Υπολογισιμότητα και Μηχανές Turing.
Μια Διδακτική Πρόταση

Α. Σκαρλάτος[1], Ε. Ρόμπολα[2]

[1]*Πληροφορικός, Μεταπτυχιακός στο τμήμα Πληροφορικής & Τηλεπικοινωνιών ΕΚΠΑ,*
[2]*MSc Πληροφορικής, 5ο ΓΕΛ Βύρωνα*
[1]*antonisskarlatosj@gmail.com,* [2]*eleni.rompola@gmail.com*

Περίληψη

Η συγκεκριμένη εισήγηση καταθέτει μια διδακτική πρόταση, την οποία εντάσσουμε στην γενικότερη πληροφορική παιδεία για μαθητές Λυκείου, με στόχο την παρουσίαση της έννοιας της Υπολογισιμότητας. Οι μαθητές μέσα από δραστηριότητες κατευθυνόμενης διερεύνησης, οι περισσότερες από τις οποίες έχουν τα χαρακτηριστικά παιχνιδιών και γίνονται χωρίς χρήση Η/Υ, καλούνται να απαντήσουν στο ερώτημα "Υπάρχουν όρια στο τι μπορεί να υπολογιστεί;". Η απάντηση δίνεται μέσω του μαθηματικού μοντέλου των Μηχανών Turing, με την βοήθεια του οποίου ορίζεται και η έννοια της Υπολογισιμότητας. Η διδασκαλία ολοκληρώνεται με τον προγραμματισμό σε Python της Καθολικής Μηχανής Turing, η οποία μπορεί να προσομοιώνει κάθε άλλη Μηχανή Turing. Η Καθολική Μηχανή αποτελεί την μέγιστη ίσως συνεισφορά του Α. Turing στην Επιστήμη των Υπολογιστών.

Λέξεις κλειδιά: Επιστήμη των Υπολογιστών, Alan Turing, Υπολογισιμότητα, Μηχανή Turing, Καθολική Μηχανή Turing, Python.

1. Εισαγωγή

Το 1936 ο Alan Turing περιέγραψε τον ευρύτερο δυνατό υπολογιστή γενικού σκοπού, την Καθολική Μηχανή Turing, η οποία είναι ικανή να υπολογίσει ό,τι είναι δυνατό να υπολογιστεί αλγοριθμικά. Σήμερα, η συντριπτική πλειοψηφία όσων χρησιμοποιούν υπολογιστικές συσκευές και προφανώς οι μαθητές, θεωρούν πως όσο ισχυρότερα χαρακτηριστικά έχει ένας υπολογιστής τόσο περισσότερα ή απαιτητικότερα προγράμματα μπορεί να τρέξει. Από την προσεκτική επιλογή επεξεργαστή, μνήμης και κάρτας γραφικών κατάλληλων για gaming, μέχρι τις αποσπασματικές γνώσεις για τους κβαντικούς υπολογιστές, η γενικότερη αντίληψη που εκφράζεται είναι πως με την πρόοδο της τεχνολογίας διευρύνεται το σύνολο όσων μπορούν να υπολογιστούν, αυξάνεται το σύνολο των υπολογίσιμων, θα λέγαμε, προβλημάτων.

Η παρούσα εργασία καταθέτει μια διδακτική πρόταση, η οποία φιλοδοξεί να φέρει τους μαθητές αντιμέτωπους με το ερώτημα *"Υπάρχουν όρια στο τι μπορεί να υπολογιστεί;"* και να ορίσει την έννοια της Υπολογισιμότητας μέσα από τις Μηχανές Turing (στο εξής ΜΤ). Επιλέξαμε το θέμα αυτό, αν και ανήκει στην Θεωρητική

ISBN:978-960-578-072-2

Πληροφορική, διότι πιστεύουμε ότι προσφέρει στους μαθητές μια σαφή διάκριση μεταξύ υλικού και λογισμικού, αναιρεί την παρανόηση ότι κάποτε στο μέλλον οι υπολογιστές θα μπορούν να κάνουν τα πάντα, συνεισφέρει στην γενικότερη πληροφορική παιδεία, καθώς αποκαλύπτει *πως ξεκίνησαν όλα*, απομυθοποιεί την ικανότητα των υπολογιστών να *κάνουν πράγματα με μαγικό τρόπο* και δείχνει πως τα σύνθετα προγράμματα χτίζονται από απλούστερα δομικά στοιχεία, σταδιακά και με υπομονή.

