγωγοί, που υποστηρίζουν την ανάγκη της διδασκαλίας του προγραμματισμού στην εκπαίδευση, δεν στηρίζονται μόνο σε επιχειρήματα που αφορούν στην ανάγκη ενός πληροφορικού αλφαριθμητισμού στο πλαίσιο της τεχνοκεντρικής ή της πραγματολογικής προσέγγισης που αναπτύχθηκε σε προηγούμενη ενότητα. Αποδέχονται, παράλληλα, ότι η μάθηση του προγραμματισμού (Dufoyer, 1988) μπορεί να οδηγήσει σε τουλάχιστον επτά σημαντικές αλλαγές στο γνωστικό σύστημα των μαθητών:

- ✚ Αυστηρότητα στη σκέψη, ακρίβεια στην έκφραση, συνειδητή ανάγκη για αποσαφήνιση των ενεργειών.
- ✚ Πρόσκτηση και κατανόηση γενικών εννοιών, όπως διαδικασία, μεταβλητή, συνάρτηση, μετασχηματισμός (που σχετίζονται άμεσα και με τη μαθηματική παιδεία).
- ✚ Πρόσκτηση ευρετικών ικανοτήτων και μεθοδολογίας: σχεδιασμός, αναζήτηση παρόμοιων περιπτώσεων, επίλυση με ανάλυση σε μέρη.
- ✚ Μάθηση τεχνικών αναζήτησης λαθών που μπορούν να μεταφερθούν και σε άλλους, εκτός προγραμματισμού, χώρους.
- ✚ Πρόσκτηση της γενικής ιδέας οικοδόμησης της λύσης με τη μορφή μικρών διαδικασιών ή στοιχειωδών τμημάτων, τα οποία μπορούν να συνδεθούν και να χρησιμοποιηθούν για την οικοδόμηση της λύσης σύνθετων προβλημάτων.
- ✚ Επέκταση της συνειδητοποίησης και της γνώσης πάνω σε τεχνικές επίλυσης προβλημάτων.
- ✚ Επέκταση και ανάπτυξη της χρήσης συγκριτικών μεθόδων που αφορούν στην πολλαπλότητα των τρόπων ώστε να επιτευχθεί ένας δεδομένος στόχος.

Συνοψίζοντας, μπορούμε να πούμε ότι:

- ✚ Μάθηση του προγραμματισμού συνεπάγεται την πρόσκτηση μιας σύνταξης, μιας δομής της γλώσσας και των αντίστοιχων αναπαραστάσεων.
- ✚ Η μάθηση αυτή υποθέτει μια αποκέντρωση του συστήματος σκέψης του προγραμματιστή προς τη λειτουργία της μηχανής.
- ✚ Ο προγραμματισμός είναι μια εξελικτική δραστηριότητα: η κατασκευή ενός προγράμματος μπορεί να δώσει ιδέες για την κατασκευή άλλων προγραμμάτων.



- ✚ Το υποκείμενο που προγραμματίζει αποκτά την εντύπωση ότι διαρκώς εξελίσσεται γνωστικά.

Ο μαθητής που προγραμματίζει μαθαίνει ταυτόχρονα πολλά πράγματα. Αναπτύσσει δεξιότητες φορμαλιστικής περιγραφής. Αναγκασμένος να περιγράφει με ακριβή τρόπο το αποτέλεσμα ενός μετασχηματισμού στον οποίο υπόκειται ένα αντικείμενο, μαθαίνει να κωδικοποιεί και να καθιστά αντικειμενικές τις πράξεις του. Τέλος, ανακαλύπτει πληροφορίες πάνω στον ίδιο τον τρόπο σκέψης διδάσκοντάς τις σε μια μηχανή (με την έννοια που χρησιμοποιείται στην προσέγγιση της Logo).

#### **2.4.5 Απαραίτητες δεξιότητες για τη μάθηση του προγραμματισμού**

Οι εγγενείς δυσκολίες της μάθησης του προγραμματισμού (Soloway & Spohrer, 1989), μας οδηγούν να διερωτηθούμε μέσα σε ποιο γνωστικό πλαίσιο διεξάγεται η μάθηση του προγραμματισμού. Δηλαδή, ποιες είναι οι απαραίτητες γνωστικές δεξιότητες που συνιστούν προαπαιτούμενο για τη μάθηση του προγραμματισμού. Εάν δεν έχουν οικοδομηθεί από τους μαθητές οι δεξιότητες αυτές, είναι μάταιο να προσπαθήσουμε εκμάθηση του προγραμματισμού. Για να είναι δυνατή η μάθηση του προγραμματισμού, ο μαθητής πρέπει να διαθέτει γνωστικές δομές οι οποίες του επιτρέπουν (Dufoyer, 1988):

- 1) Να οικοδομήσει κανόνες προγραμματισμού. Από τη μελέτη έμπειρων προγραμματιστών έχουν απαριθμηθεί τουλάχιστον εκατόν τέσσερις (104) τέτοιοι διαφορετικοί κανόνες.
- 2) Να οικοδομήσει αναλυτικές νοητικές αναπαραστάσεις όσων συμβαίνουν στη μηχανή όταν εκτελείται το πρόγραμμα (π.χ. στη μνήμη του υπολογιστή).
- 3) Να προσομοιώσει τμήματα πράξεων του υπολογιστή ώστε να μπορεί να τα προβλέπει καλύτερα (όπως είναι π.χ. οι κλήσεις μιας αναδρομικής διαδικασίας).
- 4) Να συγκρατεί νοητικά ικανές ποσότητες πληροφορίας.

Επιπρόσθετα, η δημιουργία ενός προγράμματος περνά από πολλά στάδια. Κατά τη διάρκεια κάθε σταδίου, ο προγραμματιστής πρέπει να διαθέτει κάποιες δεξιότητες (Κόμης, 2006):

- 1) Να είναι σε θέση να κατανοήσει το πρόβλημα που τέθηκε και αποτελεί αντικείμενο του προγράμματος που πρέπει να γραφτεί (ποια είναι τα δεδομένα και ποια τα ζητούμενα - αρχική και τελική κατάσταση).
- 2) Να είναι σε θέση να καθορίσει και, στη συνέχεια, να σχεδιάσει τη μέθοδο επίλυσης με την περιγραφή του αλγορίθμου (να περιγράψει δηλαδή με σαφήνεια το πέρασμα από την κατάσταση εκκίνησης στην κατάσταση - στόχο).
- 3) Να είναι σε θέση, στο πλαίσιο ενός προγραμματιστικού περιβάλλοντος, (και από τη στιγμή που θα έχει αναπτυχθεί ένας αλγόριθμος) να τον μεταφράσει σε γραπτό κώδικα.
- 4) Να είναι σε θέση να βρίσκει τα λάθη μέσα σε ένα πρόγραμμα και να καθορίζει τις αντίστοιχες λύσεις. Στο πλαίσιο αυτό, η διαδικασία της αποσφαλμάτωσης είναι εγγενές χαρακτηριστικό της προγραμματιστικής δραστηριότητας: δοκιμάζουμε ένα πρόγραμμα όχι για να δούμε εάν είναι σωστό αλλά για να βρούμε τα λάθη που περιέχει.

Δεν έχουν γίνει έρευνες οι οποίες να αφορούν στις απαραίτητες προαπαιτούμενες γνωστικές δεξιότητες των μαθητών που μαθαίνουν προγραμματισμό. Ωστόσο, έξι παράγοντες συναντώνται συχνά στη βιβλιογραφία: *μαθηματική δεξιότητα, μνημονικές ικανότητες, δεξιότητα συλλογισμού μέσω αναλογιών, δεξιότητα παραγωγής συλλογισμών υπό συνθήκη, δεξιότητα διαχείρισης και σύλληψης πρόσκαιρης σειριακότητας, δεξιότητα σκέψης με διαδικασιακό τρόπο.*

Όταν προγραμματίζουμε είναι προφανές ότι χρησιμοποιούμε γνώσεις και δεξιότητες που έχουν αποκτηθεί σε άλλους χώρους. Οι δεξιότητες αυτές είναι λογικού, γλωσσικού, φυσικομαθηματικού, κ.ά. τύπων, και εντάσσονται - όπως έχουμε ήδη αναφέρει - στις διαδικασίες επίλυσης προβλήματος.

## 2.5 Η δυσκολία εκμάθησης του προγραμματισμού

Ο Du Boulay (1989) περιγράφει πέντε βασικές περιοχές όπου επικεντρώνεται η δυσκολία της εκμάθησης του προγραμματισμού.

Η *πρώτη περιοχή (Orientation)* καλείται «*Προσανατολισμός: Τι είναι ο προγραμματισμός και σε τι μας είναι χρήσιμος*». Οι μαθητές αρκετές φορές συναντούν δυσκολίες με τις διάφορες μορφές προγραμματισμού. Σήμερα πολύ περισσότερο από ποτέ άλλοτε, ο προγραμματισμός καλύπτει μια ευρεία περιοχή των εφαρμογών λογισμικού και δεν συναντάται αποκλειστικά στα περιβάλλοντα των κλασικών γλωσσών. Πολλά πακέτα λογισμικού ενσωματώνουν δυνατότητες προγραμματισμού (συνήθως μέσω μακροεντολών), όπως λογιστικά φύλλα, επεξεργαστές κειμένου, προγράμματα δημιουργίας γραφικών κ.ά. Αυτό το γεγονός έχει καταστήσει πλέον θολά τα όρια μεταξύ του χρήστη εφαρμογών και του προγραμματιστή.

Η *δεύτερη περιοχή (Notional Machine)* καλείται «*Νοητή μηχανή – Πώς λειτουργεί ο υπολογιστής*». Ο μαθητής καλείται να «ελέγξει» μια εικονική μηχανή. Τι μορφή παίρνει η εικονική μηχανή και ποιο είδος εντολών αναμένεται να καταλαβαίνει; Πώς γίνεται η επικοινωνία, η έκδοση και η ανάγνωση των εντολών; Τα διανοητικά μοντέλα που κατασκευάζουν οι μαθητές είναι κρίσιμα για την κατανόηση κάθε νέας έννοιας στην οποία εισάγονται. Επιπλέον, η κατοχή ενός «φτωχού» διανοητικού μοντέλου μπορεί να οδηγήσει τους μαθητές να αναπτύξουν φτωχές στρατηγικές εκμάθησης με αποτέλεσμα την απουσία κινήτρου, την έλλειψη ενδιαφέροντος, την αποθάρρυνση και απογοήτευση (Kessler 1986).

Η  *τρίτη περιοχή* δυσκολίας (*Notation*), που περιγράφει ο Du Boulay (1989), αναφέρεται στα προβλήματα που προκύπτουν από την ίδια τη γλώσσα προγραμματισμού, συμπεριλαμβανομένων των συντακτικών και σημασιολογικών Διδακτική της Πληροφορικής κανόνων. Αυτά περιλαμβάνουν σχεδιασμούς που οδηγούν εύκολα σε λάθη ή περιλαμβάνουν δυσκολονόητες διαδικασίες.

Η *τέταρτη περιοχή* δυσκολίας (*Structures*) περιγράφει ένα απαραίτητο στοιχείο της μετάβασης από τον αρχάριο στον έμπειρο προγραμματιστή: την εκμάθηση και πραγματική κατάκτηση των δομών, με τέτοιο τρόπο, ώστε μελλοντικά να μπορούν να ανακληθούν εύκολα και να ενσωματωθούν σε μια λύση ενός προβλήματος.

Παραδείγματα τέτοιων δομών είναι ο υπολογισμός ενός αθροίσματος με χρήση βρόχου, ένας αλγόριθμος αναζήτησης, ένας αλγόριθμος ταξινόμησης, κώδικας για την αντιμετάθεση τιμών κ.α. Οι έμπειροι προγραμματιστές έχουν στο μυαλό τους έτοιμη τη λύση προς χρήση ανά πάσα στιγμή για τέτοιες δομές, αφού έχουν αντιμετωπίσει ανάλογες καταστάσεις στο παρελθόν. Οι αρχάριοι, λόγω απειρίας, στερούνται τέτοιας δυνατότητας με αποτέλεσμα να βρίσκει εμπόδια η προσπάθειά τους να λύσουν ανάλογα προβλήματα.

Η *πέμπτη περιοχή* δυσκολίας (**Pragmatics**) παραμελείται συχνά στα μαθήματα προγραμματισμού, παρά την προφανώς ιδιαίτερη σημασία της. Έχει να κάνει με τις βοηθητικές δεξιότητες που είναι απαραίτητες για τον προγραμματισμό και αφορούν την ικανότητα προσαρμογής και ελέγχου ενός περιβάλλοντος στον υπολογιστή που θα χρησιμοποιηθεί για τη συγγραφή κώδικα, τη μεταγλώττιση και αποσφαλμάτωση των λαθών ενός προγράμματος. Οι μαθητές ορισμένες φορές έχουν δυσκολία να εγκλιματισθούν στο περιβάλλον ανάπτυξης προγραμμάτων και να μάθουν να χρησιμοποιούν τα διαθέσιμα εργαλεία, πριν ακόμα αρχίσουν να εξετάζουν την ίδια τη γλώσσα προγραμματισμού.

## **2.6 Η κλασσική μέθοδος εισαγωγής στον προγραμματισμό**

Η πιο διαδεδομένη μέθοδος εισαγωγής στον προγραμματισμό είναι η σταδιακή παρουσίαση των δομών μιας γλώσσας προγραμματισμού γενικού σκοπού και η επίλυση προβλημάτων αυξανόμενης δυσκολίας με τη χρήση αυτών των δομών (Brusilovsky et al., 1997). Ωστόσο, η προσέγγιση αυτή κρίνεται ως αναποτελεσματική, ιδιαίτερα για μαθητές μικρής ηλικίας, καθώς θέτει μια σειρά από εμπόδια στους αρχάριους προγραμματιστές (Brusilovsky et al., 1997· Reichert, 2003): α) απαιτείται από το μαθητή να εξοικειωθεί ταυτόχρονα τόσο με την αυστηρή σύνταξη και τη σημασιολογία της ίδιας της γλώσσας όσο και με τις βασικές αρχές του προγραμματισμού, β) η διδασκαλία και χρήση μιας πλήρους γλώσσας προγραμματισμού στο σχολείο μπορεί να αποβεί πολύ χρονοβόρα, γ) παρέχεται συνήθως περιορισμένη υποστήριξη όσον αφορά στην κατανόηση των βασικών εντολών και δομών ελέγχου της γλώσσας αφού η διαδικασία της εκτέλεσης του προγράμματος παραμένει κρυμμένη από το μαθητή, ενώ η έλλειψη οπτικής ανάδρασης

εμποδίζει την κατανόηση της σημασιολογίας της γλώσσας, δ) οι μαθητές δύσκολα μπορούν να εντοπίσουν και να διορθώσουν λάθη στα προγράμματά τους, ε) τα πρώτα προβλήματα που τίθενται στους μαθητές αφορούν κατά κανόνα στην επεξεργασία αριθμών ή συμβόλων και αποτυγχάνουν να κινήσουν το ενδιαφέρον των μαθητών, ενώ η ανάπτυξη πιο ελκυστικών εφαρμογών απαιτεί την εκμάθηση ενός μεγάλου υποσυνόλου της γλώσσας. Ειδικότερα, όσον αφορά στην παρακίνηση των μαθητών, αν η συγγραφή ενός προγράμματος που εμφάνιζε στην οθόνη τη φράση “Hello world” κινούσε το ενδιαφέρον των μαθητών παλαιότερα, δεν συμβαίνει το ίδιο με τη σημερινή ‘γενιά του Nintendo’ που έλκεται από πολυμεσικά μαθησιακά περιβάλλοντα που θυμίζουν ηλεκτρονικά παιχνίδια (Guzdial & Soloway, 2002).

### 3 Εναλλακτικές Προσεγγίσεις διδασκαλίας του προγραμματισμού

---

Όπως ήδη αναφέρθηκε και στην προηγούμενη ενότητα, ένας από τους σημαντικότερους παράγοντες που έχει διαπιστωθεί ότι αποτελεί πηγή δυσκολιών για την εκμάθηση του προγραμματισμού έγκειται στο γεγονός ότι, όπως φαίνεται, η κλασική προσέγγιση διδασκαλίας (Brusilovsky et al., 1997) είναι ασύμβατη με τις πραγματικές διδακτικές ανάγκες των μαθητών.

Με τον όρο κλασική προσέγγιση διδασκαλίας εννοούμε τη διδασκαλία που συνίσταται:

- στη χρήση μιας γλώσσας γενικού σκοπού (όπως οι Pascal, C, κλπ),
- ενός επαγγελματικού περιβάλλοντος προγραμματισμού για τη γλώσσα αυτή, και
- στην επίλυση ενός συνόλου προβλημάτων επεξεργασίας αριθμών και συμβόλων.

Στη συνέχεια του κεφαλαίου, δίνεται μια σύντομη περιγραφή των χαρακτηριστικών και των πλεονεκτημάτων των εναλλακτικών προσεγγίσεων διδασκαλίας του προγραμματισμού που αναπτύχθηκαν έχοντας ως στόχο τη στήριξη της διδακτικής πράξης και την εξάλειψη των σχετικών διδακτικών προβλημάτων. Επίσης, παρουσιάζονται μερικά από τα πιο αντιπροσωπευτικά εκπαιδευτικά εργαλεία που έχουν αναπτυχθεί στα πλαίσια της κάθε προσέγγισης.

### 3.1 Η μέθοδος των Μικρόκοσμων Προγραμματισμού

Ως εναλλακτική, πιο αποτελεσματική προσέγγιση για την εισαγωγή μαθητών αλλά και φοιτητών στον προγραμματισμό έχουν προταθεί μαθησιακά μίνι-περιβάλλοντα (mini-environments) που βασίζονται σε μίνι-γλώσσες (mini-languages) και μικρόκοσμους (microworlds) (Brusilovsky et al., 1997). Οι μίνι-γλώσσες προγραμματισμού αν και είναι ένας απλός και οπτικά διαισθητικός τρόπος για την εισαγωγή των μαθητών στον προγραμματισμό, αποτελούν ταυτόχρονα και μια ισχυρή μέθοδο διδασκαλίας του προγραμματισμού. Με τη βοήθεια των μίνι-γλωσσών προγραμματισμού επιχειρείται μια εισαγωγή στον προγραμματισμό αλλά και μια πολύ καλή εισαγωγή στην Επιστήμη των Η/Υ, αφού οι βασικές αρχές του προγραμματισμού αποτελούν τη βάση για τη θεμελίωση του λογικού και του αφαιρετικού συλλογισμού που είναι θεμελιώδεις έννοιες στην επιστήμη της Πληροφορικής.

Η βασική ιδέα των μικρόκοσμων προγραμματισμού στηρίζεται στην κατασκευή μιας μικρής και απλής γλώσσας. Σε πολλές περιπτώσεις ο μικρόκοσμος έχει επικρατήσει να δηλώνει το συνδυασμό ενός πρωταγωνιστή (actor), ο οποίος ενεργεί στον μικρόκοσμο, και μιας γλώσσας, που συνήθως αναφέρεται σε μικρό σύνολο εντολών, οι οποίες ελέγχουν τη λειτουργία του πρωταγωνιστή. Στις πιο πολλές υπάρχουσες μίνι-γλώσσες ο πρωταγωνιστής είναι συνήθως μια χελώνα (όπως οι παραλλαγές της Logo), ένα ρομπότ (Karel, StageCreator) ή οποιαδήποτε άλλη οντότητα που μπορεί να λειτουργήσει μέσα στον μικρόκοσμο. Οι περισσότερες μίνι-γλώσσες περιλαμβάνουν όλες τις βασικές δομές μιας κλασσικής γλώσσας προγραμματισμού όπως οι δομές ελέγχου (δομές διακλάδωσης, επαναληπτικές δομές κτλ) και ένα μηχανισμό που επιτρέπει την κατασκευή νέων εντολών (Brusilovsky et al., 1994· Kelleher & Pausch, 2005).

Η επιλογή μιας μίνι-γλώσσας, για μια εύκολη εισαγωγή στον προγραμματισμό, παρουσιάζει πολλά πλεονεκτήματα. Ενδεικτικά κάποια από αυτά παρουσιάζονται παρακάτω (Brusilovsky et al., 1997· Alessi & Trollip, 2001· Costelloe, 2004· Kelleher & Pausch, 2005):

- Μια μίνι-γλώσσα έχει μικρό συντακτικό και απλή σημασιολογία. Επομένως, οι μαθητές μπορούν γρήγορα να τη μάθουν και να τη χρησιμοποιήσουν με ενδιαφέροντα αποτελέσματα, επενδύοντας το χρόνο τους σε σημαντικότερα

ζητήματα, όπως η κατανόηση προγραμματιστικών δομών και αρχών, η ανάπτυξη αλγορίθμων και η σχεδίαση προγραμμάτων.

- Το όλο προγραμματιστικό περιβάλλον είναι κτισμένο πάνω σε κάποια οπτικά ελκυστική και παρακινητική για τους μαθητές μεταφορά (metaphor) και επιτρέπει στον εκπαιδευτικό να δημιουργήσει ενδιαφέροντα προβλήματα που σχετίζονται με τις καθημερινές εμπειρίες των μαθητών.
- Οι διάφορες ενέργειες που εκτελεί η οντότητα προκαλούν ορατές αλλαγές στο μικρόκοσμο που αναπαρίσταται στην οθόνη, πράγμα που βοηθά τον αρχάριο προγραμματιστή να αντιληφθεί τι κάνει το πρόγραμμά του και να κατανοήσει τη σημασιολογία των διαφόρων δομών της γλώσσας.
- Τα προβλήματα που συνοδεύουν το περιβάλλον μοιάζουν περισσότερο με σπαζοκεφαλιές παρά με «σοβαρά» προβλήματα και η δραστηριότητα της επίλυσής τους γίνεται ένα είδος παιχνιδιού για τους μαθητές.
- Προάγεται η δημιουργικότητα των μαθητών καθώς και η εποικοδομητική (constructivist) μάθηση μέσα από τον ενεργό πειραματισμό.
- Παρέχεται στους μαθητές η δυνατότητα να οικοδομούν νοητικά μοντέλα και να αναπτύσσουν στρατηγικές επίλυσης προβλημάτων που είναι πιθανό να μεταφερθούν αργότερα σε άλλα πλαίσια.

Οι πρώτες προσπάθειες για μίνι-γλώσσες ξεκίνησαν με τη χελώνα της Logo (Papert, 1980), αν και η Logo δεν σχεδιάστηκε για εισαγωγικά μαθήματα προγραμματισμού. Ακολούθησε το Ρομπότ Karel (Pattis, 1981 & 1995). Από τότε έχουν παρουσιαστεί και άλλοι μικρόκοσμοι, όπως: Josef the robot (Tomek, 1983), Turingal (Brusolovsky, 1994), Karel-3D (Hvorecky, 1992), Karel++ (Bergin, 1996).

Όλες οι προσπάθειες που αναφέραμε παραπάνω επικεντρώθηκαν στην κατασκευή μίνι-γλωσσών προγραμματισμού και στην επινόηση ενός συνόλου προβλημάτων που στόχο έχουν να διδάξουν την ανάπτυξη αλγορίθμων, το σχεδιασμό προγράμματος και την αποσφαλμάτωση των προγραμμάτων, προκαλώντας το ενδιαφέρον του μαθητή. Ωστόσο, θα πρέπει να σημειωθεί ότι σε ελάχιστες μόνο περιπτώσεις έγινε κάποια προσπάθεια εκμετάλλευσης της τεχνολογίας ώστε να ενσωματωθούν στα αναπτυσσόμενα εκπαιδευτικά περιβάλλοντα λειτουργίες που θα διευκόλυναν τη



διδασκτική χρήση τους. Για παράδειγμα, σε λίγους σχετικά μικρόκοσμους προγραμματισμού υπάρχουν τμήματα για την αξιολόγηση των μαθητών ή την καταγραφή των ενεργειών τους, προκειμένου ο καθηγητής να διαπιστώσει τις φάσεις από τις οποίες πέρασε ο κάθε μαθητής στην πορεία λύσης ενός προβλήματος.

Συμπερασματικά, οι διδασκτικοί μικρόκοσμοι ανοίγουν μια νέα προοπτική στην εκπαιδευτική διαδικασία όχι μόνο ως μέσο διδασκαλίας εισαγωγικών εννοιών της Πληροφορικής αλλά και ως μέσο αξιολόγησης και μελέτης των αντιλήψεων των μαθητών.

### 3.1.1 Λογισμικά δημιουργίας μικρόκοσμων

#### Microworlds EX

Το Microworlds Logo είναι λογισμικό δημιουργίας μικρόκοσμων το οποίο έχει σαν βάση τη γλώσσα προγραμματισμού Logo (Papert, 1980) η οποία χρησιμοποιείται αρκετά συχνά από τους μαθητές για την κατασκευή γεωμετρικών σχημάτων. Είναι λογισμικό φραστικών εντολών (ανοικτής γλώσσας), το οποίο έχει ήδη μεταφραστεί στην ελληνική γλώσσα. Το Microworlds Logo μπορεί να χρησιμοποιηθεί για την ανάπτυξη προσομοιώσεων καθώς υποστηρίζει γραφικές αναπαραστάσεις, animation και αλληλεπιδράσεις μεταξύ χαρακτήρων.

Για την οικοδόμηση μοντέλων / προσομοιώσεων στο Microworlds Logo οι χρήστες χρειάζεται να πληκτρολογούν οδηγίες για τους χαρακτήρες (τη χελώνα) τις οποίες θα πρέπει να ακολουθούν οι χαρακτήρες έτσι ώστε να κινούνται στο δισδιάστατο χώρο της οθόνης. Η χελώνα μπορεί να έχει χαρακτηριστικά όπως μάζα και ταχύτητα, και να αλληλεπιδρά με άλλες χελώνες. Σε αντίθεση με τα παραδοσιακά περιβάλλοντα Logo, το Microworlds Logo μπορεί να συμπεριλάβει γραφικές αναπαραστάσεις, ήχο, εικόνες, διαφορετικά σχήματα για τις χελώνες, ακόμη και την δυνατότητα δημιουργίας animations. Για τη χρήση του Microworlds Logo, ο χρήστης πρέπει να γνωρίζει τις φραστικές εντολές του λογισμικού. Ωστόσο, μπορεί να φτιάξει συνδυασμούς εντολών τους οποίους το πρόγραμμα να αναγνωρίζει ως διεργασίες. Συνεπώς είναι εφικτό οι μαθητές να προγραμματίζουν αντικείμενα για να επιδεικνύουν συγκεκριμένες συμπεριφορές. Το Σχήμα 1 παρέχει ένα παράδειγμα προγράμματος στο Microworlds Logo.



Σχήμα 1: Παράδειγμα προγράμματος στο Microworlds Logo.

### Επίσημη Ιστοσελίδα Microworlds Logo:

<http://www.microworlds.com/solutions/mwex.html>

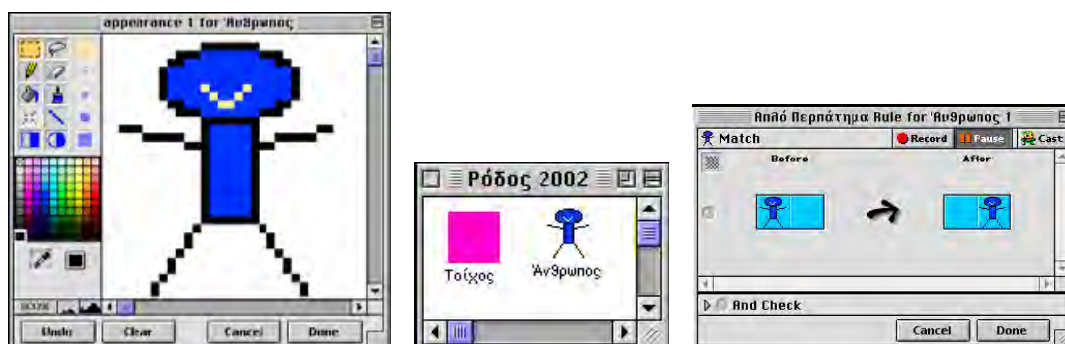
### Demo Version of Microworlds Logo:

[http://www.microworlds.com/solutions/demo\\_ex.html](http://www.microworlds.com/solutions/demo_ex.html)

## Cocoa

Το Cocoa (Gilmore et al, 1995) αποτελεί ένα προγραμματιστικό περιβάλλον, στο οποίο τα παιδιά μπορούν να δημιουργήσουν απλά έργα (projects), ορίζοντας διαφορετικούς χαρακτήρες και τοποθετώντας τους σε ένα σκηνικό.

Στη συνέχεια, και με πλήρως γραφικό τρόπο, μπορούν να ορίσουν μεταβάσεις (κινήσεις) των χαρακτήρων αυτών, ανάλογα με την ικανοποίηση συγκεκριμένων προϋποθέσεων. Στο Σχήμα 2 αναπαριστώνται οι φάσεις δημιουργίας και καθορισμού κανόνων ενός απλού χαρακτήρα στο περιβάλλον Cocoa. Το Σχήμα 2.α επιδεικνύει τις δυνατότητες καθορισμού της όψης ενός χαρακτήρα, μέσα από απλά εργαλεία σχεδίασης. Για τις ανάγκες του παραδείγματος ζωγραφίσαμε δύο χαρακτήρες (τοιχος, άνθρωπος), που φαίνονται στο Σχήμα 2.β και θα τους χρησιμοποιήσουμε για να κατασκευάσουμε το μοντέλο ενός κινούμενου ανθρώπου μέσα σε ένα σκηνικό, που περιλαμβάνει εμπόδια. Ο πρώτος κανόνας, που πρέπει να ορίσουμε, είναι η κίνηση του ανθρώπου. Μέσα από το παράθυρο, που φαίνεται στο Σχήμα 2.γ μπορούμε να ορίσουμε, με κινήσεις του ποντικιού, τον εξής απλό κανόνα: "αν δεξιά του ανθρώπου υπάρχει κενό πλαίσιο, τότε ο άνθρωπος πρέπει να μετακινηθεί στο πλαίσιο αυτό". Η εικόνα αναπαριστά γραφικά αυτόν τον κανόνα, επιτρέποντας τη συνεχή δεξιά οριζόντια κίνηση του ανθρώπου όσο δεν συναντά εμπόδια.



Σχήμα 2: Ορισμός Χαρακτήρων και Κανόνων στο Περιβάλλον Cocoa

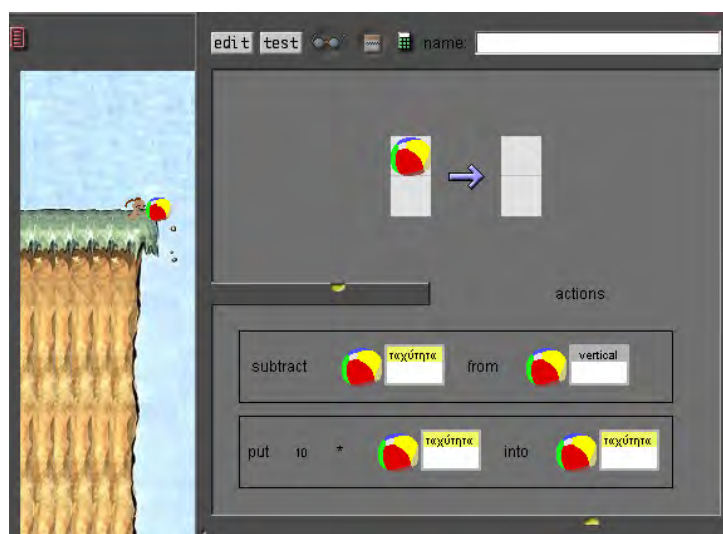
## Stagecast Creator

Το Stagecast Creator (Smith & Cypher, 1999) είναι λογισμικό δημιουργίας μικρόκοσμων σχεδιασμένο για μαθητές μικρής ηλικίας που επιτρέπει την οικοδόμηση συμβολικών προσομοιώσεων. Αποτελεί εξέλιξη του προηγούμενου λογισμικού Cocoa, με μόνη, αλλά σημαντική διαφορά την εμπορική διαθεσιμότητά του. Πρόκειται, λοιπόν, για ένα γραφικό σύστημα αλληλεπίδρασης υπολογιστή - μαθητών, το οποίο προσφέρει :

- αναλογική αναπαράσταση αντικειμένων,
- διαφοροποίηση μεταξύ φυσικών μεγεθών,
- χρήση αντικειμενοστραφούς διακείμενου (object oriented interface),
- οικοδόμηση μικρόκοσμων, μέσω επίδειξης, με τη χρήση μόνο ποντικιού χωρίς να είναι απαραίτητη η χρήση γλώσσας προγραμματισμού.

Το Stagecast Creator χρησιμοποιεί αντικειμενοστραφές διακείμενο που επιτρέπει την κατασκευή συμβολικών προσομοιώσεων, με σκοπό να μειώσει τα προβλήματα που σχετίζονται με την εκμάθηση μιας νέας γλώσσας προγραμματισμού. Ο προγραμματισμός γίνεται μέσω του άμεσου χειρισμού των αντικειμένων, ορίζοντας κανόνες συμπεριφοράς αντικειμένων με τη μορφή "Αν... Τότε...": για μια δεδομένη κατάσταση, καθορίζεται μια συμπεριφορά.

Κατά τη διάρκεια της δημιουργίας κανόνων οι ενέργειες του χρήστη καταγράφονται από τον υπολογιστή, δημιουργώντας ένα σενάριο από κανόνες. Το Σχήμα 3 είναι ένα παράδειγμα ενός προγράμματος στο Stagecast Creator.



Σχήμα 3: Παράδειγμα προγράμματος στο Stagecast Creator

**Επίσημη Ιστοσελίδα Stagecast Creator:** <http://www.stagecast.com/>  
**Evaluation Version of Stagecast Creator:** <http://www.stagecast.com/cgi-bin/templator.cgi?PAGE=Shared/software/SOFTWARE#eval>  
**Ελληνική έκδοση Tutorial Stagecast Creator με οδηγίες εγκατάστασης:**  
<http://www.stagecast.com/cgi-bin/templator.cgi?PAGE=Shared/software/SOFTWARE#greek>

## **ToonTalk™**

Το περιβάλλον ToonTalk™ (Kahn, 2000), (Morgado et al, 2001) είναι το πιο απαλλαγμένο από κείμενα (αν και όχι ολοκληρωτικά) και έχει σαφή κατεύθυνση σε μικρές ηλικίες. Τόσο τα γραφικά του σύμβολα (μεγάλα και απλά), όσο και ο τρόπος χειρισμού του (ίδιες αποκρίσεις σε διαφορετικά πλήκτρα του ποντικιού), απευθύνονται σε παιδιά προσχολικών ή πρώτων σχολικών ηλικιών. Επιπροσθέτως, το "πεδίο δράσης" του ποντικιού γίνεται άμεσα σαφές, με ένα "τρεμόπαιγμα" των αντικειμένων, που βρίσκονται υπό την επίδρασή του. Στο Σχήμα 4 βλέπουμε τα βασικά εργαλεία που μπορεί να χρησιμοποιήσει ο χρήστης.



**Σχήμα 4:** Το Περιβάλλον Προγραμματισμού ToonTalk™

**Επίσημη Ιστοσελίδα ToonTalk:** [www.toontalk.com](http://www.toontalk.com)

**Demo Version of ToonTalk:** <http://www.toontalk.com/English/free.htm>

## Alice

Το πρόγραμμα Alice δημιουργήθηκε από μια ομάδα ερευνητών του πανεπιστημίου Carnegie Mellon (2003) υπό την καθοδήγηση του Randy Pausch. Είναι ένα τρισδιάστατο (3D) διαδραστικό περιβάλλον το οποίο χρησιμοποιεί δυναμική απεικόνιση με κίνηση (animation). Το Alice υποστηρίζει τον αντικειμενοστραφή προγραμματισμό. Η τριών διαστάσεων απεικόνιση “ενισχύει” την οπτικοποίηση των αντικειμένων παρέχοντας μια αίσθηση πραγματικότητας γι’ αυτά και παρέχει ένα ευέλικτο και εποικοδομητικό πλαίσιο για την κατανόηση των αντικειμενοστραφών εννοιών. Ουσιαστικά ακολουθεί την παράδοση των μικρόκοσμων Karel παρέχοντας μια επιπλέον διάσταση. Οι τριών διαστάσεων κόσμοι είναι περισσότερο ρεαλιστικοί από τους αντίστοιχους δισδιάστατους. Επίσης, αξιοσημείωτο είναι το γεγονός ότι περιέχει εκατοντάδες έτοιμα υλοποιημένα τρισδιάστατα μοντέλα αντικειμένων. Ένα δείγμα αυτών βλέπουμε στο Σχήμα 5.

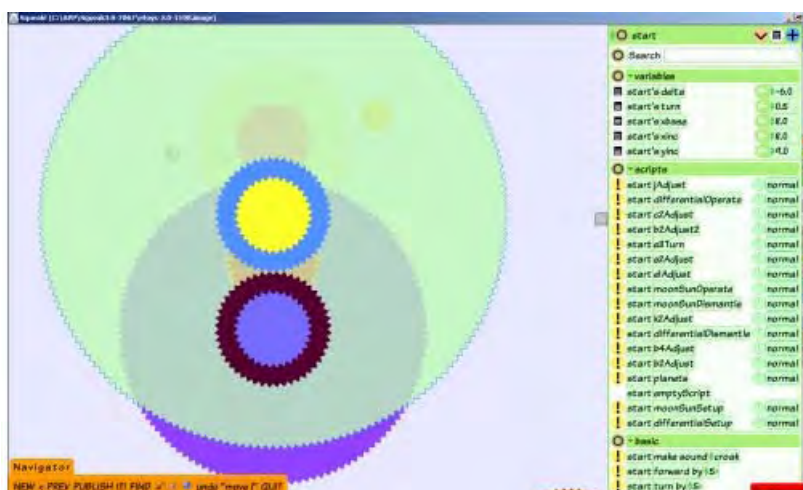


**Σχήμα 5:** Τρισδιάστατα μοντέλα αντικειμένων

**Επίσημη Ιστοσελίδα Alice:** [www.alice.org](http://www.alice.org)

## Squeak Etoys

Το Squeak Etoys (1997) έχει τις ρίζες του στη LOGO, στην PARC-Smalltalk, στην Hypercard και στην StarLOGO. Είναι ένα αντικειμενοστραφές προγραμματιστικό περιβάλλον πλούσιο σε εργαλεία ανάπτυξης, όπου τα αντικείμενα μπορούν να προγραμματιστούν είτε με έτοιμα script είτε με script που μπορεί να δημιουργήσει ο χρήστης στη γλώσσα προγραμματισμού Smalltalk-80. Είναι ελεύθερο και ανοικτό λογισμικό. Τρέχει σε πολλές πλατφόρμες (Windows, MacOS, Linux), υποστηρίζει δισδιάστατα και τρισδιάστατα γραφικά, εικόνες, κείμενο, παρουσιάσεις τύπου PowerPoint, βίντεο και ήχο. Επιπλέον, παρέχει τη δυνατότητα να μοιράζεται κάποιος χρήστης την επιφάνεια εργασίας του σε πραγματικό χρόνο με κάποιον ή κάποιους άλλους χρήστες που βρίσκονται οπουδήποτε στον κόσμο, προσφέροντας έτσι δυνατότητες συνεργασίας και αλληλοβοήθειας. Το Σχήμα 6 παρέχει ένα παράδειγμα του προγράμματος στο Squeak Etoys.



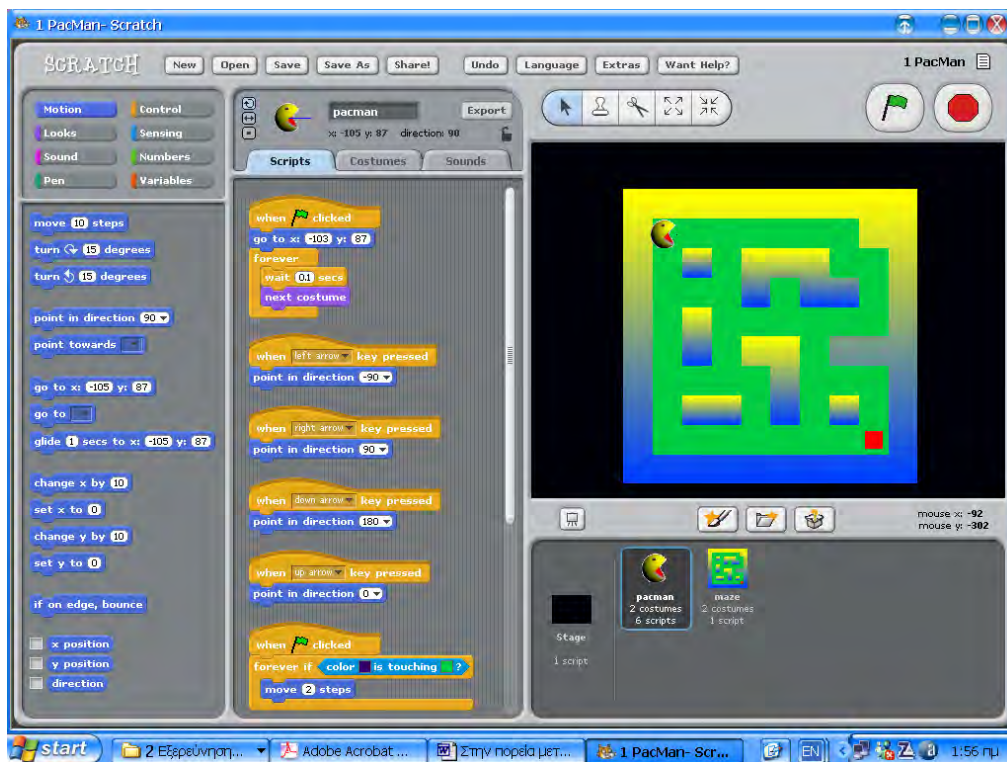
Σχήμα 6: Παράδειγμα προγράμματος από το Squeak Etoys

Επίσημη Ιστοσελίδα Squeak: [www.squealand.org](http://www.squealand.org)

## Scratch

Το Scratch αναπτύχθηκε από το Lifelong Kindergarten group στο MIT Media Lab και πήρε το όνομά του από την τεχνική των DJs (scratching). Είναι ένα πλούσιο σε οπτικοακουστικά μέσα προγραμματιστικό περιβάλλον στο οποίο οι αρχάριοι

προγραμματιστές μπορούν να εκφράσουν την δημιουργικότητά τους ενώ μαθαίνουν να σκέφτονται υπολογιστικά όπως θα δούμε και σε επόμενο κεφάλαιο. Το Scratch είναι γραμμένο στο Squeak (www.squeak.org), μία ανοιχτή εφαρμογή της γλώσσας Smalltalk-80 και γενικά θα μπορούσαμε να πούμε ότι εμπνέεται από άλλα περιβάλλοντα προγραμματισμού, που έχουν σχεδιαστεί για νέους και αρχάριους προγραμματιστές. Το Σχήμα 7 παρέχει ένα παράδειγμα του προγράμματος στο Scratch.



Σχήμα 7: Παράδειγμα υλοποίησης του παιχνιδιού Pac-Man στο Scratch

Επίσημη Ιστοσελίδα Scratch: <http://scratch.mit.edu>



## **3.2 Η μέθοδος του προγραμματισμού ρομποτικών κατασκευών**

### **3.2.1 Εκπαιδευτική Ρομποτική**

Στο πλαίσιο της εκπαιδευτικής ρομποτικής μπορούμε να διακρίνουμε τρεις τουλάχιστον επιμέρους παιδαγωγικές προσεγγίσεις. Μια πρώτη προσέγγιση συνδέεται άμεσα με την ανάπτυξη και την περιγραφή τεχνικών καταστάσεων με τη βοήθεια γλωσσών εντολών (τυπικές γλώσσες προγραμματισμού) και αντιστοιχεί στην προβληματική της Τεχνολογίας Ελέγχου (Control Technology). Μια δεύτερη παιδαγωγική προσέγγιση έρχεται απευθείας από την παιδαγωγική παράδοση της Logo, με τη δημιουργία ποικίλων μικρόκοσμων (που απαιτούν ύπαρξη αυτομάτων με πιο χαρακτηριστικό παράδειγμα την προγραμματιζόμενη «χελώνα» εδάφους), οι οποίοι χρησιμοποιούνται μέσα σε διάφορες παιδαγωγικές καταστάσεις με σημασία και νόημα για τους μαθητές. Μια τρίτη προσέγγιση αφορά στη χρήση της παιδαγωγικής ρομποτικής ως ενός εναλλακτικού τρόπου εκμάθησης του προγραμματισμού (κυρίως όσον αφορά στην αλγοριθμική προσέγγιση) κάτω από το πρίσμα της ανάπτυξης της οργάνωσης της σκέψης μέσω πρόβλεψης για τη μετακίνηση αντικειμένων μέσα στο χώρο (Miglino, Hautop & Cardaci, 1999· Κόμης, 2004).

Σε κάθε περίπτωση, η ενασχόληση με τη ρομποτική ενέχει δύο ειδών δραστηριότητες: μια κατασκευαστική και μια προγραμματιστική (Κόμης, 2005). Η παρούσα ενότητα εστιάζεται στην προγραμματιστική δραστηριότητα που αφορά σχεδίαση και υλοποίηση ρομποτικών εφαρμογών από μαθητές μικρής ηλικίας.

### **3.2.2 Ο προγραμματισμός ρομποτικών κατασκευών**

Ο προγραμματισμός ρομποτικών κατασκευών έχει μια ιδιαιτερότητα σε σχέση με τον προγραμματισμό σε άλλες συνθήκες ή καταστάσεις. Ταυτίζεται με την απόδοση συμπεριφοράς σε μια τεχνητή κατασκευή. Η τεχνητή κατασκευή δημιουργείται από τους μαθητές αξιοποιώντας ένα σύνολο δομικών υλικών, όπως για παράδειγμα τα Lego Mindstorms (<http://mindstorms.lego.com/>). Μπορεί να διαθέτει αισθητήρες για να συλλαμβάνει συμβάντα ή καταστάσεις του περιβάλλοντος (θερμοκρασία, απόσταση από εμπόδιο, ένταση φωτός, επαφή με άλλα αντικείμενα,

κλπ). Μπορεί επίσης να διαθέτει μηχανισμό κίνησης (μοτέρ) που θέτει σε κίνηση ολόκληρη την κατασκευή ή ένα τμήμα της. Μια τυπική συμπεριφορά της ρομποτικής κατασκευής είναι η αντίδραση σε ένα πιθανό ερέθισμα. Πρόκειται για ένα χαρακτηριστικό ξεκάθαρα ανθρωπομορφικό και δεν είναι τυχαίο που πολλοί ερευνητές προσπάθησαν να μελετήσουν συμπεριφορές ζώντων οργανισμών ή βιολογικών συστημάτων με τη βοήθεια των ρομποτικών κατασκευών (Chiocciariello et al, 2000).

Αυτή η ιδιαιτερότητα στον προγραμματισμό των ρομποτικών κατασκευών δημιουργεί ένα εντελώς νέο περιβάλλον εργασίας για τους μαθητές με τα εξής χαρακτηριστικά (Κόμης, 2005):

- α) Είναι έντονα παρακινητικό, και συνεπώς παράγοντας υψίστης σημασίας για τη διδακτική.
- β) Έχει άμεση σύνδεση με κοινωνικές πρακτικές αναφοράς δεδομένου ότι η κατασκευή διαφόρων αντικειμένων συνιστά πλέον διαδεδομένη κοινωνική πρακτική ακόμα και στον κόσμο των παιδιών. Οι συμπεριφορές προκύπτουν από μεταφορά υπάρχοντων και ήδη γνωστών συμπεριφορών από τους ζώντες οργανισμούς.
- γ) Ευνοεί τη στρατηγική δοκιμής – πλάνης, που είναι στρατηγική οικεία στους μαθητές του δημοτικού.
- δ) Αναδεικνύει παραδεκτές προσεγγίσεις και λύσεις και όχι μια και μοναδική σωστή λύση αφού μια συμπεριφορά μπορεί να αποδοθεί με πολλούς τρόπους.
- ε) Υποστηρίζει μεταγνωστικές διεργασίες μάθησης, δεδομένου ότι η προγραμματιστική δραστηριότητα οδηγεί στη συγκρότηση, την ανάλυση και την εξωτερίκευση νοητικών διεργασιών. Αυτή η προσπάθεια έχει μεταγνωστικό χαρακτήρα αφού μας αναγκάζει να σκεφτούμε πάνω στον τρόπο που σκεφτόμαστε και ενεργούμε.

### **3.2.3 LEGO® Mindstorms™: Η πιο δημοφιλής ρομποτική**

#### **κατασκευή**

Το πιο γνωστό και διαδεδομένο προϊόν αυτής της κατηγορίας είναι με διαφορά τα LEGO® Mindstorms™. Πρόκειται για προϊόν αποτέλεσμα της συνεργασίας του Papert με την εταιρεία παιχνιδιών LEGO (LEGO Mindstorms 2005).

Ο μικροεπεξεργαστής έχει ενσωματωθεί σε ένα τουβλάκι Lego και λέγεται LEGO RCX. Διαθέτει τρεις εισόδους και τρεις εξόδους στις οποίες συνδέονται κινητήρες, λαμπτήρες και διάφοροι αισθητήρες που επίσης έχουν τη μορφή τούβλων της LEGO. Με τη βοήθεια διαφόρων δομικών στοιχείων (μεταξύ των οποίων γρανάζια, μάντες και άξονες) είναι δυνατό να κατασκευαστούν πολύπλοκες ρομποτικές διατάξεις, ο προγραμματισμός των οποίων αποτελεί αυθεντική πρόκληση. Υπάρχει ακόμα η δυνατότητα επικοινωνίας των RCX μεταξύ τους επεκτείνοντας ακόμα περισσότερο τη γκάμα και το ενδιαφέρον των εφαρμογών.

Το RCX είναι δυνατό να προγραμματιστεί με διάφορους τρόπους. Η LEGO προτείνει το περιβάλλον ROBOLAB (Τσοβόλας και Κόμης 2005) ενώ υπάρχουν και διάφορα άλλα βασισμένα στην JAVA, τη C (π.χ NQC), κ.α. Όπως έχει άλλωστε αναφερθεί παραπάνω, υπάρχει ειδική έκδοση του microworlds EX της LCSΙ για τον προγραμματισμό του RCX. Στην Ελλάδα το συγκεκριμένο προϊόν αντιπροσωπεύεται από την εταιρεία «Διερευνητική Μάθηση» [<http://www.why.gr>].



Ρομπότ κατασκευασμένο  
με το σύστημα LEGO Mindstorms

### 3.2.3.1 Το λογισμικό ρομποτικής Lego Robolab

Το Lego Robolab είναι λογισμικό ρομποτικής το οποίο σχεδιάστηκε από το Πανεπιστήμιο Tufts για να χρησιμοποιείται με το Lego RCX τσιπ, για την οικοδόμηση και προγραμματισμό ρομπότ από Lego. Το Lego Robolab είναι λογισμικό γραφικών εντολών, το οποίο περιέχει προγραμματισμένες εντολές σε μορφή εικονιδίων, τις οποίες οι μαθητές μπορούν να χρησιμοποιήσουν για δημιουργία προγραμμάτων για το ρομπότ. Έτσι, οι μαθητές οικοδομούν απλά προγράμματα με τη σύνθεση δοσμένων εντολών στη μορφή εικονιδίων (χωρίς να

χρειάζονται να μάθουν οποιαδήποτε γλώσσα προγραμματισμού), τα οποία εκτελεί το ρομπότ που κινείται στον πραγματικό Νευτώνειο χώρο (και όχι σε μικρόκοσμο στην οθόνη του υπολογιστή). Το Σχήμα 8 παρέχει ένα παράδειγμα του προγράμματος στο Lego Robolab.



**Σχήμα 8:** Παράδειγμα προγράμματος από το Lego Robolab

**Επίσημη Ιστοσελίδα Lego Robolab:**

<http://www.lego.com/eng/education/mindstorms/home.asp?pagename=robolab>

**Trial Version of Lego Robolab:**

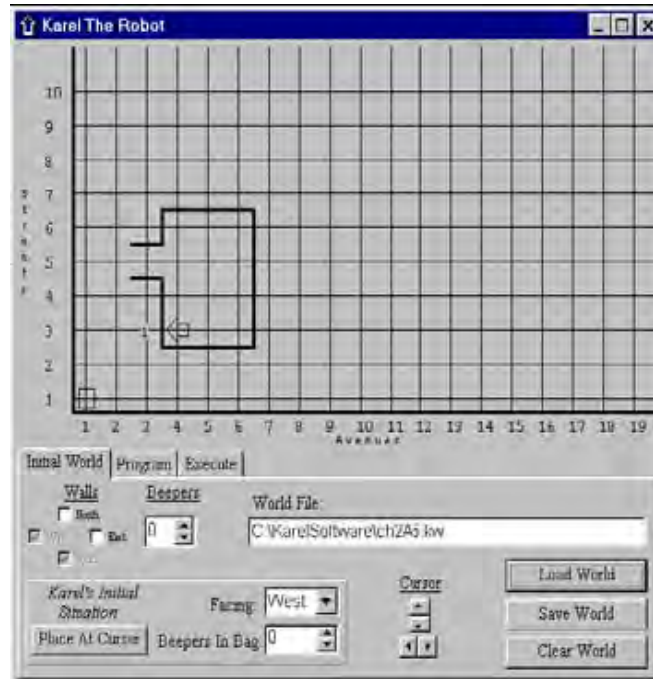
<http://www.lego.com/eng/education/mindstorms/home.asp?pagename=software>

### 3.2.4 Τα ρομποτάκια Karel και CeeBot

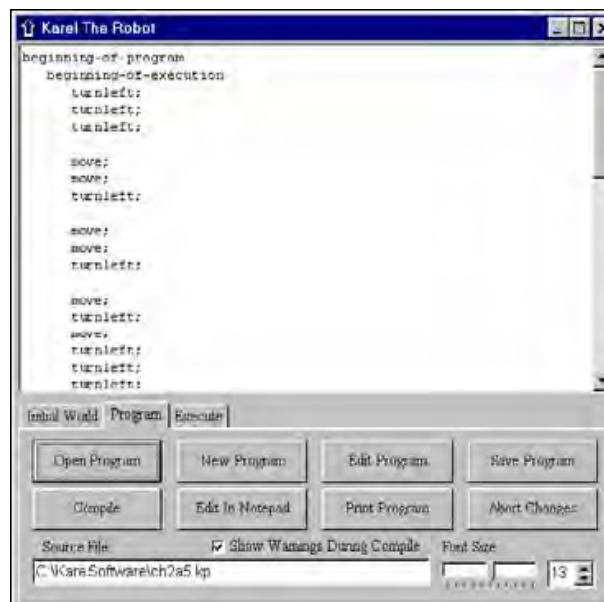
#### Karel the Robot

Το robot Karel σχεδιάστηκε από τον Richard Pattis (Pattis, 1995), προκειμένου να χρησιμοποιηθεί για το μάθημα της εισαγωγής στον προγραμματισμό. Ο Karel είναι ένα ρομπότ που εκτελεί διάφορες αποστολές (προγράμματα) σε ένα κόσμο που αποτελείται από οριζόντιους δρόμους και κατακόρυφες λεωφόρους. Στον κόσμο του Karel μπορούν να τοποθετηθούν τμήματα τοίχου μεταξύ των διασταυρώσεων δημιουργώντας εμπόδια (π.χ. λαβύρινθους), που καλείται να ξεπεράσει ο Karel, και beepers, μικροί πλαστικοί κώνοι, που παράγουν ένα ήχο (μπιπ). Ο Karel έχει τη δυνατότητα να κινείται προς την τρέχουσα κατεύθυνση κατά 1 μπλοκ, να στρίβει αριστερά κατά 90 μοίρες, να εντοπίζει beepers που βρίσκονται στην ίδια διασταύρωση μ' αυτόν, να εντοπίζει τοίχους που βρίσκονται μπροστά του σε απόσταση μισού μπλοκ χρησιμοποιώντας μια κάμερα, να σηκώνει και να κατεβάζει beepers με το μηχανικό του χέρι και τέλος μπορεί να καθορίζει προς ποια κατεύθυνση βλέπει χρησιμοποιώντας την πυξίδα του. Ο Karel μπορεί και

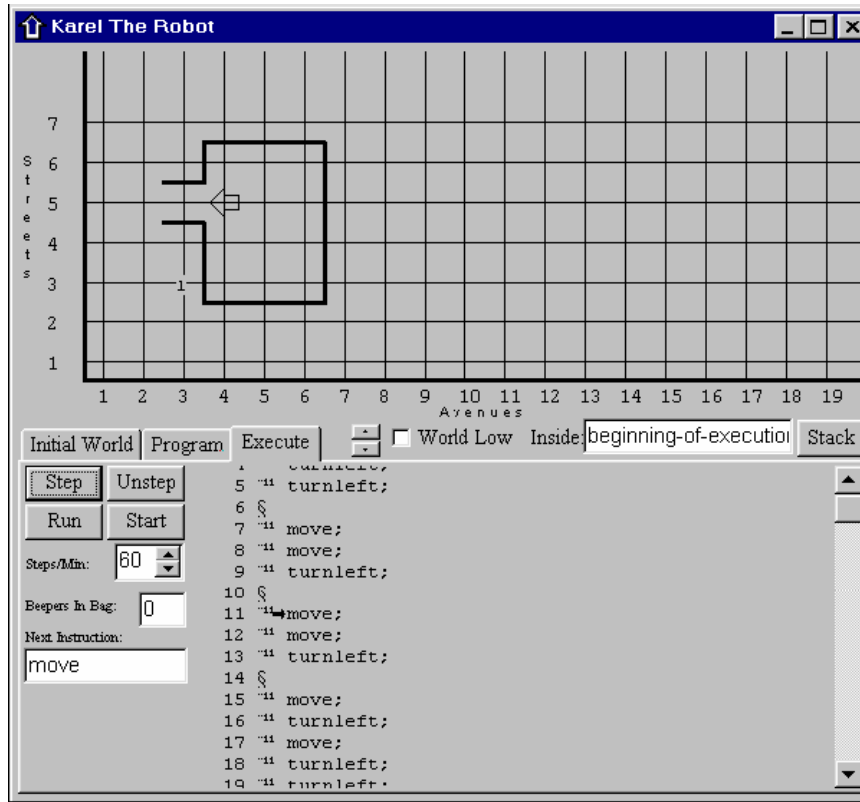
ανταποκρίνεται στις εξής 5 βασικές εντολές: move, turnleft, pickbeeper, putbeeper τις οποίες χρησιμοποιούμε για να «διατάξουμε» τον Karel να κινηθεί μέσα στον κόσμο και να πραγματοποιήσει αποστολές. Τα Σχήματα 9, 10, 11 είναι ένα παράδειγμα ενός προγράμματος στο Karel.



Σχήμα 9: Ο συντάκτης του κόσμου του Karel



Σχήμα 10: Ο συντάκτης του προγράμματος του Karel



**Σχήμα 11:** Η εκτέλεση ενός προγράμματος

Ως φυσική εξέλιξη και προσαρμογή του Karel εμφανίσθηκαν παρόμοια περιβάλλοντα για να καλύψουν τις ανάγκες νέων υποδειγμάτων προγραμματισμού που διαδέχθηκαν τον δομημένο διαδικαστικό (imperative) προγραμματισμό που πρεσβεύει η PASCAL.

Ειδικότερα εμφανίσθηκαν με χρονολογική σειρά τα περιβάλλοντα Karel++ (Bergin et al. 1996) (κατά αναλογία με τα ονόματα των γλωσσών C και C++), JKarelRobot (Buck & Stucki 2001), KarelJ (Bergin et al. 2002) και το JEROO (Sanders & Dorn 2003).

## **CeeBot**

Το CeeBot έχει διάφορες εκδόσεις όπως τις CeeBot-Teen και CeeBot-3 για ηλικίες 10-15 ετών και τις CeeBot-A και CeeBot-4 για ηλικίες από 15 ετών και άνω. Η βασική ιδέα είναι και εδώ ο προγραμματισμός λογισμικών ρομπότ-μινιατούρων για την εκτέλεση αποστολών πάνω σε ένα γραφείο, ή σε ένα υπόγειο, κλπ. Οι διάφορες εκδόσεις υποστηρίζουν διάφορες γλώσσες όπως Java και C#. Όσο αφορά στην εκπαιδευτική τεκμηρίωση, σε κάθε προϊόν υπάρχει ένα σύνολο από προβλήματα για

επίλυση, τα οποία είναι οργανωμένα σε ενότητες. Μεταξύ άλλων περιλαμβάνονται δραστηριότητες όπως αγώνες ταχύτητας των ρομπότ, ποδόσφαιρο και απόδραση από λαβύρινθο.

Οι εκδόσεις που απευθύνονται σε ευρύτερες ηλικίες περιγράφονται ως ένα είδος παιχνιδιού σε υπολογιστή. Το CeeBot επιτρέπει την εισαγωγή εννοιών όπως μεταβλητή, επιλογή, επανάληψη, πίνακες, συναρτήσεις-υποπρογράμματα, κλάση, κ.α.



Σχήμα 12: Διεπαφή χρήστη του λογισμικού CeeBot-3

Επίσημη ιστοσελίδα CeeBot: <http://www.ceebot.com>

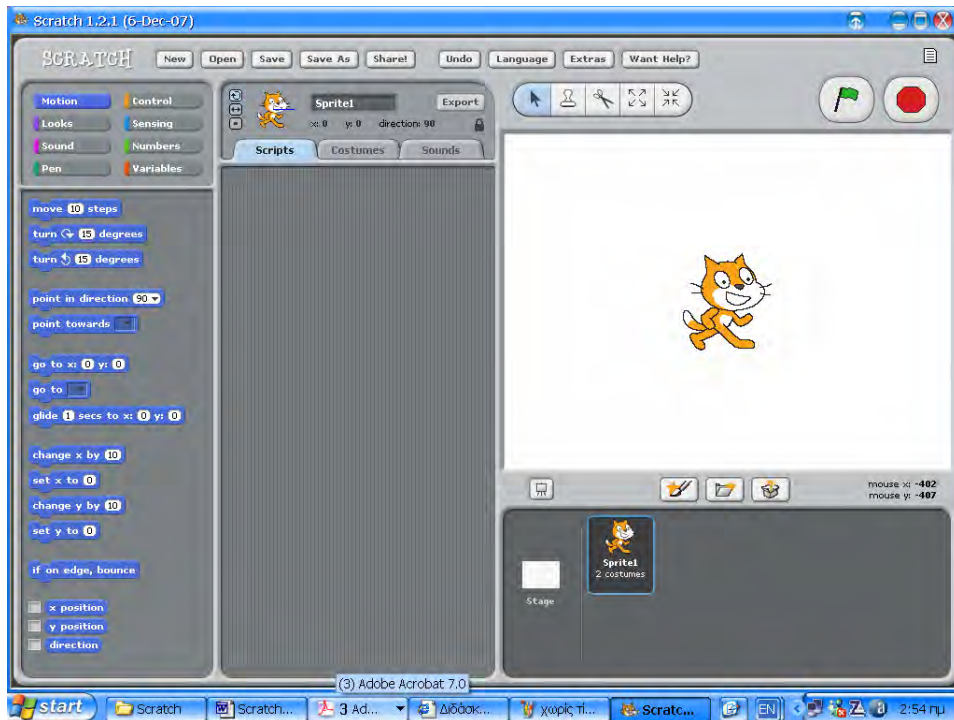
## 4 Το εκπαιδευτικό περιβάλλον προγραμματισμού Scratch

---

### 4.1 Γενική περιγραφή του Scratch

Το Scratch αναπτύχθηκε από την ερευνητική ομάδα Lifelong Kindergarten Group στο MIT Media Lab, με την υποστήριξη του Εθνικού Ιδρύματος Επιστήμης (National Science Foundation) του Ιδρύματος Intel (Intel Foundation) και του MIT Media Lab research consortia. Αποτελεί μια σχετικά νέα γλώσσα προγραμματισμού, σχεδιασμένη για την εκπαίδευση, για χρήση από την ηλικία των 8 ετών (Resnick, 2007). Προσθέτει τον προγραμματισμό στο ήδη πλούσιο σε πολυμέσα και δικτύωση περιβάλλον στο οποίο δρουν οι μαθητές (Maloney et al, 2004). Επιτρέπει στους χρήστες να δημιουργήσουν εύκολα διαδραστικές ιστορίες, κινούμενα σχέδια, ηλεκτρονικά παιχνίδια, μουσική και ψηφιακή τέχνη και ενθαρρύνει το διαμοιρασμό τους (Monroy-Hernández & Resnick, 2008). Η κοινότητα που έχει δημιουργηθεί γύρω από το περιβάλλον Scratch (Εικόνα 1) στο διαδίκτυο δίνει την ευκαιρία ανταλλαγής ιδεών και απόψεων μεταξύ των δημιουργών και ενεργής εμπλοκής σε μια κοινότητα πρακτικής και μάθησης. Οι σχεδιαστές του Scratch στοχεύουν στην ανάπτυξη βασικών ικανοτήτων όπως: *δημιουργική σκέψη, σαφή επικοινωνία, συστηματική ανάλυση, αποδοτική συνεργασία, επαναληπτικό-προοδευτικό σχεδιασμό και δεξιοτήτων δια βίου μάθησης*. Το Scratch υποστηρίζει τις μαθησιακές δεξιότητες του 21ου αιώνα ([www.21stcenturyskills.org](http://www.21stcenturyskills.org)): δεξιότητες συνεργασίας και πληροφορίας (πληροφορίες και δεξιότητες τεχνολογικού εγγραμματισμού, δεξιότητες επικοινωνίας), δεξιότητες σκέψης και επίλυσης προβλημάτων - problem solving (κριτική σκέψη και διαδικασίες σκέψης, προσδιορισμός, διατύπωση και επίλυση προβλήματος, δημιουργικότητα) και διαπροσωπικές και ενδοπροσωπικές-ενδοκεντρικές δεξιότητες (δεξιότητες συνεργασίας, ενδοπροσωπικές - ενδοκεντρικές δεξιότητες, υπευθυνότητα και προσαρμοστικότητα, κοινωνική ευθύνη).





**Εικόνα 1:** Το περιβάλλον εργασίας / προγραμματισμού του Scratch

Οι μαθητές που προγραμματίζουν στο Scratch μπορούν να έρχονται σε επαφή με σημαντικές μαθηματικές και υπολογιστικές ιδέες, ενώ παράλληλα να κατανοούν καλύτερα τη γενική διαδικασία του σχεδιασμού. Ο [Λαπόντες](#) (2008), απαριθμεί τρεις εκδοχές για το τι είναι το Scratch : «...Τι είναι πάλι αυτό το ... Scratch; Πρώτη εκδοχή: *SCRATCH me and I'll Scratch you* (Βοήθα και βοηθώ να ανεβούμε στο βουνό). Δεύτερη εκδοχή: *SCRATCH = LOGO 2.0* δηλαδή νέο περιβάλλον προγραμματισμού για όλους με καταγωγή από τη Logo. Τρίτη εκδοχή: *SCRATCH = προγραμματισμός στο Web 2.0* (κοινωνικό Web) με σύνθημα *Φαντάζομαι-Προγραμματίζω-Επικοινωνώ...*».

## 4.2 Η αποδοχή του Scratch

Η αποδοχή του Scratch από την παγκόσμια κοινότητα ήταν ιδιαίτερα θερμή. Αποτέλεσμα αυτού είναι να έχουν ήδη αναρτηθεί ως τις 10/2/2009 321.802 projects με συνολικά 8.388.869 scripts και 2.530.772 sprites δημιουργημένα από 49.776 συνεισφέροντα από τα 226.203 εγγραμμένα μέλη του σχετικού επίσημου δικτυακού τόπου του M.I.T. (<http://scratch.mit.edu/>). Ήδη το 2008 πραγματοποιήθηκε στο M.I.T. το πρώτο συνέδριο αφιερωμένο στο Scratch

(<http://scratch.mit.edu/conference/>). Το λογισμικό έχει ήδη μεταφραστεί στην ελληνική γλώσσα (το αρχείο μετάφρασης είναι διαθέσιμο στο <https://dev.laptop.org/translate/el/srcatch/srcatch.po> ) και η ελληνική εκπαιδευτική κοινότητα το υποδέχθηκε με ειδική συνεδρία με θέμα «Γνωριμία με το εκπαιδευτικό περιβάλλον προγραμματισμού Scratch» στο 4ο Πανελλήνιο Συνέδριο Διδακτικής της Πληροφορικής και με μετάφραση του οδηγού του (Φεσάκης κ.α., 2008). Στο ιστολόγιό του δε ο Ν. Δαπόντες (Δαπόντες, 2008) έχει αναρτήσει εξαιρετικές δημιουργίες του - εφαρμογές του Scratch σε θέματα των θετικών επιστημών, τις οποίες μοιράζεται παράλληλα με την παγκόσμια κοινότητα, μέσω της ανάρτησής τους στον σχετικό δικτυακό τόπο (<http://srcatch.mit.edu/>).

#### **4.2.1 Εκδηλώσεις-Γιορτές Scratch Day**

Την περασμένη χρονιά οι υπεύθυνοι του Scratch (Media Lab, MIT) διοργάνωσαν μια μεγάλη εκδήλωση στην έδρα τους: ένα συνέδριο με συμμετοχές από πολλές χώρες. Φέτος, αποφασίστηκε να γίνουν εκδηλώσεις αφιερωμένες στο Scratch σε διάφορες χώρες <http://day.scratch.mit.edu/>. Έτσι, οι ενδιαφερόμενοι εκπαιδευτικοί από όλες τις βαθμίδες (κυρίως όσοι συμμετέχουν ενεργά στην «Κοινότητα του Scratch») διοργάνωσαν workshops, μαθήματα για αρχαρίους και ποικίλες παρουσιάσεις που απευθύνονται σε μαθητές, σε εκπαιδευτικούς ή γονείς.

Ένα μήνα περίπου μετά από την επίσημη γιορτή **Scratch Day** μπορούμε να πούμε ότι αριθμός των εκδηλώσεων έφτασε τις 112. Μια ματιά στο χάρτη (Google map <http://day.scratch.mit.edu/>) με τα «σηματάκια» δείχνει ότι οι εκδηλώσεις πραγματοποιήθηκαν κυρίως στις Ηνωμένες Πολιτείες (πολύ φυσικό) και στην Ευρώπη. Κοιτάζοντας πιο αναλυτικά το χάρτη μπορούμε να πούμε ότι η Αφρική απουσίαζε παντελώς ενώ στην Ασία συμμετείχαν κυρίως χώρες της Άπω Ανατολής.

#### **Στην Ελλάδα διοργανώθηκαν δύο events:**

Το ένα πραγματοποιήθηκε στις 9 Μαΐου στη Σύρο (**Scratch Day in Syros**) στο πλαίσιο του 5ου Συνεδρίου των Εκπαιδευτικών για τις ΤΠΕ-E (<http://www.e-diktyo.eu/>) που διοργάνωσαν οι Ενώσεις Εκπαιδευτικών e-diktyo και «Μιχάλης Δερτούζος».

Το δεύτερο πραγματοποιήθηκε στις 16 Μαΐου στην Πάτρα (**Scratch Day in Patras**, ) στο πλαίσιο της διημερίδας του παραρτήματος Πάτρας της OMEP (<http://www.omep.gr/>) από το Τμήμα Επιστημών της Εκπαίδευσης και της Αγωγής στην Προσχολική Ηλικία του Πανεπιστημίου Πατρών και το Εργαστήριο Μαθησιακής Τεχνολογίας & Διδακτικής Μηχανικής του Τμήματος των Επιστημών της Προσχολικής Αγωγής και του Εκπαιδευτικού Σχεδιασμού –Πανεπιστήμιο Αιγαίου.



### 4.3 Χαρακτηριστικά του Scratch

Όλες οι απαραίτητες προγραμματιστικές δομές αναπαρίστανται στο αριστερό μέρος της οθόνης ως τούβλα, τα οποία τοποθετούμενα σε στοίβες δημιουργούν προγράμματα. Τα τούβλα αυτά είναι σχεδιασμένα έτσι ώστε να ταιριάζουν μεταξύ τους μόνο όταν ο συνδυασμός τους έχει κάποιο συντακτικό νόημα. Κάθε τύπος δεδομένων έχει διαφορετικό σχήμα, ελαχιστοποιώντας έτσι την πιθανότητα λάθος επιλογής. Το πρόγραμμα παραμένει ζωντανό καθ' όλη τη διάρκεια της εκτέλεσης, έτσι ο χρήστης μπορεί να επιφέρει αλλαγές σε αυτό και να βλέπει άμεσα τα αποτελέσματα.

Οι εφαρμογές στο Scratch οικοδομούνται από αντικείμενα που λέγονται sprites. Τα sprites έχουν εμφάνιση-κουστούμι η οποία τοποθετείται στην «σκηνή-stage» της εφαρμογής. Η εμφάνιση ενός sprite αλλάζει αν ορίσουμε διαφορετικό

κουστούμι. Το sprite μπορεί να μοιάζει με άνθρωπο, τραίνο, πεταλούδα ή οτιδήποτε άλλο. Μπορείς να χρησιμοποιήσεις οποιαδήποτε ψηφιογραφική εικόνα σαν κουστούμι: μπορείς να δημιουργήσεις μια εικόνα από την ζωγραφική, να εισάγεις μια εικόνα από το σκληρό σου δίσκο ή από το Διαδίκτυο.

Μπορείς να δώσεις οδηγίες σε ένα αντικείμενο ώστε να κινηθεί, να παίξει μουσική ή να αλληλεπιδράσει με άλλα sprite. Για να πεις στο αντικείμενο τι να κάνει, στοιβάζεις τουβλάκια μεταξύ τους, για να σχηματίσεις σενάρια ενεργειών. Τα σενάρια καθορίζουν την συμπεριφορά των αντικειμένων. Όταν κάνεις διπλό κλικ στις ενέργειες, το scratch εκτελεί τις ενέργειες από την αρχή μέχρι το τέλος των ενεργειών. Αυτό επιτρέπει την άμεση δοκιμή των προγραμμάτων χωρίς την παρεμβολή σταδίων μετάφρασης πηγαίου κώδικα, σύνδεσης κλπ.

#### 4.4 Αρχές εύρους και χρήσης της γλώσσας

Κατά το σχεδιασμό της γλώσσας, η κύρια προτεραιότητα ήταν να γίνει η γλώσσα και το περιβάλλον ευκολονόητα έτσι ώστε η εκμάθηση τους από παιδιά που δεν έχουν προηγούμενη προγραμματιστική εμπειρία να γίνεται ακόμα πιο εύκολη. Υπάρχει μια ισχυρή αντίθεση μεταξύ των ισχυρών πολυμεσικών λειτουργιών και του πολυνηματικού προγραμματιστικού στυλ και του σχετικά περιορισμένου εύρους της γλώσσας προγραμματισμού (Φεσάκης, Δημητρακοπούλου, Σεραφείμ, Ζαφειροπούλου, Ντούνη & Τούκα, 2008).

Εμπειρικές μελέτες έχουν γίνει για διάφορες λειτουργίες — αυτές που παρεμπόδιζαν την φυσική εκμάθηση απορρίφθηκαν, ενώ αυτές που ενθάρρυναν τους αρχάριους και έκαναν εύκολη γι' αυτούς τη διερεύνηση και την εκμάθηση διατηρήθηκαν (Resnick, 2007). Ορισμένα από τα αποτελέσματα προκαλούν έκπληξη, κάνοντας το Scratch αρκετά διαφορετικό από άλλες διδασκόμενες γλώσσες (όπως η BASIC, Logo, ή Alice).

Για παράδειγμα, ο πολυνηματικός κώδικας μέσω περάσματος μηνυμάτων είναι θεμελιώδης για το Scratch, αλλά δεν έχει διαδικασίες ή [Είσοδο/Εξοδο](#) αρχείων (I/O) και υποστηρίζει μόνο μονοδιάστατους [πίνακες](#), γνωστούς ως Λίστες. Αριθμοί κινητής υποδιαστολής και [αλφαριθμητικά](#) υποστηρίζονται από την έκδοση 1.3, αλλά με περιορισμένη ικανότητα διαχείρισης αλφαριθμητικών.

Η εστίαση ήταν πάντα στην παιγνιώδη μάθηση, ώστε τα παιδιά του δημοτικού σχολείου να μπορούν να κάνουν απλά έργα και οι έφηβοι να μπορούν γρήγορα να έχουν εκπληκτικά αποτελέσματα. Επίσης υπάρχει ένας αριθμός έμπειρων ενήλικων προγραμματιστών στην κοινότητα του Scratch, που γενικά αναζητούν τρόπους να διδάξουν προγραμματισμό στα παιδιά τους (Pepler & Kafai, 2007).

## **4.5 Η Παιδαγωγική χρήση του Scratch**

Το πρόγραμμα Scratch έχει σχεδιαστεί έτσι ώστε να ευνοεί τους αρχάριους προγραμματιστές. Συγκεκριμένα, στοχεύει στην ανάπτυξη βασικών ικανοτήτων, όπως είναι: η δημιουργική σκέψη, η σαφής επικοινωνία, η συστηματική ανάλυση, η αποδοτική συνεργασία, ο επαναληπτικό-προοδευτικός σχεδιασμός, και οι δεξιότητες της δια βίου μάθησης (Monroy-Hernández & Resnick, 2008).

Το περιβάλλον του Scratch λόγω του ότι φέρει καινοτομίες στην προσέγγιση δυσνόητων εννοιών και τεχνικών προγραμματισμού, η διάδοση του στην εκπαίδευση και την εξωσχολική ενασχόληση των παιδιών βελτιώνει την σχέση των παιδιών με την επιστήμη των υπολογιστών γενικά, ενώ ταυτόχρονα καταστεί τον προγραμματισμό αντικείμενο περισσότερο ενδιαφέρον για ομάδες όπως τα κορίτσια και τους μαθητές των θεωρητικών επιστημών. Ο σχεδιασμός του με τα δομικά στοιχεία επιτρέπει τον εύκολο προγραμματισμό με εξάλειψη των λαθών στην σύνταξη, επιτρέποντας ανάδραση από τον χώρο που είναι στοιβαγμένα τα δομικά στοιχεία και δίνοντας άμεση ανάδραση για πειραματισμό. Με τον προγραμματισμό έρχονται αντιμέτωποι με τις ανεπτυγμένες προγραμματιστικές αρχές των αλγορίθμων, της συμπύκνωσης, του σχεδιασμού και της αφαίρεσης (Φεσάκης, Καφούση & Σκουμπουρδή, 2008).

Όσον αφορά τους εκπαιδευτικούς το περιβάλλον δίνει νέες δυνατότητες στον καθηγητή Πληροφορικής (Πρωτοβάθμιας ή Δευτεροβάθμιας εκπαίδευσης) να εμπλέξει τα παιδιά με εισαγωγή στον προγραμματισμό αλλά και προχωρημένες έννοιες (αντικείμενο, γεγονός, κ.α.). Εκτός της Διδακτικής του Προγραμματισμού το scratch παρέχει σε κάθε εκπαιδευτικό ένα εύκολο τρόπο παραγωγής διαδραστικού εκπαιδευτικού υλικού, μικρόκοσμων, προσομοιώσεων κλπ. (Φεσάκης, Δημητρακοπούλου, Σεραφείμ, Ζαφειροπούλου, Ντούνη & Τούκα, 2008).

Τέλος, όσον αφορά τους ερευνητές το scratch περιστοιχίζεται από ομάδα επιστημόνων που εργάζονται πάνω σε ενδιαφέροντα ζητήματα της ουσιαστικής ένταξης και αξιοποίησης των προγραμματιστικών περιβαλλόντων στην εκπαίδευση. Οι νέοι ερευνητές θα μπορέσουν να προσεγγίσουν καλύτερα το σώμα των θεωρητικών ιδεών και ερευνητικών δεδομένων που παράγονται από την κοινότητα του scratch. Επιπλέον η εξοικείωση με το περιβάλλον τους παρέχει την δυνατότητα να σχεδιάζουν και να δοκιμάζουν ερευνητικά ιδέες, αναπτύσσοντας εύκολα τα πρότυπα των υπολογιστικών περιβαλλόντων που απαιτούνται (Φεσάκης & Δημητρακοπούλου, 2007).

#### **4.6 Σχέση με άλλα συστήματα**

Το Scratch είναι γραμμένο στο Squeak ([www.squeak.org](http://www.squeak.org)), μία ανοιχτή εφαρμογή της γλώσσας Smalltalk-80. Εμπνέεται από άλλα περιβάλλοντα προγραμματισμού, που έχουν σχεδιαστεί για νέους και αρχάριους προγραμματιστές. Η προσέγγιση του εποικοδομισμού επισύρει την προσοχή σε προηγούμενη έρευνα για το πρόγραμμα LogoBlocks (Begel, 1996) και το Etoys (Steinmetz, 2001), τα οποία έχουν αποδειχθεί πολύ καλά για τους αρχάριους προγραμματιστές. Συγκεκριμένα, οι εντολές στο περιβάλλον LogoBlocks αναπαρίστανται με τον ίδιο τρόπο που αναπαρίστανται και στο Scratch, δηλαδή με γραφικά σχήματα που μπορούν να συρθούν (με χρήση drag & drop) από την παλέτα των εργαλείων στην περιοχή δημιουργίας, δίπλα σε άλλα σχήματα, και να σχηματίσουν έτσι το πρόγραμμα.

Η αλληλεπίδραση με τον χρήστη και το σύστημα πλοήγησης είναι εμπνευσμένο από το Logo Microworlds. Τα γραφικά του θυμίζουν τη LOGO, την εξερευνητική γλώσσα προγραμματισμού, η οποία δημιουργήθηκε τέσσερις δεκαετίες πριν στο Cambridge, με την διαφορά όμως τώρα ότι αυτή η καινούρια γλώσσα είναι πολύ πιο δυναμική.

## 4.7 Περιβάλλον ανάπτυξης, ιστότοπος και Scratch Player

Το περιβάλλον ανάπτυξης του Scratch μπορεί να μεταφορτωθεί δωρεάν και να εγκατασταθεί σε οποιοδήποτε υπολογιστή με Windows ή Mac OS X. Διάφορες εκδόσεις του Linux μπορούν να τρέξουν το Scratch, συγκεκριμένα για τα Suse Linux (openSUSE 11 και άνω) και Ubuntu, οι εγκαταστάτες (RPM για το Suse και DEB για Ubuntu) είναι διαθέσιμοι από τα επίσημα κοινοτικά καταθετήρια αυτών των εκδόσεων Linux. Ένας πειραματικός εγκαταστάτης για το Ubuntu είναι διαθέσιμος από το MIT.

Τα προγράμματα Scratch μπορούν να φορτωθούν αυτόματα από το περιβάλλον ανάπτυξης σε προσωπικές σελίδες στον ιστότοπο του Scratch, όπου άλλα μέλη της κοινότητας του Scratch μπορούν να τα μεταφορτώσουν (συμπεριλαμβανομένου του πλήρους πηγαίου κώδικα) για μάθηση ή ανάμιξη σε νέα έργα. Ο ιστότοπος, που είναι υλοποιημένος πάνω στην πλατφόρμα ScratchR, παρέχει επίσης δυνατότητα στα μέλη της κοινότητας να σχολιάσουν έργα, πέραν της παροχής γενικών χώρων συζητήσεων και χώρων επίδειξης έργων. Προγράμματα που έχουν αναπτυχθεί σε Scratch μπορούν να εκτελούνται είτε στο περιβάλλον ανάπτυξης ή μέσω μιας μικροεφαρμογής Java γνωστής ως Scratch Player. Ο Scratch Player επιτρέπει σε προγράμματα Scratch να εκτελεστούν από σχεδόν οποιαδήποτε εφαρμογή εμφάνισης ιστοσελίδων.

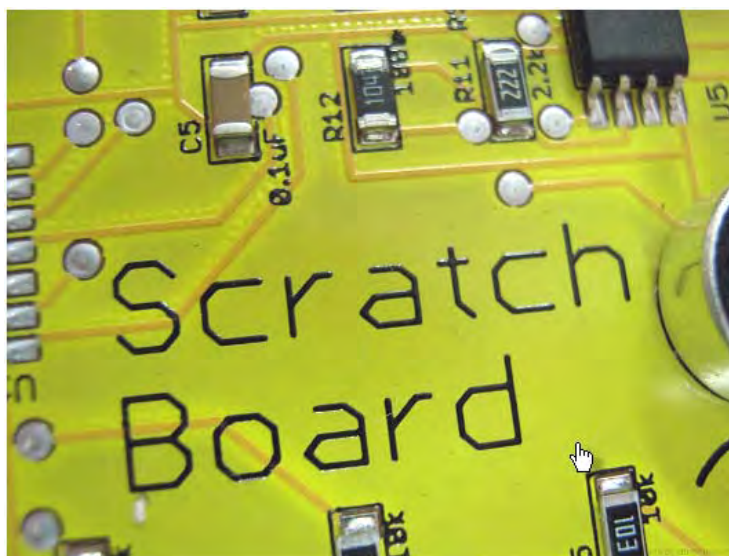
## 4.8 Το Scratchboard

Το Scratchboard είναι μια πλακέτα με αισθητήρες (Εικόνες 2,3), που κατασκευάστηκε από την ομάδα του Lifelong Kindergarten group του MIT Media Lab (<http://llk.media.mit.edu>) για να συνδέσει το προγραμματιστικό περιβάλλον και τις εφαρμογές του Scratch με τιμές φυσικών μεγεθών του πραγματικού κόσμου, με real time μετρήσεις τους. Ανοίγει δε πολλούς ορίζοντες στον προγραμματιστή του Scratch, προκαλώντας ιδέες για διάφορες πρωτότυπες εφαρμογές. Με το Scratchboard, οι άνθρωποι μπορούν να χρησιμοποιούν τους αισθητήρες για να ελέγχουν διαδραστικά τις ιστορίες και τα παιχνίδια που δημιουργούν με τη γλώσσα προγραμματισμού Scratch. Με τη σύνδεση του φυσικού / πραγματικού και του εικονικού περιβάλλοντος, το Scratchboard επεκτείνει τα όρια σε αυτά που μπορούν

να σχεδιαστούν με το Scratch αλλά και σε αυτά που μπορούν να μαθευτούν μέσα από τις σχετικές διαδικασίες.

Διαθέτει ενσωματωμένους:

- έναν αισθητήρα φωτός
- έναν αισθητήρα ήχου
- ένα πλήκτρο
- έναν μεταβολέα
- τέσσερις ακροδέκτες στους οποίους μπορούν να συνδεθούν 4 ζεύγη καλωδίων με κροκοδειλάκια (συμπεριλαμβάνονται στη συσκευασία), μέσω των οποίων μετριέται ηλεκτρική αντίσταση.

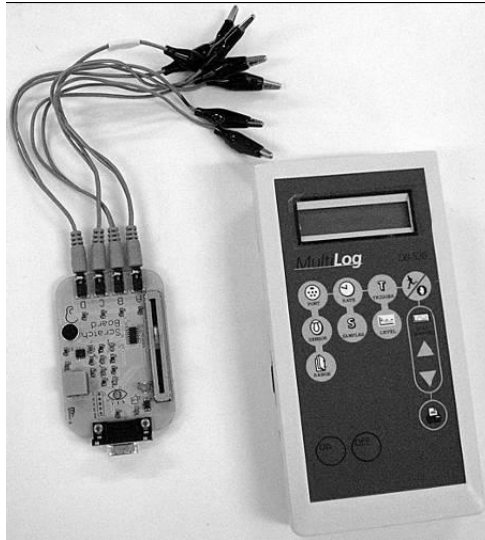


**Εικόνα 2:** Το Scratchboard

Επικοινωνεί με τον υπολογιστή μέσω σειριακής θύρας (ή θύρας USB) - συνοδεύεται δε στη συσκευασία του από καλώδιο serial-to-USB.

Κατασκευάζονταν μέχρι το 2008 από φοιτητές του MIT και διατίθενταν άμεσα από το MIT Media Lab στην τιμή των 30\$. Πλέον κατασκευάζεται και διατίθεται ακριβότερα, ως picoboard από την picocricket ([www.picocricket.com](http://www.picocricket.com)). Δεν απευθύνεται άμεσα σε εφαρμογές των Φυσικών Επιστημών - υπάρχουν ολοκληρωμένα ΣΣΛΑ για το σκοπό αυτό. Η ύπαρξη όμως των αισθητήρων, η πολύ χαμηλή τιμή και -το σημαντικότερο- η επικοινωνία του με το πολλών εκπαιδευτικών δυνατοτήτων Scratch, κινεί την περιέργεια για περαιτέρω διερεύνηση.





**Εικόνα 3:** Το Scratchboard δίπλα στο multilog

Το Scratchboard επικοινωνεί με τον υπολογιστή μέσω του πρωτοκόλλου RS-232, με baud rate 38.4k. Η τροφοδοσία του σε ηλεκτρική ενέργεια πραγματοποιείται μέσω της σειριακής θύρας. Το Scratch «διαβάζει» τις μετρήσεις του Scratchboard σε μια κλίμακα από 0 έως 100. Υπάρχει όμως και η δυνατότητα πρόσβασης και στις «ακατέργαστες» 10-bit τιμές, επιλέγοντας “report raw data” στο Scratch Board Watcher. Με τον τρόπο αυτό, οι τιμές από τους αισθητήρες κυμαίνονται από 0 έως 1023. Οι τιμές των μετρήσεων που καταγράφονται μπορούν να δίνονται άμεσα σε μεταβλητές του προγράμματος, αλλά και να αποθηκεύονται σε αρχείο, για περαιτέρω επεξεργασία και αξιοποίηση. Περισσότερες σχετικές πληροφορίες διατίθενται στο [http://scratch.mit.edu/files/scratchboard/ScratchBoard\\_Tech\\_InfoR1.pdf](http://scratch.mit.edu/files/scratchboard/ScratchBoard_Tech_InfoR1.pdf).

## **4.9 Οφέλη από την αξιοποίηση του Scratch**

### **4.9.1 Δεξιότητες διαχείρισης πληροφοριών και πολυμέσων**

- ✚ Δουλεύοντας με το Scratch, οι αρχάριοι μαθαίνουν να επιλέγουν, να δημιουργούν και να διαχειρίζονται πολλαπλές μορφές δεδομένων, συμπεριλαμβανομένων κειμένων, εικόνων, animations και ηχητικών ηχογραφήσεων (Adams, 2007).

- ✚ Καθώς οι μαθητές αποκτούν εμπειρία δημιουργώντας, γίνονται διορατικοί και αποκτούν κριτική σκέψη (Φεσάκης, Δημητρακοπούλου, Σεραφείμ, Ζαφειροπούλου, Ντούνη & Τούκα, 2008).

#### **4.9.2 Κριτική Σκέψη**

- ✚ Καθώς μαθαίνουν να προγραμματίζουν με το Scratch, οι αρχάριοι έρχονται σε επαφή με την κριτική σκέψη και τα ευφυή συστήματα (Φεσάκης & Δημητρακοπούλου, 2007).
- ✚ Για να δημιουργηθούν ολοκληρωμένες εργασίες, οι μαθητές χρειάζεται να συσχετίσουν το χρόνο με τις ενέργειες πολλαπλών sprites. Έτσι εξοικειώνονται με την αίσθηση του περιβάλλοντος, την ανάδραση και άλλες βασικές έννοιες (Kelleher & Pausch, 2007).

#### **4.9.3 Αναγνώριση, διαμόρφωση και επίλυση προβλημάτων**

- ✚ Το Scratch υποστηρίζει τη διερεύνηση και επίλυση προβλημάτων (Δαπόντες, 2008). Η δημιουργία μιας εργασίας απαιτεί (Φεσάκης, Δημητρακοπούλου, Σεραφείμ, Ζαφειροπούλου, Ντούνη & Τούκα, 2008).
  - τη σύλληψη μιας ιδέας,
  - το διαχωρισμό του προβλήματος σε βήματα και
  - την υλοποίηση τους χρησιμοποιώντας τα προγραμματιστικά εργαλεία του Scratch.
- ✚ Το Scratch έχει σχεδιαστεί ώστε να παραμένει διαρκώς ενεργό: οι μαθητές μπορούν να αλλάξουν δυναμικά κομμάτια του κώδικα και να δουν άμεσα αποτελέσματα (Kafai et al, 2007).

#### **4.9.4 Δημιουργικότητα**

- ✚ Το Scratch ενθαρρύνει την κριτική σκέψη, μια εξαιρετικά σημαντική δεξιότητα στο σημερινό διαρκώς μεταβαλλόμενο κόσμο, ενώ οδηγεί τους χρήστες του στην αναζήτηση καινοτόμων λύσεων σε απρόβλεπτα προβλήματα (Δαπόντες, 2008).

- ✚ Οι αρχάριοι μαθαίνουν όχι μόνο να λύνουν ένα πλήρως καθορισμένο πρόβλημα αλλά και να αντιμετωπίζουν και όποιο άλλο ασθενώς δομημένο ανακύψει (Δαπόντες, 2008).

#### 4.9.5 Διαπροσωπικές και συνεργατικές δεξιότητες

- ✚ Επειδή τα προγράμματα του Scratch αποτελούνται από γραφικά «τουβλάκια», ο κώδικάς του είναι πιο αναγνώσιμος και διαμοιράσιμος σε σχέση με άλλες γλώσσες (Resnick, 2007 ).
- ✚ Τα οπτικά αντικείμενα και ο πρότυπος κώδικας υποστηρίζουν συνεργασία, επιτρέποντας στους μαθητές να δουλεύουν μαζί σε εργασίες και να ανταλλάσσουν κομμάτια του κώδικα (Pepler & Kafai, 2007).
- ✚ μαθαίνουν να εργάζονται συνεργατικά για τη δημιουργία έργων, που δεν αποτελούν “μαύρα κουτιά”, αλλά είναι ανοιχτά για σχολιασμό, επαναχρησιμοποίηση και βελτιώσεις από οποιοδήποτε μέλος της κοινότητας (Δαπόντες, 2008)

#### 4.9.6 Υπευθυνότητα

- ✚ Όταν οι μαθητές δημιουργούν εργασίες στο Scratch, έχουν στο μυαλό τους το κοινό στο οποίο θα απευθυνθούν και πρέπει να σκεφτούν πως θα αντιδράσει σε αυτές (Φεσάκης, Καφούση & Σκουμπουρδή, 2008).
- ✚ Είναι εύκολο να δημιουργείς εφαρμογές χρήσιμες και ευχάριστες (Φεσάκης, Καφούση & Σκουμπουρδή, 2008).

## 5 Η αξιολόγηση ως μέρος της διαδικασίας διδασκαλίας-μάθησης

---

Οι μαθητές συμμετέχουν ενεργά στην εκπαιδευτική διαδικασία με απώτερο σκοπό τη μάθηση γνωστικών αντικειμένων, συμπεριφορών, στάσεων και ψυχοκινητικών δεξιοτήτων. Βασιζόμενοι σε αυτή την παραδοχή, σκοπός των εκπαιδευτικών, ανεξάρτητα της εκπαιδευτικής βαθμίδας στην οποία ανήκουν, είναι να καθοδηγήσουν αποτελεσματικά τους μαθητές τους στην κατάκτηση και τον εμπλουτισμό της μάθησης (γνωστικής, συναισθηματικής και ψυχοκινητικής).

Ένα καθοριστικό μέσο για την απόκτηση και επεξεργασία σημαντικών πληροφοριών αναφορικά με τη συνεχή ή μη μαθησιακή πρόοδο των μαθητών είναι και η διαδικασία της αξιολόγησης. Η διαδικασία αυτή δεν θα πρέπει να αποτελεί το τελευταίο στάδιο της διδακτικής πρακτικής (τελική αξιολόγηση-summative evaluation), αλλά θα πρέπει να αποτελεί αναπόσπαστο μέρος των διδακτικών προσεγγίσεων με σκοπό την τροποποίηση και την ενίσχυση (διαμορφωτική αξιολόγηση-formative evaluation). Η αξιολόγηση δεν αποτελεί από μόνη της καταληκτικό σημείο. Είναι το μέσο που θα μας οδηγήσει στην αναθεώρηση και αναπροσαρμογή των διδακτικών μας στόχων και τεχνικών (Airasian, 1997). Η διαμορφωτική αξιολόγηση αποβλέπει στη βελτίωση της καθοδήγησης προς τους μαθητές, ικανοποιεί τις ατομικές ανάγκες του κάθε παιδιού (εξατομικευμένη διδασκαλία), εμπλουτίζει τη μάθηση και ελέγχει το βαθμό επίτευξης των στόχων όπως αυτοί καθορίζονται από τα αναλυτικά προγράμματα σπουδών.

Καθώς η μαθησιακή διαδικασία είναι μια περίπλοκη διαδικασία και οι εκπαιδευτικοί στόχοι είναι ποικιλόμορφοι, ένας συνδυασμός από μεθόδους αξιολόγησης απαιτείται προκειμένου να συγκεντρωθούν αντικειμενικές, έγκυρες και αξιόπιστες πληροφορίες αναφορικά με την πρόοδο των μαθητών (Wiggins, 1989).

## 5.1 Οι τύποι αξιολόγησης του Bloom

Ο Bloom και οι συνεργάτες του (1971) αναφέρονται σε τρεις τύπους αξιολόγησης:

- α) Την *αρχική ή διαγνωστική αξιολόγηση*.
- β) Τη *διαμορφωτική – σταδιακή αξιολόγηση* (formative evaluation).
- γ) Την *τελική ή αθροιστική αξιολόγηση* (summative evaluation).

Στη διαγνωστική αξιολόγηση εντάσσονται οι δοκιμασίες εκείνες που αποσκοπούν στο να προσδιορίσουν το επίπεδο γνώσεων των εκπαιδευομένων σε έναν ορισμένο τομέα, προκειμένου να προσαρμοστεί κατάλληλα η εκπαιδευτική διαδικασία που έπεται. Η αρχική ή διαγνωστική αξιολόγηση κατέχει σπουδαία θέση στις σύγχρονες διδακτικές μεθόδους. Χρησιμοποιούνται ειδικές δοκιμασίες-τεστ, γνωστές με το όνομα δοκιμασίες επιπέδου (placement tests), που επιτρέπουν την κατάταξη των εκπαιδευομένων σε ορισμένο επίπεδο, ανάλογα με τις προηγούμενες γνώσεις και τις εμπειρίες του καθενός γύρω από το προς διδασκαλία θέμα. Η διδασκαλία προσαρμόζεται σε κάθε περίπτωση στις δυνατότητες και στο αρχικό επίπεδο του κάθε ατόμου ή μιας μικρής ομάδας ατόμων του αυτού επιπέδου και αποφεύγεται έτσι η ανισότητα στην αφετηρία για νέα μάθηση, που υπάρχει μεταξύ των διαφόρων κατηγοριών εκπαιδευομένων.

Η διαμορφωτική αξιολόγηση αποσκοπεί στον έλεγχο της πορείας του εκπαιδευομένου προς την κατάκτηση συγκεκριμένου εκπαιδευτικού στόχου. Επιδιώκεται να εξαχθούν οι απαραίτητες πληροφορίες που απαιτούνται για την τροποποίηση του προγράμματος ή των μεθόδων διδασκαλίας.

Η τελική αξιολόγηση είναι εκείνη που πραγματοποιείται κατά τη στιγμή της συνολικής εκτίμησης του τι επιτεύχθηκε σε σύγκριση βέβαια με το στόχο που είχε αρχικά καθοριστεί. Ο τελικός στόχος μπορεί να αναλύεται σε υπό-στόχους με βάση επιμέρους διδακτικές ενότητες, των οποίων ο έλεγχος αποτελεί αντικείμενο της διαμορφωτικής αξιολόγησης, ενώ η εκτίμηση της επίτευξης του τελικού στόχου είναι αντικείμενο της αθροιστικής ή τελικής αξιολόγησης. Η τελική αξιολόγηση δεν είναι απαραίτητο να πραγματοποιείται μόνο στο τέλος μιας μακράς περιόδου ή ύστερα από την ολοκλήρωση μιας ορισμένης εκτάσεως ύλης ούτε αφορά υποχρεωτικά στο σύνολο των υποστόχων στα πλαίσια ενός μαθήματος. Η τελική αξιολόγηση μπορεί να

γίνεται και για υποσύνολα στόχων που απαρτίζουν μια ευρύτερη ενότητα του γενικού συνόλου.

Σε αυτό το σημείο αξίζει να αναφέρουμε ότι κατά τη διάρκεια της διαγνωστικής αξιολόγησης, όπου επιχειρείται η ανίχνευση του βαθμού κατανόησης πρότερων γνωστικών αντικειμένων των μαθητών, οι τεχνικές (υποβολή ερωτήσεων, τεστ πολλαπλών επιλογών) κυρίως αποσκοπούν στον έλεγχο της ικανότητας ανάκλησης πληροφοριών. Ο περιορισμένος χρόνος της διδασκαλίας και οι συνήθεις μέθοδοι αξιολόγησης δεν δίνουν την ευκαιρία στο διδάσκοντα να διαπιστώσει την ικανότητα εφαρμογής των αποκτηθέντων γνώσεων των μαθητών σε καθημερινές καταστάσεις. Για παράδειγμα, η ικανότητα ενός μαθητή να αναπαράγει δομημένο, περιεκτικό και κριτικό γραπτό λόγο είναι αδύνατο να διαπιστωθεί μέσα από μια διαδικασία αξιολόγησης στην οποία κυρίαρχα μέσα αποτελούν τα τεστ ανάκλησης πληροφοριών (πολλαπλών επιλογών) (Mayer, 2002).

Για αυτό το συγκεκριμένο λόγο αναπτύχθηκαν και υποστηρίχθηκαν από μια μεγάλη μερίδα της εκπαιδευτικής κοινότητας οι μέθοδοι αξιολόγησης που εκτιμούν την ικανότητα για πρακτική εφαρμογή των αποκτηθέντων γνώσεων και εμπειριών (performance-based assessments).

Αν προσπαθούσαμε να προσδιορίσουμε τους γενικούς στόχους που επιχειρεί να επιτύχει η συγκεκριμένη τεχνική αξιολόγησης, θα μπορούσαμε να αναφέρουμε τρεις (Stiggins & Chappuis, 2005):

- ✚ Συγκέντρωση πληροφοριών για τον εξατομικευμένο τρόπο κατανόησης του / της μαθητή / τριας.
- ✚ Συγκέντρωση πληροφοριών για το βαθμό εφαρμογής των αποκτηθέντων γνώσεων από τους μαθητές.
- ✚ Δυνατότητα ελέγχου του βαθμού αποτελεσματικότητας των διδακτικών τεχνικών μέσα από μια διαδικασία ενσωμάτωσης της αξιολόγησης στη διδακτική μεθοδολογία.

Ο Hibbard (1996) αναφέρει ότι το είδος αυτό της αξιολόγησης αντιπροσωπεύει ένα σύνολο στρατηγικών για τον έλεγχο της αποτελεσματικής εφαρμογής της γνώσης, των δεξιοτήτων και των καθημερινών μαθησιακών τεχνικών των μαθητών μέσα από τη διεξαγωγή δραστηριοτήτων που είναι κατάλληλα σχεδιασμένες για να ενισχύουν την ενεργό συμμετοχή τους.

## 5.2 Performance-Based Assessment (Αξιολόγηση επίδοσης)

Η αξιολόγηση επίδοσης δεν προσδιορίζεται από μία συγκεκριμένη τεχνική αξιολόγησης αλλά από μια σειρά προσεγγίσεων οι οποίες έχουν ως κύριο σκοπό να αξιολογήσουν αντικειμενικά την επίδοση των μαθητών και παράλληλα να βελτιώσουν τη διδακτική διαδικασία. Αυτές οι διαφορετικές, αλλά αλληλοσυμπληρούμενες, προσεγγίσεις συνήθως αναφέρονται στη βιβλιογραφία ως (Πανταζής, 2007):

- **Εναλλακτική αξιολόγηση (alternative assessment):** Ο όρος αυτός χρησιμοποιείται εννοιολογικά για να υποδηλώσει τη λειτουργική διαφοροποίησή του από τις παραδοσιακές μορφές αξιολόγησης οι οποίες και περιελάμβαναν κυρίως τα γνωστά ως αντικειμενικά τεστ (objective tests). Τα τεστ αυτά ήταν κυρίως: α) δύο επιλογών (Σωστό / Λάθος), β) πολλαπλών επιλογών ή γ) περιορισμένων απαντήσεων (μιας ή δύο λέξεων). Δοκίμια, εκθέσεις, προφορικές ερωτήσεις και η αυτοαξιολόγηση των μαθητών περιλαμβάνονται σε αυτή τη κατηγορία.

- **Αυθεντική αξιολόγηση (authentic assessment):** Το είδος αυτό της αξιολόγησης σχεδιάστηκε για να αξιολογήσει την επίδοση των μαθητών μέσα από διαδικασίες επίλυσης προβλημάτων και ανάθεσης δραστηριοτήτων. Σκοπός τους είναι να δραστηριοποιήσουν τους μαθητές και να τους φέρουν σε επαφή με καταστάσεις που θα αντιμετώπιζαν στην καθημερινή τους ζωή. Συστηματικές παρατηρήσεις για το πώς εργάζονται και συνεργάζονται οι μαθητές και οι φάκελοι των μαθητών (portfolios) περιλαμβάνονται σε αυτό το είδος της αξιολόγησης.

- **Αξιολόγηση επίδοσης (performance-based-assessment):** Είναι το είδος της αξιολόγησης που ελέγχει την ικανότητα των μαθητών να χρησιμοποιούν διάφορα μέσα (υπολογιστές, μικροσκόπια, όργανα φυσικής, φωτογραφικές μηχανές), να δημιουργούν, να παράγουν ή να εκτελούν μια δραστηριότητα η οποία και διεξάγεται κάτω από συγκεκριμένες συνθήκες και με προκαθορισμένα κριτήρια. Έργα ζωγραφικής, χειροτεχνίας, κατασκευές, παρουσιάσεις, χρήση μουσικών οργάνων, έρευνες, projects, αθλητικές επιδείξεις και εκθέσεις περιλαμβάνονται στις αξιολογικές δραστηριότητες οι οποίες και μπορούν να ενταχθούν σε αυτή τη κατηγορία (Brualdi, 1999).

Ανεξάρτητα από τον όρο τον οποίο χρησιμοποιούμε, το είδος αυτό της αξιολόγησης αποσκοπεί στο διαρκή έλεγχο της κατάκτησης αντικειμενικών και εφαρμόσιμων μαθησιακών προϊόντων και όχι στην αναγνώριση του βαθμού ανάκλησης γνώσεων που επιτυγχάνεται μέσα από μια διαδικασία επιλογής των σωστών απαντήσεων (Kane and Mitchell, 1996). Έμφαση δηλαδή δίνεται στην καλλιέργεια κριτικής και δημιουργικής σκέψης και συνειδητά αποφεύγεται η στείρα συσσώρευση γνώσεων που πιθανόν να αποδεσμεύονται από τη καθημερινή μας ζωή.

Στον προσανατολισμό προς την ανάπτυξη της τεχνικής αυτής συντέλεσε και το εποικοδομητικό (constructivistic) γνωστικό μοντέλο το οποίο και βοήθησε στην αναθεώρηση της διδακτικής / μαθησιακής διαδικασίας. Το μοντέλο αυτό, αναγνωρίζοντας τον ιδιαίτερο τρόπο με τον οποίο ο κάθε μαθητής δομεί τη γνώση, κατευθύνθηκε περισσότερο προς την εξατομικευμένη διδασκαλία. Μέσα από κατάλληλα διαμορφωμένες δραστηριότητες ο μαθητής μαθαίνει στην ουσία πως να μαθαίνει. Αντίστοιχα και οι τεχνικές αξιολόγησης βασισμένες σε αυτό το μοντέλο θα πρέπει να εξελιχθούν με τέτοιο τρόπο ώστε να συγκεντρώνουν αξιόπιστες πληροφορίες για:

- ✚ τη διαδικασία μάθησης των μαθητών,
- ✚ τις πρότερες γνώσεις τους,
- ✚ τη δυνατότητα εφαρμογής των αποκτηθέντων γνώσεων στη καθημερινή ζωή.

Συμπληρωματικά, οι υποστηρικτές της αξιολόγησης επίδοσης επισημαίνουν ότι η χρησιμοποίηση αυτών των τεχνικών ενισχύουν την παρόρμηση των μαθητών και επίσης εμπλουτίζουν τη μαθησιακή διαδικασία. Ιδιαίτερα, η διαδικασία αξιολόγησης κοινών δραστηριοτήτων τις οποίες μπορούμε να εφαρμόσουμε σε περισσότερες από μία διδακτικές ενότητες και σχετίζονται με καθημερινές καταστάσεις οδηγεί στην ενεργό συμμετοχή των παιδιών. Καθοριστικό ρόλο παίζει η σύνδεση των δραστηριοτήτων με τα προσωπικά βιώματα και τα ενδιαφέροντα των μαθητών για να μεγιστοποιηθεί η δημιουργική τους ενασχόληση με αυτές (Russell, Elton, Swinglehurst, & Greenhalgh, 2006).

Στις παραπάνω επισημάνσεις θα πρέπει να προσθέσουμε και τις σύγχρονες απαιτήσεις του σύγχρονου επαγγελματικού χώρου για την απόκτηση λειτουργικών προσόντων που εκτείνονται πέρα από την απλή συσσώρευση γνώσεων στην αποτελεσματική εφαρμογή των εξειδικευμένων γνώσεων και των αντίστοιχων



δεξιοτήτων. Η παραδοχή αυτή αναμφίβολα βρίσκει εφαρμογή και ευνοεί την ενσωμάτωση τεχνικών αξιολόγησης επίδοσης.

Η διαδικασία αξιολόγησης θα πρέπει και μπορεί να εξελιχθεί σε μια διαδικασία συνεχούς επικοινωνίας ανάμεσα στους μαθητές και στον εκπαιδευτικό. Η πρακτική αυτή θα αποτελεί το υπόβαθρο και θα υποστηρίζει συνεχώς τη διδασκαλία έχοντας ως γενικό της σκοπό τη διαρκή βελτίωση της μάθησης. Η φράση η οποία χαρακτηρίζει την ανάγκη ενσωμάτωσης της αξιολόγησης στη διδακτική διαδικασία είναι: «*Αξιολογούμε για να βελτιώνουμε τη διδακτική / μαθησιακή διαδικασία και δεν διδάσκουμε απλά για να αξιολογούμε*» (Ματσαγγούρας, 2001).

### **5.3 Rubrics (Ρούμπρικες) ως εργαλεία αξιολόγησης**

Οι ρούμπρικες είναι κλίμακες βαθμολόγησης οι οποίες σχεδιάζονται και αναπροσαρμόζονται σύμφωνα με τους διδακτικούς στόχους και επιτρέπουν την εξαγωγή αντικειμενικών, αξιόπιστων και άμεσων συμπερασμάτων αναφορικά με την επίδοση και τις δεξιότητες των μαθητών (Arter, 2000). Οι κλίμακες αυτές έχουν μία καθορισμένη δομή με τα διάφορα επίπεδα επίδοσης να τοποθετούνται στην πρώτη σειρά και τα κριτήρια επίδοσης στην αριστερή στήλη. Στα υπόλοιπα μέρη (κελιά) εισάγουμε αναλυτικά τα δευτερεύοντα κριτήρια και ανάλογα με το βαθμό ικανοποίησής τους από τους μαθητές προσδίδουμε και ένα συγκεκριμένο βαθμό.

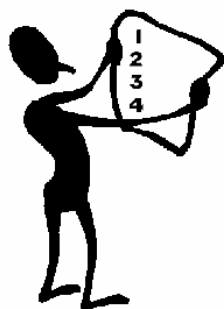
Οι ρούμπρικες αποτελούν εξαιρετικά χρήσιμα εργαλεία αξιολόγησης και ανατροφοδότησης τόσο για τους εκπαιδευτικούς όσο και για τους μαθητές και μπορούν να χρησιμοποιηθούν για να αξιολογήσουν μια ποικιλία προφορικών και γραπτών δραστηριοτήτων και εργασιών. Θα πρέπει να κατανοήσουμε ό,τι το είδος αυτό της αυθεντικής αξιολόγησης αποδεικνύεται αποτελεσματικό στο βαθμό που η υιοθέτηση του ή η κατασκευή του χαρακτηρίζεται αξιόπιστη, δηλαδή, να είναι σε θέση να μετράει τις μεταβλητές που αρχικά σκόπευε να μετρήσει.

Θα μπορούσαμε να πούμε ότι η κατασκευή τους αποσκοπεί στο να απαντήσει ικανοποιητικά σε συνήθεις ερωτήσεις που σχετίζονται με τα αποτελέσματα της διδακτικής διαδικασίας όπως (Wilson, 2004):

- ✚ Ποια είναι τα μετρήσιμα στοιχεία τα οποία αποτελούν και καθορίζουν την κατάκτηση της προσφερόμενης γνώσης από τους μαθητές;

- ✚ Ποια είναι τα διακριτά επίπεδα που προσδιορίζουν τη λειτουργική απόκτηση ή την αδυναμία απόκτησης προσφερόμενων προϊόντων μάθησης και αντίστοιχων δεξιοτήτων;
- ✚ Ποια είναι τα κριτήρια που καθορίζουν όλο το φάσμα της επίδοσης των μαθητών της τάξης με την ολοκλήρωση της διδακτικής διαδικασίας;
- ✚ Πώς επιτυγχάνεται η αξιοπιστία, αντικειμενικότητα και η αναγκαία εγκυρότητα στις τεχνικές αξιολόγησης επίδοσης;
- ✚ Πώς μεταφράζεται και αναλύεται η βαθμολογία (ποσοτική / ποιοτική) που έχουμε καταγράψει για να αξιολογήσουμε τη διδακτική / μαθησιακή διαδικασία;
- ✚ Πώς περιγράφουμε και διακρίνουμε τα διάφορα επίπεδα επίδοσης των μαθητών έτσι ώστε να διευκολύνουμε και να βελτιώσουμε τον έλεγχο της διδακτικής μας προσέγγισης;

Βασικό χαρακτηριστικό του είδους αυτού της αξιολόγησης είναι η συστηματική και λεπτομερής καταγραφή της επίδοσης του μαθητή (εργασίες, παρουσιάσεις, συμπεριφορά, προσαρμοστικότητα, επίλυση προβλημάτων, φάκελοι μαθητή, projects) μέσα από τη χρησιμοποίηση συγκεκριμένων, προεπιλεγμένων κριτηρίων. Τα κριτήρια αυτά είναι προσαρμοσμένα στη μαθησιακή διαδικασία και έχουν καθολική εφαρμογή στους μαθητές της τάξης της οποίας εφαρμόζεται η αξιολόγηση. Επίσης επιτρέπει και ενθαρρύνει τους μαθητές να γνωρίζουν τις διαβαθμίσεις της επίδοσής τους καθώς και πως μπορούν να τη βελτιώσουν (Mitchell, 2006).



Τα βασικά μέρη τα οποία αποτελούν μια ρούμπρικα είναι τέσσερα (Kan, 2007):

- ✚ Η κλίμακα η οποία περιλαμβάνει τους βαθμούς που αναλογούν για την αξιολόγηση μιας δραστηριότητας.
- ✚ Η περιγραφή των επιπέδων της επίδοσης που περιλαμβάνει παραδείγματα της αναμενόμενης επίδοσης και που διακριτά χαρακτηρίζει το κάθε επίπεδο.

- ✚ Τα κριτήρια που περιγράφουν τις παραμέτρους τις οποίες οι επιδόσεις θα πρέπει να ικανοποιούν για να θεωρούνται επιτυχείς.
- ✚ Αναμενόμενες γενικές επιδιώξεις οι οποίες και καθορίζουν συγκεκριμένα το βαθμό ικανοποίησης των κριτηρίων.

### 5.3.1 Τύποι rubrics (ρουμπρικών)

Οι κατηγορίες στις οποίες μπορούμε να κατατάξουμε τις ρούμπρικες είναι τέσσερις (Kap, 2007):

1. **Ολιστικές:** εμπεριέχουν ένα και μόνο βαθμό βασισμένο στη γενικότερη εντύπωση της επίδοσης του μαθητή σε μια μαθησιακή δραστηριότητα.
2. **Αναλυτικές:** παρέχουν λεπτομερή ανατροφοδότηση σε όλες τις παραμέτρους της επίδοσης που κρίνεται απαραίτητο να αξιολογηθούν.
3. **Γενικών στόχων:** οι οποίες μπορούν να εφαρμοστούν σε περισσότερες από μία διδακτικές και θεματικές ενότητες.
4. **Ειδικών στόχων:** οι οποίες συντάσσονται για να εκτιμήσουν το βαθμό απόκτησης συγκεκριμένων γνωστικών αντικειμένων και δεξιοτήτων από τους μαθητές (Arter, 2000).

Οι παράγοντες οι οποίοι επηρεάζουν την επιλογή ενός τύπου ρούμπρικας είναι ο χρόνος που έχουμε στη διάθεσή μας τόσο για να τη συντάξουμε όσο και για να την εφαρμόσουμε καθώς και το είδος και η δυσκολία των προσφερομένων γνωστικών αντικειμένων και δεξιοτήτων.

Ανεξάρτητα όμως από το είδος και την έκταση της ρούμπρικας υπάρχουν κάποιες γενικές αρχές τις οποίες και θα πρέπει να γνωρίζουμε και να εφαρμόζουμε (McAteer, 2005):

1. Όταν αναφερόμαστε περιγραφικά στα κριτήρια / ενδείξεις της επιτυχούς επιτέλεσης των δραστηριοτήτων θα πρέπει να είμαστε διεξοδικά λεπτομερείς και τα κριτήρια στα οποία αναφερόμαστε θα πρέπει να γίνονται απολύτως κατανοητά από τους μαθητές. Για παράδειγμα, όταν αναφερόμαστε στην αναζήτηση και καταγραφή των κύριων σημείων μέσα σε ένα κείμενο θα πρέπει να: α) αναφερθούμε στα χαρακτηριστικά των σημείων, β) τους λόγους για τους οποίους μερικά σημεία μπορούν να χαρακτηριστούν ως κύρια, γ) σε ποια μέρη του κειμένου συνήθως μπορούμε να τα αναζητήσουμε, δ) πόσα

σημεία μπορούμε να καταγράψουμε ώστε ο αριθμός τους να θεωρείται ικανοποιητικός, ε) πώς θα μπορούσαμε να τα κατατάξουμε (κριτήριο σπουδαιότητας, συχνότητα εμφάνισης, από την αρχή προς το τέλος).

**2.** Τα διαφορετικά επίπεδα της επίδοσης που θα αξιολογηθεί θα πρέπει να είναι διακριτά ώστε να διευκολύνεται η γρήγορη αναγνώριση και καταγραφή τους καθώς και να αποφεύγονται πιθανές παρερμηνείες. Ένας τρόπος με τον οποίο μπορούμε να διαφοροποιήσουμε τα επίπεδα αυτά, ανεξάρτητα από τον αριθμό τους (ας υποθέσουμε πέντε) είναι να προσδιορίσουμε τρεις δομικές κατηγορίες (κατώτερο, μεσαίο, ανώτερο) και στη συνέχεια να ενσωματώσουμε σε αυτές συμπληρωματικές διαβαθμίσεις. Οι διαφορές ανάμεσα στις διαβαθμίσεις θα πρέπει επίσης να είναι εύκολα κατανοητές.

**3.** Τα κριτήρια και οι ενδείξεις που καθορίζουν το κάθε επίπεδο αξιολόγησης επίδοσης θα πρέπει να είναι ισάριθμα, αυτούσια και λεπτομερή και να αποφεύγονται οι μη αναγκαίες συγκρίσεις. Στόχος αυτού του είδους της αξιολόγησης είναι να προσφέρει αξιόπιστα και αντικειμενικά στοιχεία για την απόδοση του μαθητή, ενώ παράλληλα να δίνει τη δυνατότητα στους μαθητές να διαπιστώσουν το επίπεδο προόδου στο οποίο βρίσκονται, τους λόγους για τους οποίους βρίσκονται στο συγκεκριμένο επίπεδο και τι απαιτείται για να προχωρήσουν στο αμέσως ανώτερο επίπεδο (με την προϋπόθεση ότι υπάρχουν ανώτερα επίπεδα).

**4.** Μπορούμε να παροτρύνουμε τους μαθητές να συμμετάσχουν ενεργά στη διαδικασία κατασκευής μιας ρούμπρικας προτείνοντας επίπεδα προόδου και συμπληρωματικά κριτήρια αξιολόγησης βασισμένα στην ατομική τους πρόοδο και σε ενδεχόμενες δυσκολίες που πιθανόν να αντιμετωπίσουν. Επίσης, μπορούν να εμπλουτίσουν υφιστάμενες ρούμπρικες ή να μεταλλάξουν ομαδικές ρούμπρικες σε ατομικές αναγνωρίζοντας τις ατομικές τους διαφορές και βελτιώνοντας τη διαδικασία αυτοαξιολόγησης.

### 5.3.2 Πλεονεκτήματα των rubrics (ρουμπρικών)

Πολλοί εμπειρογνώμονες θεωρούν ότι οι ρούμπρικες βελτιώνουν την επίδοση των μαθητών και επομένως συντελούν πολύ θετικά στην διαδικασία της μάθησης. Πιο συγκεκριμένα, όταν οι μαθητές λαμβάνουν τις ρούμπρικες εκ των προτέρων, καταλαβαίνουν πώς θα αξιολογηθούν και μπορούν να προετοιμαστούν αναλόγως.

Ενδεικτικά κάποια από τα πλεονεκτήματα της ένταξης και χρήσης των ρουμπρικών στην εκπαιδευτική κοινότητα παρουσιάζονται παρακάτω (Mitchell, 2006):

- ✚ Οι ρούμπρικες παρουσιάζουν σαφώς στους μαθητή πώς η εργασία τους θα αξιολογηθεί και τι αναμένεται από αυτούς, βελτιώνοντας έτσι με αυτό τον τρόπο την επίδοση τους.
- ✚ Οι ρούμπρικες βοηθούν τους μαθητές να κρίνουν την ποιότητας της εργασίας τους.
- ✚ Οι ρούμπρικες επιτρέπουν στην αξιολόγηση να είναι αντικειμενικότερη, συνεπής και ακριβής.
- ✚ Οι ρούμπρικες μειώνουν το χρονικό διάστημα που ο δάσκαλος σπαταλά προκειμένου να αξιολογήσει τις εργασίας των μαθητών.
- ✚ Οι ρούμπρικες βοηθούν τους μαθητές να συνειδητοποιήσουν τα κριτήρια που χρησιμοποιούν για την αξιολόγηση της συνεργατικής απόδοσης.
- ✚ Οι ρούμπρικες παρέχουν χρήσιμη ανατροφοδότηση (feedback) στο δάσκαλο σχετικά με την αποτελεσματικότητα της διδασκαλίας
- ✚ Οι ρούμπρικες παρέχουν στους σπουδαστές περισσότερη πληροφοριακή ανατροφοδότηση (feedback) σχετικά με τις γνώσεις τους, τις δεξιότητες τους καθώς και τις αδυναμίες τους.
- ✚ Οι ρούμπρικες προσαρμόζουν τις ετερογενείς κατηγορίες με την προσφορά μιας σειράς ποιοτικών επιπέδων.
- ✚ Οι ρούμπρικες είναι εύχρηστες και πολύ εύκολο να εξηγηθούν.

### 5.3.3 Απροθυμία & Αδυναμία των εκπαιδευτικών να εισάγουν τα rubrics (ρούμπρικες) στην διαδικασία μάθησης

Αν και η αξιολόγηση της εφαρμογής γνώσεων και δεξιοτήτων αναγνωρίστηκε ως ένα σημαντικό είδος αξιολόγησης, παρ' όλα αυτά δεν μπορούμε να πούμε ότι εφαρμόστηκε από ένα μεγάλο μέρος εκπαιδευτικών. Η διαπίστωση αυτή είναι αναμενόμενη μια και όπως συμβαίνει και με το σύνολο των καινοτόμων εφαρμογών προσεγγίστηκε από τους εκπαιδευτικούς με έναν δικαιολογημένο σκεπτικισμό.

Στην επισκόπηση της ξένης βιβλιογραφίας αναφέρονται προσπάθειες επεξήγησης της διστακτικότητας των εκπαιδευτικών να εφαρμόσουν την προαναφερόμενη μέθοδο αξιολόγησης (Wiggins, 1998). Αξίζει να σημειωθεί πως αν και οι σχετικές έρευνες διεξήχθησαν σε σχολεία των Η.Π.Α τα αποτελέσματα αναφορικά με τις στάσεις και τις συμπεριφορές των εκπαιδευτικών παρουσιάζουν στο μεγαλύτερο μέρος τους κοινά σημεία.

Δύο είναι οι κατηγορίες στις οποίες μπορούμε να εντάξουμε τη γενικότερη απροθυμία των εκπαιδευτικών να συμπεριλάβουν ενεργά τη συγκεκριμένη διαδικασία αξιολόγησης στις διδακτικές τους τεχνικές:

- ✚ Οι δάσκαλοι αισθάνονται στο μεγαλύτερο μέρος τους ότι δε γνωρίζουν όλες τις λεπτομέρειες της τεχνικής για να μπορέσουν εφαρμόσουν με επιτυχία την αξιολόγηση της επίδοσης των μαθητών τους (Airasian, 1991).
- ✚ Όταν προσπάθησαν να την εφαρμόσουν, είτε συνάντησαν δυσκολίες σε κάποια φάση της, είτε τα αποτελέσματα τα οποία συγκέντρωσαν ήταν ανολοκλήρωτα και κατά συνέπεια μη αξιοποιήσιμα (Stiggins, 1994).

Θα μπορούσαμε να συμπληρώσουμε ότι οι ανωτέρω δυσκολίες μπορούν να αντιμετωπιστούν με δύο κυρίως τρόπους (Stiggins & Chappuis, 2005):

- α) με τη διεξαγωγή επιμορφωτικών σεμιναρίων που θα έχουν ως κύριο θέμα τους την εφαρμογή αποτελεσματικής αξιολόγησης και
- β) με την ανταλλαγή απόψεων και προτάσεων των εκπαιδευτικών που επιθυμούν να την εφαρμόσουν.

Ο Stiggins (1994) μας πληροφορεί ό,τι για να εφαρμόσουμε αποτελεσματικά μια διαδικασία αξιολόγησης το πρώτο πράγμα που θα πρέπει να καθορίσουμε είναι ο ειδικός σκοπός για τον οποίο διενεργούμε την αξιολόγηση.

Μερικές από τις ερωτήσεις που πρέπει να θέσει ο εκπαιδευτικός στον εαυτό του πριν την αξιολόγηση είναι οι εξής (Lane, 2004):

- ✚ Ποιες γνώσεις, έννοιες, ή δεξιότητες θέλω να διαπιστώσω αν θα κατακτηθούν από τους μαθητές μου;
- ✚ Τι ακριβώς οι μαθητές μου πρέπει να γνωρίζουν πριν την έναρξη της διδακτικής / μαθησιακής διαδικασίας ;
- ✚ Ποιο είναι το γενικό γνωστικό επίπεδο το οποίο αναμένεται να κατακτηθεί / εμπλουτιστεί από τους μαθητές μου;
- ✚ Ποιο είναι το επίπεδο της γνώσης και των δεξιοτήτων τα οποία αξιολογούνται; Για παράδειγμα, είναι η κριτική ικανότητα, η ανάκληση πληροφοριών, ή η διαδικαστική δεξιότητα;

### 5.3.4 Χαρακτηριστικά υποδείγματα rubrics (ρουμπρικών)

Στη συνέχεια θα προχωρήσουμε στην παρουσίαση κάποιων χαρακτηριστικών rubrics, που σχεδιάστηκαν για να αξιολογήσουν την επίδοση των μαθητών σε διάφορα γνωστικά αντικείμενα, να αναδείξουν τις δεξιότητες τους καθώς και να επισημάνουν τις αδυναμίες τους.

#### **Rubrics 1.**

**Πίνακας 1. Κατασκευή ρούμπρικας για την αξιολόγηση της συνεργατικότητας των μαθητών της Στ' τάξης.**

Ονοματεπώνυμο μαθητή / τριας: \_\_\_\_\_

Μάθημα: \_\_\_\_\_

Κριτήρια	Αρχικό 1	Αναπτυσσόμενο 2	Σταθεροποιημένο 3	Υποδειγματικό 4
<b>A) Συνεισφορά στην ομάδα.</b>				
<input type="checkbox"/> <i>Εγκαιρότητα.</i>	Δεν ολοκληρώνει τις εργασίες του.	Ολοκληρώνει τις εργασίες του αλλά με καθυστέρηση.	Συνήθως ολοκληρώνει τις εργασίες του έγκαιρα.	Ολοκληρώνει όλες τις εργασίες του έγκαιρα.
<input type="checkbox"/> <i>Συλλογή ερευνητικών στοιχείων.</i>	Δε συγκεντρώνει πληροφορίες.	Συγκεντρώνει μερικές πληροφορίες.	Συγκεντρώνει μικρό αριθμό σχετικών πληροφοριών.	Συγκεντρώνει μεγάλο αριθμό σχετικών πληροφοριών.
<input type="checkbox"/> <i>Ανταλλαγή πληροφοριών.</i>	Κρατάει τις πληροφορίες για τον εαυτό του και δεν τις μοιράζεται με τα υπόλοιπα μέλη.	Μοιράζεται μερικές μόνο πληροφορίες με τα υπόλοιπα μέλη.	Μοιράζεται σημαντικές μόνο πληροφορίες με τα υπόλοιπα μέλη.	Μοιράζεται όλες τις πληροφορίες με τα υπόλοιπα μέλη.
<b>B) Συνεργασία μέσα στην ομάδα.</b>				
<input type="checkbox"/> <i>Συνεργασία με τα υπόλοιπα μέλη της ομάδας.</i>	Δε συνεργάζεται με τα υπόλοιπα μέλη.	Σπάνια συνεργάζεται με τα υπόλοιπα μέλη.	Συνήθως συνεργάζεται με τα υπόλοιπα μέλη.	Πάντα συνεργάζεται με τα υπόλοιπα μέλη.
<input type="checkbox"/> <i>Διαλεκτική και διαλογική δεξιότητα.</i>	Πάντα μιλάει και ποτέ δεν αφήνει τα άλλα μέλη να μιλήσουν.	Συνήθως μιλάει τον περισσότερο χρόνο και σπάνια αφήνει τα άλλα μέλη να μιλήσουν.	Μερικές φορές μιλάει αρκετά, αλλά έχει αναπτυγμένη τη δεξιότητα ακρόασης.	Είναι στον ίδιο βαθμό ικανός ομιλητής και ακροατής.
<input type="checkbox"/> <i>Διενέργεια αντικειμενικών εκτιμήσεων.</i>	Θέλει να υπερισχύει πάντα η άποψή του/της.	Συχνά συμφωνεί με μέλη που θεωρεί φίλους του και δεν εκτιμά τις απόψεις όλων των μελών.	Συνήθως εκτιμά όλες τις απόψεις.	Λαμβάνει υπόψη του και επεξεργάζεται όλες τις απόψεις.
<b>Γ) Υποχρεώσεις προς τα υπόλοιπα μέλη της ομάδας.</b>				
<input type="checkbox"/> <i>Εκτέλεση καθηκόντων.</i>	Δεν εκτελεί καθήκοντα.	Εκτελεί μόνο τα καθήκοντα που θέλει.	Εκτελεί σχεδόν όλα τα καθήκοντά του/της.	Εκτελεί όλα τα καθήκοντά του/της.
<input type="checkbox"/> <i>Ανάληψη υποχρεώσεων και πρωτοβουλιών.</i>	Πάντα βασίζεται στα άλλα μέλη να κάνουν τις εργασίες.	Σπάνια εκτελεί τις εργασίες και χρειάζεται συνεχή υπενθύμιση.	Συνήθως εκτελεί τις εργασίες και σπάνια χρειάζεται υπενθύμιση.	Πάντα εκτελεί τις εργασίες και δε χρειάζεται υπενθύμιση.




## Rubrics 2.

### Πίνακας 2. Κατασκευή ρούμπρικας για την αξιολόγηση της επίδοσης των μαθητών στα Μαθηματικά.

Όνοματεπώνυμο μαθητή / τριας: \_\_\_\_\_

Μάθημα: \_\_\_\_\_

Name: _____ Date: _____ Math Problem Solving Rubric					
	Levels (Criteria)				
Dimensions (Categories)	Expert 4	Practitioner 3	Apprentice 2	Novice 1	Points
<b>Understanding</b>	<p>The solution shows a deep understanding of the math concepts and the procedures needed to reach it.</p> <p>Math concepts and procedures are applied correctly.</p>	<p>The solution is complete.</p> <p>Math concepts and procedures are applied correctly.</p>	<p>A solution is attempted but isn't complete.</p> <p>Some math concepts are used but not all of the necessary ones.</p> <p>Some, but not all, procedures are correct.</p>	<p>There isn't a solution or the solution is inappropriate.</p> <p>Inappropriate math concepts or procedures are used.</p>	
<b>Reasoning</b>	<p>Uses an efficient strategy that leads directly to a correct solution.</p> <p>Can verify the</p>	<p>Uses a strategy that leads to a solution.</p> <p>Uses effective math</p>	<p>Knows some of what is needed to do find a solution but doesn't find a</p>	<p>No evidence of a strategy or the strategy shown is inappropriate.</p> <p>There are many errors in math</p>	

	<p>solution and evaluate the reasonableness of it.</p> <p>Makes relevant math observations and connections.</p>	<p>reasoning and procedures.</p> <p>All parts of the solution are correct.</p>	<p>complete solution.</p> <p>Does not complete all of the math procedures that the problem needs.</p> <p>Some parts may be right but the right answer is not achieved.</p>	<p>procedural so that a solution can't be reached.</p>	
<p><b>Communication</b></p>	<p>Writing and drawing is clear, well-organized and detailed.</p> <p>All steps are included.</p> <p>A variety of words and symbols are used accurately and appropriately. Sophisticated language is used in some parts of the solution.</p>	<p>Writing and drawing is clearly done. The reader may have to fill in some details.</p> <p>A variety of words and symbols are used accurately and appropriately.</p>	<p>Writing and drawing may be unclear in parts.</p> <p>Words and symbols are used but show errors or lack of variety.</p>	<p>Writing and drawing is unclear or inappropriate.</p> <p>Words or symbols were used inaccurately or inappropriately.</p>	

### Rubrics 3.

#### **Πίνακας 3. Κατασκευή ρούμπρικας για την αξιολόγηση της επίδοσης των μαθητών στην Γεωγραφία.**

Όνοματεπώνυμο μαθητή / τριας: \_\_\_\_\_

Μάθημα: \_\_\_\_\_

### **World Map Rubric**

<b>Rating</b>	<b>Criteria</b>
Excellent 4	Correctly filled out all eight steps of the word map. Steps: 1. Cited word and page number. 2. Cited the sentence or phrase in which the word appears. 3. Recorded the contextually correct definition. 4. Recorded synonym. 5. Recorded antonym or nonexample (if applicable). 6. Recorded other word forms. 7. Made a personal connection or sketch of the word. 8. Created an original sentence using the word.
Very Good 3	Correctly filled out at least six steps of the word map, including steps 2, 3, 7, and 8.
Good 2	Correctly filled out at least four steps of the word map, including steps 2 and 3.
Needs Improvement 1	Correctly filled out at least two steps of the word map, including step 2.
Poor 0	Made no attempt to map out the vocabulary word.

Comments \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Evaluated By:		
Self	Peer	Teacher
_____	_____	_____

### Rubrics 4.

**Πίνακας 4. Κατασκευή ρούμπρικας για την αξιολόγηση της επίδοσης των μαθητών στο αντικείμενο του προγραμματισμού.**

Όνοματεπώνυμο μαθητή / τριας: \_\_\_\_\_

Μάθημα: \_\_\_\_\_

### Scratch Rating Rubric

Criterion	3 = Excellent	2 = Adequate	1 = Substandard	0 = Not even close
<b>A</b> dherence to standards Code blocks incorporate Color, design, movement and <b>purpose</b>	No errors	Minor details are overlooked Some of the standards are unclear. Purpose and audience hard to determine	Significant details of the are missing, but the program still fulfills essential functions.	Misses the point of the problem. Purpose hard to determine
<b>D</b> esign Displays order and sequence	No errors	1-3 minor errors	More than three minor errors OR one major error	More than one major error
<b>C</b> orrectness of code Does it work? <i>This category counts double.</i>	Works correctly in all cases.	Works for typical input, may fail for minor special cases.	Fails for typical input, for a minor reason.	Fails for typical input, for a significant reason.
<b>U</b> ser interface Is it clear and easy to manage. Control keys marked and labeled Is the us	No errors	1-3 minor errors	More than three minor errors OR one major error	More than one major error
<b>U</b> se of code Does it use the language features well?	No errors	1-3 minor errors	More than three minor errors OR one major error	More than one major error
<b>F</b> ast coding Is the programmer at ease with the language?	Varies with problem			
<b>G</b> eneral knowledge	Varies with problem			

## 6 Διδακτικό υλικό για τον μαθητή και τον εκπαιδευτικό

---

### 6.1 Για μια ουσιαστική διδασκαλία του προγραμματισμού

Η αφετηρία για έναν οποιουδήποτε είδους σχεδιασμό της διδασκαλίας του Προγραμματισμού έγκειται στον καθορισμό του επιδιωκόμενου μέσω αυτής στόχου. Στόχου ο οποίος θα πρέπει να είναι συνυφασμένος και με τις γενικότερες επιδιώξεις συνολικά του μαθήματος της Πληροφορικής στο δημοτικό.

Σε αυτές τις ηλικίες σκοπός της διδασκαλίας, τόσο της Πληροφορικής, όσο και ειδικότερα του Προγραμματισμού, θα πρέπει να είναι ο εξής: πώς να μάθουμε στους μαθητές να σκέφτονται ως λιλιπούτσιοι επιστήμονες της Πληροφορικής. Πώς δηλαδή θα αναπτύξουμε στους μαθητές μερικές από τις καλύτερες δεξιότητες που συναντούμε στα μαθηματικά, τη μηχανική και τις φυσικές επιστήμες: Όπως και οι μαθηματικοί, οι επιστήμονες της Πληροφορικής χρησιμοποιούν τυπικές γλώσσες για να καθορίσουν κάποιες ιδέες, κυρίως υπολογιστικές. Όπως και οι μηχανικοί, σχεδιάζουν πράγματα, συναρμολογώντας επιμέρους στοιχεία σε συστήματα, επιλέγοντας ανάμεσα σε εναλλακτικές λύσεις. Τέλος, όπως οι φυσικοί επιστήμονες, παρατηρούν τη συμπεριφορά πολύπλοκων συστημάτων, διατυπώνουν υποθέσεις και δοκιμάζουν την ισχύ των προβλέψεων τους.

Η πρωταρχική και μοναδική ίσως δεξιότητα ενός επιστήμονα Πληροφορικής είναι η επίλυση προβλημάτων. Αυτό σημαίνει την ικανότητα να τυποποιεί τα προβλήματα, να σκέφτεται δημιουργικά για τις λύσεις τους και να διατυπώνει μία λύση με σαφήνεια και ακρίβεια. Όπως έχει αποδειχθεί, η διαδικασία της εκμάθησης του Προγραμματισμού αποτελεί μια θαυμάσια ευκαιρία για την εξάσκηση δεξιοτήτων επίλυσης προβλημάτων.

Για να επιτευχθεί όμως ένας τέτοιος στόχος, όπως παρουσιάστηκε και στο κεφάλαιο 3, η προσέγγιση δεν ενδείκνυται να είναι αλγοριθμική αλλά ούτε και να βασίζεται στην κλασική μέθοδο διδασκαλίας προγραμματισμού.

Ως εναλλακτική, πιο αποτελεσματική προσέγγιση για την εισαγωγή μαθητών μικρής ηλικίας στον προγραμματισμό έχουν προταθεί μαθησιακά μίνι-περιβάλλοντα (mini-environments) που βασίζονται σε μίνι-γλώσσες (mini-languages) και

μικρόκοσμους (microworlds) (Brusilovsky et al., 1997). Από την παραπάνω παρουσίαση καθίσταται προφανές ότι τα προγραμματιστικά μινι-περιβάλλοντα συνιστούν ένα απλό αλλά ισχυρό μέσο για την εισαγωγή των μαθητών στις βασικές αρχές του προγραμματισμού και στη συστηματική επίλυση προβλημάτων, ενώ παράλληλα παρέχουν τα θεμέλια για τη μετέπειτα εκμάθηση μιας γλώσσας προγραμματισμού γενικού σκοπού.

## **6.2 Εργαλεία ανάπτυξης διδακτικού υλικού**

### **6.2.1 Scratch: Ισχυρό προγραμματιστικό περιβάλλον**

Ο διδακτικός μικρόκοσμος προγραμματισμού Scratch θεωρείται από μια μεγάλη μερίδα μαθητών, εκπαιδευτικών και ερευνητών ως το πιο αποδοτικό και αποτελεσματικό εργαλείο ανάπτυξης γνωστικών δεξιοτήτων προγραμματισμού σε μικρές ηλικίες.

Στην ενότητα αυτή, θα παρουσιάσουμε κάποιους από τους λόγους που μας οδήγησαν να χρησιμοποιήσουμε το περιβάλλον του Scratch για την ανάπτυξη διδακτικού υλικού διδασκαλίας του προγραμματισμού σε μαθητές του δημοτικού.

Είναι ευρέως γνωστό στην εκπαιδευτική κοινότητα ότι το Scratch έχει σχεδιαστεί έτσι ώστε να ευνοεί τους αρχάριους προγραμματιστές. Στοχεύει στην ανάπτυξη βασικών ικανοτήτων, όπως είναι: η δημιουργική σκέψη, η σαφής επικοινωνία, η συστηματική ανάλυση, η αποδοτική συνεργασία, ο επαναληπτικό-προοδευτικός σχεδιασμός, και οι δεξιότητες της δια βίου μάθησης (Monroy-Hernández & Resnick, 2008). Λόγω του ότι φέρει καινοτομίες στην προσέγγιση δυσνόητων εννοιών και τεχνικών προγραμματισμού, η διάδοση του στην εκπαίδευση και την εξωσχολική ενασχόληση των παιδιών βελτιώνει την σχέση των παιδιών με την επιστήμη των υπολογιστών γενικά, ενώ ταυτόχρονα καταστεί τον προγραμματισμό αντικείμενο περισσότερο ενδιαφέρον.

Ο σχεδιασμός του με τα δομικά στοιχεία επιτρέπει τον εύκολο προγραμματισμό με εξάλειψη των λαθών στην σύνταξη, επιτρέποντας ανάδραση από τον χώρο που είναι στοιβαγμένα τα δομικά στοιχεία και δίνοντας άμεση ανάδραση για πειραματισμό. Παράλληλα, υποστηρίζει τη διερεύνηση και επίλυση προβλημάτων και ενθαρρύνει την κριτική σκέψη, μια εξαιρετικά σημαντική δεξιότητα στο σημερινό

διαρκώς μεταβαλλόμενο κόσμο, ενώ οδηγεί τους χρήστες του στην αναζήτηση καινοτόμων λύσεων σε απρόβλεπτα προβλήματα.

Επίσης, αξίζει να αναφερθεί ότι οι αρχάριοι με τη βοήθεια του Scratch όχι μόνο μαθαίνουν να λύνουν ένα πλήρως καθορισμένο πρόβλημα αλλά και να αντιμετωπίζουν και όποιο άλλο ασθενώς δομημένο προκύψει (Δαπόντες, 2008).

Συμπερασματικά, μπορούμε να πούμε ότι η ιδιαιτερότητα του Scratch δημιουργεί ένα εντελώς νέο περιβάλλον εργασίας για τους μαθητές με τα εξής χαρακτηριστικά:

- ✚ Απλότητα, χωρίς περιορισμό των προσφερόμενων δυνατοτήτων προγραμματισμού.
- ✚ Περιέχει γραφικά και ήχους υψηλής ποιότητας που είναι ελκυστικά αλλά όχι ερεθιστικά.
- ✚ Αντικατοπτρίζει δραστηριότητες του πραγματικού κόσμου.
- ✚ Οπτική διαχείριση των βασικών αντικειμένων ελέγχου.
- ✚ Πολλαπλή αναπαράσταση των βασικών αντικειμένων (π.χ. εικόνα, σύμβολο, νέο σχέδιο).
- ✚ Υποστήριξη ομαδικής εργασίας και ανταλλαγής απόψεων – συζητήσεων, κατά τη φάση ανάπτυξης ή αντιμετώπισης ενός προβλήματος.
- ✚ Εφικτή εφαρμογή σε υπάρχοντα σχολικά περιβάλλοντα (π.χ. κόστος απόκτησης, συντήρηση υποδομής, γνωστικές απαιτήσεις από τους διδάσκοντες).

## **6.2.2 Rubrics (Ρούμπρικες): Ισχυρά εργαλεία αυτό-αξιολόγησης και ανατροφοδότησης για τη διδασκαλία του προγραμματισμού.**

Πολλοί εμπειρογνώμονες θεωρούν ότι οι ρούμπρικες αποτελούν εξαιρετικά χρήσιμα εργαλεία αξιολόγησης και ανατροφοδότησης τόσο για τους εκπαιδευτικούς όσο και για τους μαθητές και μπορούν να χρησιμοποιηθούν για να αξιολογήσουν μια ποικιλία προφορικών και γραπτών δραστηριοτήτων και εργασιών.

Στην ενότητα αυτή, θα αναλύσουμε τους κυριότερους λόγους που συντέλεσαν να οδηγηθούμε στον σχεδιασμό και στην ανάπτυξη ειδικών φορμών αυτό-αξιολόγησης rubrics με σκοπό την αξιολόγηση της επίδοσης των μαθητών στο αντικείμενο του προγραμματισμού.

Βασικό χαρακτηριστικό του είδους αυτού της αξιολόγησης είναι η συστηματική και λεπτομερής καταγραφή της επίδοσης του μαθητή (εργασίες, παρουσιάσεις, συμπεριφορά, προσαρμοστικότητα, επίλυση προβλημάτων, φάκελοι μαθητή, projects) μέσα από τη χρησιμοποίηση συγκεκριμένων, προεπιλεγμένων κριτηρίων. Τα κριτήρια αυτά είναι προσαρμοσμένα στη μαθησιακή διαδικασία και έχουν καθολική εφαρμογή στους μαθητές της τάξης της οποίας εφαρμόζεται η αξιολόγηση. Επίσης επιτρέπει και ενθαρρύνει τους μαθητές να γνωρίζουν τις διαβαθμίσεις της επίδοσής τους καθώς και πως μπορούν να τη βελτιώσουν (Mitchell, 2006).

Ενδεικτικά, κάποια από τα πλεονεκτήματα των ρουμπρικών που μας οδήγησαν να τις εντάξουμε και να τις χρησιμοποιήσουμε για την αξιολόγηση των μαθητών στα πλαίσια ανάπτυξης διδακτικού υλικού για την διδασκαλία του προγραμματισμού παρουσιάζονται παρακάτω:

- ✚ Οι ρούμπρικες παρουσιάζουν σαφώς στους μαθητές πώς η εργασία τους θα αξιολογηθεί και τι αναμένεται από αυτούς, βελτιώνοντας έτσι με αυτό τον τρόπο την επίδοσή τους και δίνοντας τους παράλληλα την ευκαιρία να ασχοληθούν με την δραστηριότητα του προγραμματισμού εκτός τάξης.
- ✚ Οι ρούμπρικες βοηθούν τους μαθητές να κρίνουν την ποιότητα της εργασίας τους, δίνοντας τους έτσι το κίνητρο για προσωπική ενασχόληση με το εκπαιδευτικό περιβάλλον προγραμματισμού Scratch.
- ✚ Οι ρούμπρικες προάγουν την διαδικασία εκμάθησης του προγραμματισμού με την βοήθεια του περιβάλλοντος Scratch.
- ✚ Οι ρούμπρικες επιτρέπουν στην αξιολόγηση να είναι αντικειμενικότερη, συνεπής και ακριβής.
- ✚ Οι ρούμπρικες δίνουν μια νέα διάσταση στην διδακτική και μαθησιακή διαδικασία του προγραμματισμού παραγκωνίζοντας τις τυχόν αδυναμίες των εκπαιδευτικών να εντάξουν και να χρησιμοποιήσουν τις ΤΠΕ στην καθημερινή τους διδακτική πρακτική.
- ✚ Οι ρούμπρικες διευκολύνουν τον εκπαιδευτικό ρόλο του δασκάλου που δεν έχει προηγούμενη εμπειρία με τον προγραμματισμό.
- ✚ Οι ρούμπρικες μειώνουν το χρονικό διάστημα που ο δάσκαλος σπαταλά προκειμένου να αξιολογήσει τις εργασίες των μαθητών.



- ✚ Οι ρούμπρικες βοηθούν τους μαθητές να συνειδητοποιήσουν τα κριτήρια που χρησιμοποιούν για την αξιολόγηση της συνεργατικής απόδοσης.
- ✚ Οι ρούμπρικες παρέχουν χρήσιμη ανατροφοδότηση (feedback) στο δάσκαλο σχετικά με την αποτελεσματικότητα της διδασκαλίας.
- ✚ Οι ρούμπρικες παρέχουν στους σπουδαστές περισσότερη πληροφοριακή ανατροφοδότηση (feedback) σχετικά με τις γνώσεις τους, τις δεξιότητες τους καθώς και τις αδυναμίες τους.
- ✚ Οι ρούμπρικες είναι εύχρηστες και πολύ εύκολο να εξηγηθούν.
- ✚ Οι ρούμπρικες είναι πολύ εύκολο να κατανοηθούν από τους μαθητές χωρίς ιδιαίτερες επεξηγήσεις από τον δάσκαλο.

#### **6.2.2.1 Ερωτήσεις που θέσαμε πριν την κατασκευή των rubrics (ρουμπρίκων).**

Μερικές από τις ερωτήσεις που θέσαμε πριν τον σχεδιασμό και την ανάπτυξη των ρουμπρίκων είναι οι εξής:

- ✚ Ποιες γνώσεις, έννοιες, ή δεξιότητες θέλω να διαπιστώσω αν θα κατακτηθούν από τους μαθητές;
- ✚ Τι ακριβώς οι μαθητές πρέπει να γνωρίζουν πριν την έναρξη της διδακτικής / μαθησιακής διαδικασίας ;
- ✚ Ποιο είναι το γενικό γνωστικό επίπεδο το οποίο αναμένεται να κατακτηθεί / εμπλουτιστεί από τους μαθητές;
- ✚ Ποιο είναι το επίπεδο της γνώσης και των δεξιοτήτων τα οποία αξιολογούνται; Για παράδειγμα, είναι η κριτική ικανότητα, η ανάκληση πληροφοριών, ή η διαδικαστική δεξιότητα;

### **6.3 Παρουσίαση διδακτικού υλικού**

Στα πλαίσια της παρούσης εργασίας αναπτύξαμε μια πλήρη σειρά διδακτικών ενοτήτων για την εισαγωγή των μαθητών στον προγραμματισμό με την βοήθεια του μικρόκοσμου Scratch. Πιο συγκεκριμένα, η σειρά αυτή χωρίζεται σε 10 ενότητες.

Η κάθε ενότητα αποτελείται από:

- ✚ Ένα αρχείο παρουσίασης PowerPoint, το οποίο χρησιμοποιείται για να εισάγει τους μαθητές στις βασικές έννοιες του Scratch και του προγραμματισμού.
- ✚ Μια έως δυο ασκήσεις-παιχνίδια, οι οποίες έχουν την μορφή οδηγού που καθοδηγεί τους μαθητές βήμα προς βήμα στη λύση του προβλήματος.
- ✚ Την εκφώνηση μιας άσκησης-παιχνιδιού χωρίς την αναλυτική περιγραφή των βημάτων που απαιτούνται για την υλοποίηση του παιχνιδιού.
- ✚ Την φόρμα αυτό-αξιολόγησης rubrics (ρούμπρικες), η οποία έχει σχεδιαστεί με βάση το αντίστοιχο παιχνίδι που καλείται να υλοποιήσει ο μαθητής και έχει ως διδακτικό στόχο την αξιολόγηση της επίδοσης των μαθητών στο αντικείμενο του προγραμματισμού.

Ακόμη, το γεγονός ότι οι εκπαιδευτικοί δεν διαθέτουν το κατάλληλο γνωστικό υπόβαθρο που απαιτείται για την διδασκαλία του προγραμματισμού στο δημοτικό, μας οδήγησε να αναπτύξουμε έναν οδηγό εκμάθησης του περιβάλλοντος Scratch (tutorial) και μια σειρά από 14 projects με σκοπό την πρόσκτηση βασικών δεξιοτήτων προγραμματισμού από τους εκπαιδευτικούς.

Συγκεκριμένα, ο οδηγός εκμάθησης του Scratch έχει ως στόχο να κατανοήσουν οι εκπαιδευτικοί τη χρήση και λειτουργία του περιβάλλοντος Scratch αλλά και των βασικών δομών προγραμματισμού που υποστηρίζονται μέσα από αυτό.

Τα 14 projects έχουν σχεδιαστεί με απώτερο στόχο:

- ✚ Την εξοικείωση των δασκάλων με το περιβάλλον του Scratch.
- ✚ Την ανάπτυξη δεξιοτήτων επίλυσης προβλημάτων.
- ✚ Την οικοδόμηση των βασικών προγραμματιστικών εννοιών.
- ✚ Να διεγείρουν την φαντασία των δασκάλων για τον σχεδιασμό και την ανάπτυξη νέων ασκήσεων-παιχνιδιών που θα έχουν ως στόχο την ανάπτυξη προγραμματιστικών ικανοτήτων και δεξιοτήτων σε μαθητές μικρής ηλικίας.

### 6.3.1 Ο φάκελος του μαθητή

Συγκεκριμένα, ο φάκελος του μαθητή περιέχει τα παρακάτω:

#### Μάθημα 1<sup>ο</sup>: Εισαγωγή και αντικείμενα

##### Λιδακτικός στόχος:

- Γνωριμία με το περιβάλλον του Scratch.
- Εισαγωγή αντικειμένων στην σκηνή του προγράμματος.

##### Προγενέστερη γνώση:

- Βασικές γνώσεις χρήσης Η/Υ.

##### Έτοιμο παιχνίδι:

- «Η γατούλα μας έχει τα γενέθλια της.»

##### Φτιάξε το δικό σου παιχνίδι:

- «Η γατούλα μας πάει διακοπές.»
- Rubrics 1

#### Μάθημα 2ο: Διαγραφή και διπλασιασμός αντικειμένων

##### Λιδακτικός στόχος:

- Γνωριμία με το περιβάλλον του Scratch.
- Διπλασιασμός των αντικειμένων που βρίσκονται στην σκηνή του προγράμματος.
- Διαγραφή αντικειμένων από την σκηνή του προγράμματος.

##### Προγενέστερη γνώση:

- Βασικές γνώσεις χρήσης Η/Υ.

##### Έτοιμα παιχνίδια:

- «Η γατούλα μας δεν έχει χρήματα.»
- «Η γατούλα μας έχει δίδυμη αδελφή»

##### Φτιάξε το δικό σου παιχνίδι:

- «Η γατούλα μας και τα γατάκια της.»
- Rubrics 2

### Μάθημα 3ο: Η πράσινη σημαία και οι εντολές κίνησης

#### Διδακτικός στόχος:

- *Η σημασία της πράσινης σημαίας.*
- *Εισαγωγή κίνησης στα αντικείμενα που υπάρχουν στην σκηνή του προγράμματος.*

#### Προγενέστερη γνώση:

- *Βασικές γνώσεις χρήσης H/Y.*
- *Εισαγωγή αντικειμένων στην σκηνή του προγράμματος.*

#### Έτοιμο παιχνίδι:

- «Η γατούλα μας πάει βόλτα.»

#### Φτιάξε το δικό σου παιχνίδι:

- «Ο σκύλος και η γατούλα μας.»
- Rubrics 3

### Μάθημα 4ο: Ηχητικά εφέ

#### Διδακτικός στόχος:

- *Εισαγωγή ήχου στα αντικείμενα που υπάρχουν στην σκηνή του προγράμματος.*

#### Προγενέστερη γνώση:

- *Βασικές γνώσεις χρήσης H/Y.*
- *Εισαγωγή αντικειμένων στην σκηνή του προγράμματος.*
- *Διαγραφή αντικειμένων από την σκηνή του προγράμματος.*

#### Έτοιμο παιχνίδι:

- «Η γατούλα μας κάνει Νιάου.»

**Φτιάξε το δικό σου παιχνίδι:**

- «Ο Γιαννάκης στο Ωδείο.»
- Rubrics 4

**Μάθημα 5ο:** Πολλά τουβλάκια στη σειρά.

**Διδακτικός στόχος:**

- Σειριακή εκτέλεση εντολών (πολλά τουβλάκια το ένα κάτω από το άλλο).
- Σύνταξη προγραμμάτων που χρησιμοποιούν την βασική δομή της ακολουθίας.

**Προγενέστερη γνώση:**

- Βασικές γνώσεις χρήσης H/Y.
- Εισαγωγή αντικειμένων στην σκηνή του προγράμματος.
- Εισαγωγή κίνησης στα αντικείμενα που υπάρχουν στην σκηνή του προγράμματος.
- Εισαγωγή ήχου στα αντικείμενα που υπάρχουν στην σκηνή του προγράμματος.

**Έτοιμο παιχνίδι:**

- «Η γατούλα μας μαθαίνει να παίζει μπάσκετ.»

**Φτιάξε το δικό σου παιχνίδι:**

- «Η γατούλα μας τρώει το ποντικάκι.»
- Rubrics 5

**Μάθημα 6ο:** Αριθμοί

**Διδακτικός στόχος:**

- Μετατροπή των αριθμητικών πράξεων σε εντολές προγράμματος.

**Προγενέστερη γνώση:**

- Βασικές γνώσεις χρήσης H/Y.

## Μάθημα 7ο: Μεταβλητές

### Διδακτικός στόχος:

- Εισαγωγή στην έννοια της μεταβλητής.
- Δημιουργία μιας μεταβλητής.
- Ανάθεση (ή εκχώρηση) τιμής σε μια μεταβλητή.
- Αλλαγή τιμής σε μια μεταβλητή.

### Προγενέστερη γνώση:

- Βασικές γνώσεις χρήσης H/Y.
- Άνοιγμα και τροποποίηση υλοποιημένων παιχνιδιών στο Scratch.

### Έτοιμο παιχνίδι:

- «Η Αννούλα και το ενυδρείο της.»

### Φτιάξε το δικό σου παιχνίδι:

- «Η γατούλα μας και οι μπάλες της.»
- Rubrics 7

## Μάθημα 8ο: Έλεγχος και αριθμοί

### Διδακτικός στόχος:

- Εισαγωγή στην δομή επιλογής (ελέγχου).
- Σχηματισμός λογικών εκφράσεων, απλών και σύνθετων.
- Σύνταξη προγραμμάτων που χρησιμοποιούν τις βασικές δομές της ακολουθίας και της επιλογής.

### Προγενέστερη γνώση:

- Βασικές γνώσεις χρήσης H/Y.
- Άνοιγμα και τροποποίηση υλοποιημένων παιχνιδιών.
- Μετατροπή των αριθμητικών πράξεων σε εντολές προγράμματος.

### Έτοιμο παιχνίδι:

- «Η Μαιρούλα κάνει τα Μαθηματικά της.»

**Φτιάξε το δικό σου παιχνίδι:**

- «Η Μαιρούλα έχει και άλλη απορία στα Μαθηματικά της.»
- Rubrics 8

**Μάθημα 9ο:** Έλεγχος και Μεταβλητές

**Λιδακτικός στόχος:**

- Οικοδόμηση της δομής επιλογής (ελέγχου).
- Σχηματισμός λογικών εκφράσεων, απλών και σύνθετων.
- Ένθεση μιας μεταβλητής σε μια δομή επιλογής (ελέγχου).
- Σύνταξη προγραμμάτων που χρησιμοποιούν τις βασικές δομές της ακολουθίας και της επιλογής.

**Προγενέστερη γνώση:**

- Βασικές γνώσεις χρήσης H/Y.
- Άνοιγμα και τροποποίηση υλοποιημένων παιχνιδιών.
- Μετατροπή των αριθμητικών πράξεων σε εντολές προγράμματος.
- Δημιουργία μιας μεταβλητής.
- Ανάθεση (ή εκχώρηση) τιμής σε μια μεταβλητή.
- Αλλαγή τιμής σε μια μεταβλητή.

**Έτοιμο παιχνίδι:**

- «Ο Γιωργάκης και ο Κωστάκης στον κινηματογράφο.»

**Φτιάξε το δικό σου παιχνίδι:**

- «Ο Γιαννάκης πάει σχολείο.»
- Rubrics 9

**Μάθημα 10ο:** Επανάληψη

**Λιδακτικός στόχος:**

- *Εισαγωγή στην επαναληπτική διαδικασία.*
- *Οικοδόμηση και σχεδιασμός της επεξεργασίας, δηλαδή ανάπτυξη της διατύπωσης των εντολών που θα επαναληφθούν.*
- *Σύνταξη προγραμμάτων που χρησιμοποιούν τις βασικές δομές της ακολουθίας και της επανάληψης.*

#### **Προγενέστερη γνώση:**

- *Βασικές γνώσεις χρήσης H/Y.*
- *Άνοιγμα και τροποποίηση υλοποιημένων παιχνιδιών στο Scratch.*
- *Εισαγωγή αντικειμένων στη σκηνή του προγράμματος.*
- *Εισαγωγή κίνησης στα αντικείμενα που υπάρχουν στη σκηνή του προγράμματος.*

#### **Έτοιμο παιχνίδι:**

- «Η γατούλα μας και η πισίνα της.»

#### **Φτιάξε το δικό σου παιχνίδι:**

- «Ο μικρός Γιαννάκης γράφει το όνομά του.»
- Rubrics 10

### **6.3.2 Ο φάκελος του εκπαιδευτικού**

Συγκεκριμένα, ο φάκελος του εκπαιδευτικού περιέχει τα παρακάτω:

 Οδηγός εκμάθησης του Scratch

 Projects

- **Project 1:** Εισαγωγή στο περιβάλλον του Scratch.
- **Project 2:** Δημιουργία ενός Spinner.
- **Project 3:** Δημιουργία του ονόματος JOHN και εισαγωγή ενεργειών σε αυτό.
- **Project 4:** Δημιουργία ενός ενυδρείου μέσα στο οποίο κινούνται τα ψάρια.
- **Project 5:** Δημιουργία ενός παιχνιδιού με animation.
- **Project 6:** Δημιουργία του μικρόκοσμου “Gravity Blocks”.
- **Project 7:** Δημιουργία ενός ρολογιού στο περιβάλλον του Scratch.



- **Project 8:** Δημιουργία του μικρόκοσμου “κορώνα” ή “γράμματα”.
- **Project 9:** Δημιουργία ενός παιχνιδιού Ping-Pong.
- **Project 10:** Δημιουργία ενός μικρόκοσμου που αφορά τα μαθηματικά και συγκεκριμένα την πρόσθεση.
- **Project 11:** Δημιουργία του μικρόκοσμου “Ocean Music Box”.
- **Project 12:** Δημιουργία του μικρόκοσμου “Tamara”.
- **Project 13:** Δημιουργία του παιχνιδιού “Marble Racer”.
- **Project 14:** Δημιουργία του μικρόκοσμου “Basic Math Game”.

## **6.4 Συνοδευτικό CD-ROM**

Το συνοδευτικό CD-ROM που δίνεται με την παρούσα εργασία περιέχει το ψηφιακό διδακτικό υλικό που αναλύσαμε στην προηγούμενη ενότητα.

## 7 Επίλογος

---

### 7.1 Συμπεράσματα

Πιστεύουμε ότι οι διδακτικοί μικρόκοσμοι ανοίγουν μια νέα προοπτική στην εκπαιδευτική διαδικασία όχι μόνο ως μέσο διδασκαλίας εισαγωγικών εννοιών του Προγραμματισμού αλλά και ως μέσο αξιολόγησης και μελέτης των αντιλήψεων των μαθητών.

Για το σκοπό αυτό αναπτύξαμε το διδακτικό υλικό βασισμένοι στον μικρόκοσμο του Scratch. Η προσπάθεια μας αυτή συνίσταται στην ανάπτυξη ενός ολοκληρωμένου περιβάλλοντος το οποίο θα μπορεί να χρησιμοποιηθεί, τόσο από τον μαθητή ο οποίος θα ασχοληθεί με το υλικό μόνος, όσο και από τον μαθητή ο οποίος θα χρησιμοποιήσει το υλικό μέσα σε ένα σχολικό πλαίσιο με την επίβλεψη ενός διδάσκοντα. Παράλληλα με το διδακτικό υλικό αναπτύχθηκαν φόρμες αυτό-αξιολόγησης που θα παρέχονται στον μαθητή (rubrics). Αυτές αποβλέπουν όχι μόνο στην αξιολόγηση της τελικής επίδοσης, αλλά στην σταδιακή καταγραφή των αντιλήψεων των μαθητών όσον αφορά το αντικείμενο που διδάσκονται.

Συμπερασματικά, μπορούμε να πούμε ότι το εκπαιδευτικό περιβάλλον προγραμματισμού Scratch ανοίγει νέους ορίζοντες για την εκμάθηση βασικών εννοιών προγραμματισμού Η/Υ - και μάλιστα με έναν τρόπο που είναι ακόμη απλούστερος και της Logo. Η απλούστευση της διαχείρισης δεδομένων και αποτελεσμάτων (δηλαδή εισόδου και εξόδου), ο άμεσος και «προφανής» τρόπος ανταπόκρισης του συστήματος συνιστούν στοιχεία τα οποία, τουλάχιστον σε μια αρχική φάση, μπορούν να συντελέσουν με ουσιαστικό τρόπο στην υπέρβαση προβλημάτων κατανόησης που αντιμετωπίζουν οι αρχάριοι προγραμματιστές.

Βέβαια, ο διδακτικός μικρόκοσμος Scratch μπορεί να παρουσιάζει πολλά πλεονεκτήματα, αλλά τούτο δε σημαίνει ότι δεν παρουσιάζει και προβλήματα. Για παράδειγμα, η γνώση που θα αποκτήσουν οι μαθητές μπορεί να είναι υπερβολικά πλακαιοποιημένη, δηλαδή υπερβολικά εξαρτώμενη από το περιβάλλον του Scratch. Όπως μάλιστα τείνουν να υποστηρίζουν σχετικές εργασίες (Dagdilelis, Balasheff, Carroni, 1990), οι μαθητές δύσκολα γενικεύουν ή μεταφέρουν σε άλλα περιβάλλοντα προγραμματιστικές γνώσεις που έχουν αποκτηθεί σε συγκεκριμένα περιβάλλοντα.

Όσον αφορά την διαδικασία της αξιολόγησης των μαθητών και συγκεκριμένα τις φόρμες αυτό-αξιολόγησης rubrics, διαπιστώσαμε ότι δεν έχουν εφαρμοστεί από μια μεγάλη μερίδα εκπαιδευτικών. Αυτό οφείλεται κατά κύριο λόγο στο γεγονός ότι οι εκπαιδευτικοί δε γνωρίζουν όλες τις λεπτομέρειες αυτής της τεχνικής προκειμένου να εφαρμόσουν με επιτυχία την μέθοδο των rubrics στην εκπαιδευτική διαδικασία. Πιστεύουμε ότι οι ανωτέρω δυσκολίες μπορούν να αντιμετωπιστούν με τη διεξαγωγή επιμορφωτικών σεμιναρίων καθώς και με την ανταλλαγή απόψεων και προτάσεων μεταξύ των εκπαιδευτικών.

Ωστόσο, στην παρούσα εργασία πετύχαμε να αναπτύξουμε και να σχεδιάσουμε ειδικές φόρμες αυτό-αξιολόγησης rubrics με σκοπό την αξιολόγηση της επίδοσης των μαθητών στο αντικείμενο του προγραμματισμού. Στα πλαίσια κατασκευής των φορμών αυτό-αξιολόγησης διαπιστώσαμε ότι οι ρούμπρικες αποτελούν εξαιρετικά χρήσιμα εργαλεία αξιολόγησης και ανατροφοδότησης τόσο για τους εκπαιδευτικούς όσο και για τους μαθητές και μπορούν να χρησιμοποιηθούν για να αξιολογήσουν μια ποικιλία προφορικών και γραπτών δραστηριοτήτων και εργασιών.

## **7.2 Μελλοντική εργασία**

Η παρούσα εργασία ασχολείται με την ανάπτυξη διδακτικού υλικού για την διδασκαλία του προγραμματισμού στη πρωτοβάθμια εκπαίδευση. Το επόμενο στάδιο της έρευνας περιλαμβάνει τον σχεδιασμό και την ανάπτυξη μιας διαδραστικής εφαρμογής, με σκοπό να αποτελέσει μελλοντικά ένα διαδικτυακό περιβάλλον μάθησης. Συγκεκριμένα η εν λόγω εφαρμογή θα χρησιμοποιεί κατάλληλες φόρμες αυτό-αξιολόγησης rubrics, δίνοντας έτσι την ευκαιρία στους μαθητές να αποκτήσουν βασικές γνώσεις προγραμματισμού με βάση την δική τους προσωπική ενασχόληση.

Ένας πολύ σημαντικός μελλοντικός στόχος είναι η πιλοτική εφαρμογή του διδακτικού υλικού που αναπτύξαμε σε κάποιο σχολείο της πρωτοβάθμιας εκπαίδευσης για την εξαγωγή συμπερασμάτων αναφορικά με την παιδαγωγική του αποτελεσματικότητα. Συγκεκριμένα θα πρέπει να μελετήσουμε κατά πόσο το διδακτικό υλικό που κατασκευάσαμε, κατάφερε να κεντρίσει το ενδιαφέρον των μαθητών για περαιτέρω μάθηση κι ενασχόληση με το αντικείμενο του προγραμματισμού. Παράλληλα, θα πρέπει να ερευνήσουμε ποιά είναι η στάση και η ανταπόκριση των δύο φύλλων στην διδασκαλία του προγραμματισμού και στην

διδασκτική πρακτική που προτείνουμε. Ενδεχομένως να χρειαστεί να γίνουν κάποιες αλλαγές εφόσον κριθεί ότι οι μαθητές δεν απέδωσαν τα αναμενόμενα. Κρίνεται αναγκαίο να εξετάσουμε αν οι ίδιοι οι εκπαιδευτικοί διαθέτουν το κατάλληλο γνωστικό υπόβαθρο έτσι ώστε να ενσωματώσουν νέες μεθόδους και τεχνικές στην εκπαιδευτική διαδικασία. Όπως είναι γνωστό οι περισσότεροι δάσκαλοι έχουν υιοθετήσει παραδοσιακές μεθόδους διδασκαλίας και θα πρέπει να εντοπίσουμε ενδεχόμενες τροποποιήσεις στην διδασκτική τους συμπεριφορά.

Απώτερος στόχος μας είναι η εξοικείωση των εκκολαπτόμενων εκπαιδευτικών πρωτοβάθμιας εκπαίδευσης με το περιβάλλον του Scratch και με τη δραστηριότητα του προγραμματισμού γενικότερα. Κρίνεται απαραίτητο σε αυτό το σημείο να αναφέρουμε ότι για την επίτευξη του παραπάνω στόχου απαιτείται η ένταξη και ενσωμάτωση του εκπαιδευτικού εργαλείου Scratch σε κάποιο από τα μαθήματα πληροφορικής που διδάσκονται στα Παιδαγωγικά τμήματα. Βέβαια για να θεωρήσουμε ολοκληρωμένη την παραπάνω έρευνα επιβάλλεται να μελετήσουμε τις δυσκολίες των εκκολαπτόμενων εκπαιδευτικών στο Scratch. Πιο συγκεκριμένα, με την ολοκλήρωση της έρευνας θα πρέπει να είμαστε σε θέση να απαντήσουμε στα παρακάτω ερωτήματα: (i) Ποιες δυσκολίες αναφέρουν οι φοιτητές στην ανάπτυξη εφαρμογών με το Scratch; (ii) Ποιες οι προγραμματιστικές έννοιες που κατάφεραν οι φοιτητές να χρησιμοποιήσουν; Τι πολυπλοκότητα έχουν τα προγράμματα που φτιάχνουν οι φοιτητές; (iii) Ποια τα είδη των εφαρμογών που επέλεξαν οι φοιτητές να αναπτύξουν;

# Βιβλιογραφία

---

## Ξενογλώσση Βιβλιογραφία

Adams, J. C. (2007), *Scratch, middle schoolers & the imaginary worlds camps*, Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education, pp. 307-311, New York, NY: ACM Press.

Airasian, P.W. (1991), *Classroom assessment*, New York: McGraw-Hill.

Airasian, P. W. (1997), *Classroom Assessment*, (3η εκ.), New York: McGraw- Hill Companies, Inc.

Alessi, S. & Trollip, S. (2001), *Multimedia for learning: Methods and development*, Boston, MA: Allyn & Bacon.

Arsac, J. (1987), *Les machines a penser*, Des ordinateurs et des homes, Paris: Seuil.

Arter, J. (2000), *Rubrics, scoring guides, and performance criteria: Classroom tools for assessing and improving student learning*, Paper presented at the annual conference of the American Educational Research Association, New Orleans.

Ben-Ari, M. (2001), *Constructivism in Computer Science Education*, Journal of Computers in Mathematics and Science Teaching, 20(1), pp. 45-73.

Bergin, J., Stehlik, M., Roberts, J., Pattis, R. (1996), *Karel ++ - A Gentle Introduction to the Art of Object-Oriented Programming*, John Wiley.

Bergin, J., Stehlik, M., Roberts, J. and Pattis, R. (2002), *Karel J. Robot: A gentle Introduction to the Art of Object-Oriented Programming in Java*, Ανάκτηση στις 3 Μαΐου 2009 από το

<http://csis.pace.edu/~bergin/KarelJava2ed/Karel++JavaEdition.html>

- Bloom, B.S., Hastings, J.T., & Madaus, G. (1971), *Handbook and Summative Evaluation of Student Learning*, McGraw, New-York.
- Bonar, J. & Soloway, E. (1985), *Preprogramming knowledge: a major source of misconceptions in novice programmers*, *Human-Computer Interaction*, 1, pp. 133-161.
- Brualdi, A. (1999), *Performance-based Assessment: How Students Understand and Apply Knowledge*, Ανάκτηση στις 16 Μαΐου 2009 από το <http://wilsonweb2.hwwilson.com>
- Brusilovsky, P., Kouchnirenko, A., Miller, P., & Tomek, I. (1994), *Teaching Programming to novices: A review of approaches and tools*, In Proceedings of EDMEDIA '94, pp. 103–110.
- Brusilovsky, P., Calabrese, E., Hvorecky, J., Kouchnirenko, A. & Miller, P. (1997), *Mini-languages: A way to learn programming principles*, *Education and Information Technologies*, 2(1), pp. 65-83.
- Buck, D. and Stucki, D. (2001), *JKarelRobot: A Case Study in Supporting Levels Of Cognitive Development in the Computer Science Curriculum*, ACM SIGCSE Bulletin Proceedings of the Thirty-Second SIGCSE Technical Symposium, ACM Press 33(1), pp. 16-20.
- Chiocciariello, A., Manca, S. & Sarti, L. (2000), *Behaviour Construction Kit*, Proceedings of the i3 Annual Conference: Building Tomorrow Today: Community, Design and Technology, pp. 72-76.
- Costelloe, E. (2004), *Teaching programming: The state of the art*, Center for Research in IT in Education (CRITE) Technical Report, Dublin.

- Dagdilelis, V., Balasheff, N., Capponi, B. (1990), L' apprentissage de l' itération dans deux environnements informatiques, ASTER.
- Dahl, O.-J., Dijkstra, E.W. & Hoare, C.A.R, (1972), *Structured Programming*, Academic Press.
- Dijkstra, E.W. (1976), *A Discipline of Programming*, Englewood Clis, NJ: Prentice-Hall.
- Du Boulay, B. (1989), Some Difficulties of Learning to Program, in Soloway, E. & Spohrer, J. (Edited by), *Studying the Novice Programmer*, Lawrence Erlbaum Associates, pp. 283-314.
- Duchateau, C. (1991), *Actes du Deuxième Colloque sur la Didactique de l' Informatique*, Namur, Coédition Presses Universitaires de Namur-CeFIS-AFDI, Namur.
- Dufoyer, J.-P. (1988), *Informatique, éducation et psychologie de l'enfant*, Paris: PUF.
- Felder, R. (1996), *Matters of Style, Principles and applications of four learning style models*, ASEE Prism, 6(4), pp. 18-23.
- Felder, R. & Barbara A. Soloman (2000), *Learning Styles and Strategies*, Ανάκτηση στις 4 Ιουνίου 2009 από το <http://www2.ncsu.edu/unity/lockers/users/f/felder>
- Gilmore, D.J., Pheasey, K., Underwood, J. & Underwood, G. (1995), *Learning Graphical Programming: An Evaluation of KidSim*, In K. Nordby, P.H. Helmersen, D.J. Gilmore, S.A. Arnesen (Eds.), *Human-Computer Interaction: Interact'95*, London: Chapman & Hall, pp. 145-150.

- Green, T., Hoc, J.-M., Samurcay, R., Gilmore, D. (1990), *Psychology of Programming*, Academic Press.
- Gueraud, V. & Peyrin, J.P. (1988), *Un jeux de rôles pour l' enseignement de la programmation*, In Colloque Francophone sur la didactique de l' Informatique, pp. 47-60, Paris: EPI.
- Guzdial, M. & Soloway, E. (2002), *Teaching the Nintendo generation to program*, Communications of the ACM, 45(4), pp. 17–21.
- Hibbard, K. M. (1996), *A teacher's guide to performance-based learning and Assessment*, Alexandria, VA: Association for Supervision and Curriculum Development.
- Hoc, M. (1987), *La psychologie cognitive de la planification*, Grenoble: Presses Universitaires de Grenoble.
- Hoc, M., Green, T.R.G. Samurcay, R. & Gilmore, D.J. (1991), *Psychology of programming*, Academic Press.
- Hvorecky, J. (1992), *Karel the Robot for PC*, In Proceedings of the east-West Conference on Emerging Computer Technologies in Education, P. Brusilovsky and V. Stefanuk (eds.), Moscow, pp. 157-160.
- Jonassen, D. H. (2006), *Toward a Design Theory of Problem Solving*, Educational Technology Research and Development, 48 (4), pp. 63-85.
- Kafai, Y., Pepler, K. & Chiu, G. (2007), *High Tech Programmers in Low Income Communities: Seeding Reform in a Community Technology Center*, In C. Steinfield, B. Pentland, M. Ackerman, & N. Contractor (Eds.), Proceedings of Communities and Technologies 2007, pp. 545-564, New York: Springer.



- Kahney, H. (1993), *Problem Solving: Current Issues* (Open Guides to Psychology), London: Open University Press.
- Kahn, K. (2000), *Generalizing by Removing Detail: How Any Program Can Be Created by Working with Examples*, Ανάκτηση στις 10 Απριλίου 2009 από το <http://www.animated-programs.com/PBD>
- Kan, A. (2007), *An alternative method in the new educational program from the point of performance-based assessment: Rubric scoring scales*, Kuram ve Uygulamada Egitim Bilimleri, 7(1), pp. 144-153.
- Kane, M.B. & Mitchell, R. (1996), *Implementing Performance Assessment: Promises, Problem, and Challenges*, Hillsdale, NJ: Erlbaum.
- Kelleher, C. & Pausch, R. (2005), *Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers*, ACM Computing Surveys, 37(2), pp. 83-137.
- Kessler, C. M. & J. R. Anderson (1986), *A model of novice debugging in LISP*, in E. Soloway & S. Iyengar (Eds.), *Empirical Studies of Programmers*, pp. 198-212, Washington, DC, Ablex Publishing Corporation.
- Knuth, D. (1968), *The Art of Computer Programming*, Volume 1, Fundamental Algorithms, First Edition, Addison Wesley.
- Komis, V. (2001), *Didactics of Informatics: from the formation of the Scientific Field to the Conjunction among Research and School Practice*, in Manopoulos Y & Evripidou S (editors), *Proceedings of 8th Panhellenic Conference on Informatics with international participation*, Greek Computer Society, University of Cyprus, pp. 463-471.

- Lagrange, J.B. (1990), *Des situations connues aux traitements sur des données codifiées: une étude des difficultés d'élèves débutant en informatique*, In Actes du deuxième colloque francophone sur la didactique de l'informatique, pp. 55-79, EPI.
- Lane, S. (2004), *Validity of high-stakes assessment: Are students engaged in complex thinking?*, Education Measurement, Issues and Practice, 23(3), pp. 6-15.
- LEGO Mindstorms (2005), Τελευταία πρόσβαση: 22 Μαΐου 2009, Ιστοχώρος: <http://www.lego.com/eng/education/mindstorms/default.asp>
- Lieberman, H. (1986), *An Example Based Environment for beginning Programmers*, Instructional Science, 14(3), pp. 277-292.
- Linn, M. C. (1985), *The cognitive consequences of programming instruction in classrooms*, Educational Researcher, 14(5), pp. 14-16, 25-29.
- Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B. & Resnick, M. (2004), *Scratch: A Sneak Preview*, Second International Conference on Creating, Connecting, and Collaborating through Computing, Kyoto, Japan, pp. 104-109
- Mayer, R. (2002), *A taxonomy for computer-based assessment of problem solving*, Computers in Human Behavior, 18, pp. 623–632.
- McAteer, E. (2005), *Assessment Portfolio: a compilation of standardized tests and assessments for evaluating high school students in English language arts*, Unpublished portfolio, National University, California.
- Miglino, O., Hautop, H., & Cardaci, M. (1999), *Robotics as an Educational Tool*, Interactive Learning Research, Vol 10, No 1, pp. 25-44.

- Mitchell, A. (2006), *Introduction to rubrics: An assessment tool to save grading time, convey effective feedback and promote student learning*, *Journal of College Student Development*, 47(3), pp. 352-356.
- Monroy-Hernández, A. and Resnick, M. (2008), *Empowering kids to create and share programmable media*, *Interactions*.
- Morgado, L., Cruz, M.G.B. & Kahn, K. (2001), *Working in ToonTalk with 4- and 5-year olds*, *Playground International Seminar*, Ανάκτηση στις 24 Απριλίου 2009 από το <http://www.utad.pt/~leonelm/TTon4-5.html>
- Papert, S. (1980), *Mindstorms: Children, Computers and Powerfull Ideas*, New York: Basic Books.
- Papert, S. (1993), *The Children's, Machine*, Rethinking Schools in the Age of the Computer, New York: Harvester Wheatsheaf.
- Pattis, R. (1981), *Karel – The Robot*, A Gentle Introduction to the Art of Programming, London, Wiley.
- Pattis, R., Roberts, J., Stehlik, M. (1995), *Karel – The Robot*, A Gentle Introduction to the Art of Programming, 2nd edn., London, Wiley.
- Pea R. D. (1986), *Language-independent conceptual "bugs" in the novice programming*, *Journal of Educational Computing Research*, 2(1), pp. 25-36.
- Pennington, N. (1987), *Stimulus structures and mental representations in expert comprehension of computer programs*, *Cognitive Psychology*, 19, pp. 295-341.
- Peppler, K., & Kafai, Y. (2005), *Creative coding: The role of art and programming in the K-12 educational context*, Kyoto, Japan.

- Peppler, K. & Kafai, Y. B. (2007), *From SuperGoo to Scratch: exploring creative digital media production in informal learning*, Learning, Media, and Technology, 32(2), pp. 149–166.
- Piaget, J. (1969), *Science of education and the psychology of the child*, New York, Gossman publishers.
- Piaget, J. (1977), *The Development of Thought: Equilibration of Cognitive Structures*, New York, Viking.
- Popham, W. J. (1995), *Classroom assessment: What teachers need to know*, Needham Heights, MA: Allyn and Bacon.
- Reichert, R. (2003), *Theory of computation as a vehicle for teaching fundamental Concepts of computer science*, Dissertation No. 15035, ETH Zürich.
- Resnick, M. (2007), *Sowing the Seeds for a More Creative Society*, *Learning and Leading with Technology*, Ανάκτηση στις 27 Μαρτίου 2009 από το <http://web.media.mit.edu/~mres/papers/Learning-Leading-final.pdf>
- Rogalski, J. & Vergnaud, G. (1987), *Didactique de l'informatique et acquisitions cognitives en programmation*, Psychologie Française, 32 (2), pp. 267-273.
- Rogalski, J. (1988), *Enseignement de methods de programmation dans l' initiation a l' informatique*, In Colloque Francophone sur le didactique de l' informatique, pp. 61-74, Paris, EPI.
- Russell, J., Elton, L., Swinglehurst, D. & Greenhalgh, T. (2006), *Using the online environment in assessment for learning: A case-study of a Web-based course in primary care*, Assessment and Evaluation in Higher Education 31 (4): pp. 465–478.

- Smith, M. (1991), (Edited by), *Toward a Unified Theory of Problem Solving*,  
Lawrence Erlbaum Associates.
- Smith, D.C. & Cypher, A. (1999), *Making Programming Easier for Children*, In A.  
Druin (Ed.), *The Design of Children's Technology*, Morgan Kaufmann  
Publishers, pp. 201-221.
- Soloway, E. & Spohrer, J. (1989), (Edited by), *Studying the Novice Programmer*,  
Lawrence Erlbaum Associates.
- Stiggins, R. J. (1994), *Student-centred classroom assessment*, New York: Macmillan  
Publishing Company.
- Stiggins, R. and Chappuis, J. (2005), *Using student-involved classroom  
assessment to close achievement gaps*, *Theory Into Practice* 44 (1): pp. 11–18.
- Taggart, G. L., Phifer, S. J., Nixon, J. A. & Wood, M. (2001), *Rubrics: Handbook for  
construction and use*, Lancaster, PA: Technomic Publishing Co.
- Tallent-Runnels, M. K., Thomas, J. A., Lan, W. Y., Cooper, S., Ahern, T. C.,  
Shaw, S. M. and Liu, X. (2006), *Teaching courses online: A review of  
the research*, *Review of Educational Research* 76 (1): pp. 93–135.
- Taylor, J. (1990), *Analysing novices analysing Prolog: What stories do novices tell  
about Prolog*, *Instructional Science*, 19, pp. 283-309.
- Tomek, I. (1983), *Josef the Robot*, *Computers and Education*, 6(3), pp. 287-293.
- Urduan, T. & Turner, J. C. (2005), *Competence motivation in the classroom*, In A. E.  
Elliot and C. Dweck (Eds.), *Handbook of Competence Motivation*, pp. 297-  
317, New York: Guilford.
- Vosniadou, S. (2001b), *EMILE in Greece*, Abstracts of the 9th Conference of the

European Association for Research on Learning and Instruction, Fribourg,  
Switzerland.

Wilson, H. W. (2004), *Continuous assessment: Guaranteed learning?* ,  
Distance Education Report 8 (12).

Worsnop, Chris. (1997), *Assessing Media Work*, Mississauga, Canada: Wright  
Communications.

### **Ελληνική Βιβλιογραφία**

Γεωργιάδης, Π. (2006), *Η διδασκαλία του προγραμματισμού στο Γυμνάσιο-Χρήση της  
JavaScript*, Ηράκλειο.

Γρηγοριάδου, Μ. (επιμέλεια), (2003), *Μελέτη για τη Διδασκαλία της Πληροφορικής  
στη Δευτεροβάθμια Εκπαίδευση*, Προτάσεις Στρατηγικής, Εισήγηση Νο 7,  
Αθήνα: ΥΠΕΠΘ.

Δαπόντες, Ν. (2008), *Προγραμματίζοντας στο SCRATCH*, Ιστολόγιο, Διαθέσιμο στο:  
<http://makolas.blogspot.com/>, τελευταία ενημέρωση 10/6/2009.

ΔΕΠΠΣ, (2001), *Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών  
Πληροφορικής*, ΦΕΚ 1373, τ. Α' & Β', Αθήνα: Παιδαγωγικό Ινστιτούτο,  
Ανάκτηση στις 12 Μαρτίου 2009 από το [www.pi-schools.gr](http://www.pi-schools.gr)

ΔΕΠΠΣ, (2003), *Διαθεματικό Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής*,  
Αθήνα: Παιδαγωγικό Ινστιτούτο, Ανάκτηση στις 18 Μαρτίου 2009 από το  
[www.pi-schools.gr](http://www.pi-schools.gr)

- Δημητρόπουλος, Ε. (1998), *Θεωρία αξιολόγησης μαθητών*, Αθήνα: Εκδόσεις Γρηγόρη.
- Ε.Ε.Θ., (2002), *Επιθεώρηση Εκπαιδευτικών Θεμάτων*, τ. 7, Αθήνα: Παιδαγωγικό Ινστιτούτο, Ανάκτηση στις 11 Απριλίου 2009 από το [www.pi-schools.gr](http://www.pi-schools.gr)
- ΕΠΠΣ, (1997), *Ενιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής*, Αθήνα: Παιδαγωγικό Ινστιτούτο, Ανάκτηση στις 5 Απριλίου 2009 από το [www.pi-schools.gr](http://www.pi-schools.gr)
- Κασσωτάκης, Μ. & Φλουρής, Γ. (1986), *Μάθηση και Διδασκαλία, Παρουσιάσεις των τελευταίων απόψεων για τη μάθηση και τη μεθοδολογία διδασκαλίας*, Αθήνα.
- Κασσωτάκης, Μ. (1998), *Η αξιολόγηση της επιδόσεως των μαθητών*, Αθήνα: Εκδόσεις Γρηγόρη.
- Κόμης, Β. & Μικρόπουλος, Α. (2001), *Πληροφορική στην Εκπαίδευση*, Πάτρα: Εκδόσεις Ελληνικό Ανοικτό Πανεπιστήμιο.
- Κόμης, Β. (2001), *Μελέτη βασικών εννοιών του προγραμματισμού στο πλαίσιο μιας οικοδομητικής διδακτικής προσέγγισης*, ΘΕΜΑΤΑ στην Εκπαίδευση, 2(2-3), σελ. 243-270.
- Κόμης, Β. (2001), *Διδακτική της Πληροφορικής*, Πάτρα: Εκδόσεις Ελληνικό Ανοικτό Πανεπιστήμιο.
- Κόμης, Β. (2004), *Εισαγωγή στις εκπαιδευτικές εφαρμογές των Τεχνολογιών της Πληροφορίας και των Επικοινωνιών*, Αθήνα: Εκδόσεις Νέων Τεχνολογιών.
- Κόμης, Β. (2005), *Εισαγωγή στη Διδακτική της Πληροφορικής*, Αθήνα: Κλειδάριθμος.
- Κόμης, Β. (2006), *Παιδαγωγικές Δραστηριότητες με (και για) Υπολογιστές στην Προσχολική και την Πρώτη Σχολική Ηλικία*, Πανεπιστημιακές Παραδόσεις, 2η

Έκδοση, Πάτρα.

Κόμης, Β. (2006), *Εισαγωγή στη Διδακτική της Πληροφορικής*, Αθήνα: Εκδόσεις Κλειδάριθμος.

Μακράκης, Β. (2000), *Υπερμέσα στην Εκπαίδευση, Μια Κοινωνιο-Επικοινωνιακή Προσέγγιση*, Αθήνα: Μεταίχμιο.

Μακράκης, Β. & Κοντογιαννοπούλου, Γ. (1995), “*Υπολογιστές στην εκπαίδευση: μια κριτική επισκόπηση στο διεθνή χώρο και στην Ελλάδα*”, Αθήνα: Εθνικό Ίδρυμα Ερευνών.

Ματσαγγούρας, Η. (2001), *Στρατηγικές Διδασκαλίας*, Η κριτική σκέψη στη διδακτική πράξη, Πέμπτη Έκδοση, Εκδόσεις Gutenberg.

Μιχαηλίδης, Π.Γ. (2001), *Εκπαίδευση του Δασκάλου της Πληροφορικής στο Δημοτικό*, 5ο Πανελλήνιο Συνέδριο με Διεθνή Συμμετοχή “*Διδακτική των Μαθηματικών και Πληροφορική στην Εκπαίδευση*”, Πανεπιστήμιο Θεσσαλονίκης, Θεσσαλονίκη.

Πανταζής, Γ. (2007), *Χαρακτηριστικά και πρακτική εφαρμογή της διαδικασίας αξιολόγησης της επίδοσης (performance-based assessment) με τη κατασκευή μιας ρούμπρικας (rubric) σχεδιασμένης για να εκτιμήσει το βαθμό συνεργατικότητας των μαθητών στην Πρωτοβάθμια Εκπαίδευση*, Πανεπιστήμιο Ιωαννίνων. Ανάκτηση στις 28 Μαρτίου 2009 από το:  
<http://conf2007.edu.uoi.gr/>

Ράπτης, Α. & Ράπτη, Α. (1997), *Πληροφορική και Εκπαίδευση*, Συνολική προσέγγιση, Αθήνα: Έκδοση συγγραφέων.

Ράπτης, Α. & Ράπτη, Α. (2001), *Διδασκαλία και Μάθηση στην Εποχή της Πληροφορίας*, Συνολική προσέγγιση, Αθήνα: Έκδοση συγγραφέων.



- Ράπτης, Α. & Ράπτη, Α. (2002), *Μάθηση και διδασκαλία στην Κοινωνία της Πληροφορίας*, Ολική Προσέγγιση, Αθήνα: Έκδοση συγγραφέων.
- Σκορδαλάκης, Ε. (2007), *Λογισμική Μηχανική (Software Engineering)*, Αθήνα (Συμμετρία).
- Τζιμογιάννης, Α. (2002), *Προετοιμασία του Σχολείου της Κοινωνίας της Πληροφορίας, Προς ένα Ολοκληρωμένο Μοντέλο Ένταξης των Τεχνολογιών της Πληροφορίας και της Επικοινωνίας στο Ελληνικό Εκπαιδευτικό Σύστημα*, Σύγχρονη Εκπαίδευση, 122, σελ. 55-65.
- Τζιμογιάννης, Α. (2003), *Η διδασκαλία του προγραμματισμού στη Δευτεροβάθμια Εκπαίδευση*, Πρακτικά 3ου Πανελληνίου Συνεδρίου, Τόμος Α', σελ. 229-238, Ρόδος.
- ΥΠΕΠΘ, (1998), *Εκπαίδευση 2000: Για μια Παιδεία Ανοικτών Οριζώντων*, ΥΠΕΠΘ, Γ' Έκδοση, Αθήνα.
- Φεσάκης, Γ., Δημητρακοπούλου, Α. (2007), «*Επισκόπηση του χώρου των εκπαιδευτικών περιβαλλόντων προγραμματισμού Η/Υ: Τεχνολογικές και Παιδαγωγικές προβολές*», στο ΘΕΜΑΤΑ στην Εκπαίδευση - Ειδικό αφιέρωμα: Σύγχρονη έρευνα στη Διδακτική της Πληροφορικής: ερευνητικοί άξονες, μέθοδοι, τεχνικές, εργαλεία. Επιμέλεια: Κόμης Β., Πολίτης Π. και Τζιμογιάννης Α.
- Φεσάκης, Γ., Καφούση, Σ. & Σκουμπουρδή, Χ. (2008), *Δημιουργώντας Στοχαστικές Εμπειρίες για την Εξέλιξη των Διαισθητικών Αντιλήψεων Νηπίων με τη Βοήθεια Διαδικτυακών Μικρόκοσμων*, 6ο Συνέδριο ΕΤΠΕ, Λεμεσός, Κύπρος, σελ. 281-287.

- Φεσάκης, Γ., Δημητρακοπούλου, Α., Σεραφείμ, Κ., Ζαφειροπούλου, Α., Ντούνη, Μ., Τούκα, Β. (2008), *Γνωριμία με το εκπαιδευτικό περιβάλλον προγραμματισμού SCRATCH*, 4ο Συνέδριο Διδακτική Πληροφορικής, Πάτρα, Ελλάδα, σελ. 615-617.
- Χατζηλάκος, Α. (2000), *Η Τεχνολογία Οδηγός ή Υπηρέτης του Εκπαιδευτικού Συστήματος: Η Διαλεκτική του e-learning στο Σχολείο*, στο Β. Κόμης (επιμέλεια), 2ο Πανελλήνιο Συνέδριο «Οι Τεχνολογίες της Πληροφορικής και των Επικοινωνιών στην Εκπαίδευση», Πανεπιστήμιο Πατρών, Πάτρα, Οκτώβριος 2000.
- Χατζηλάκος, Α. (2002), Πρόλογος στο Κυνηγός, Π. & Δημαράκη, Ε. *Νοητικά εργαλεία και πληροφοριακά μέσα Παιδαγωγική αξιοποίηση σύγχρονης τεχνολογίας για τη μετεξέλιξη της εκπαιδευτικής πρακτικής*, Αθήνα: Εκδόσεις Καστανιώτη.

## **Πηγές στο Διαδίκτυο**

<http://scratch.mit.edu/>, τελευταία ενημέρωση 01-07-09

<http://www.middleweb.com/rubricsHG.html>, τελευταία ενημέρωση 16-05-09

<http://webquest.sdsu.edu/rubrics/weblessons.htm>, τελευταία ενημέρωση 17-04-09

<http://www.rubrics4teachers.com/>, τελευταία ενημέρωση 10-06-09