2. Θεωρητικό Υπόβαθρο

Μια άτυπη περιγραφή της ΜΤ (Μηχανή Τούρινγκ - Βικιπαίδεια, 2020) περιλαμβάνει:

(α) Μια άπειρη ταινία που χωρίζεται σε κελιά, το ένα δίπλα στο άλλο. Κάθε κελί περιέχει ένα σύμβολο από ένα πεπερασμένο αλφάβητο.

(β) Μια κεφαλή που μπορεί να διαβάζει και να γράφει σύμβολα πάνω στην ταινία. Η κεφαλή μετακινείται κάθε φορά κατά ένα και μόνο κελί, είτε προς τα δεξιά είτε προς τα αριστερά.

(γ) Ένα πεπερασμένο σύνολο καταστάσεων, το οποίο περιλαμβάνει μεταξύ των άλλων την αρχική κατάσταση και τις τελικές καταστάσεις (απόρριψης, αποδοχής).

(δ) Ένα σύνολο κανόνων μετάβασης μεταξύ των καταστάσεων. Κάθε μετάβαση εξαρτάται από την τρέχουσα κατάσταση της ΜΤ και το σύμβολο που υπάρχει στην ταινία στην τρέχουσα θέση της κεφαλής.

Σε κάθε βήμα, η ΜΤ επιτελεί τρεις ενέργειες:

(α) διαβάζει την τρέχουσα θέση της ταινίας και αποφασίζει σε ποια νέα κατάσταση θα μεταβεί,

(β) γράφει στην τρέχουσα θέση της ταινίας το ίδιο ή ένα νέο σύμβολο,

(γ) μετακινεί την κεφαλή κατά μία θέση δεξιά ή αριστερά.

Οι ενέργειες αυτές καθορίζονται από τους κανόνες μετάβασης και ουσιαστικά υλοποιούν την επιθυμητή λειτουργικότητα.

Ακολουθούν οι τυπικοί ορισμοί από την σχετική βιβλιογραφία για το τι είναι Πρόβλημα, Μηχανή Turing (Hopcroft, Motwani, & Ullman, 2006) και Υπολογισιμότητα (Sipser, 2006), (Lewis, & Papadimitriou, 1998).

2.1 Πρόβλημα - Γλώσσα

Έστω ένα σύνολο στοιχείων Σ. Ένα πρόβλημα ή μια γλώσσα L ορίζεται ως ένα υποσύνολο του Σ. Δηλαδή το L περιέχει κάποια από τα στοιχεία του Σ. Για

ISBN:978-960-578-072-2

παράδειγμα το Σ μπορεί να περιέχει όλους τους φυσικούς αριθμούς. Επομένως ένα πρόβλημα L είναι ένα υποσύνολο των φυσικών αριθμών.

2.2 Μηχανή Turing

Μία μηχανή Turing (MT) ορίζεται ως ένα σύνολο από επτά στοιχεία (Q, Σ, Γ, δ, q_0, $q_{αποδοχής}$, $q_{απόρριψης}$) για τα οποία ισχύει ότι:

- Το Q είναι ένα πεπερασμένο σύνολο καταστάσεων. Το Q περιέχει τις καταστάσεις q_0, $q_{αποδοχής}$, $q_{απόρριψης}$.
- Το Σ είναι ένα πεπερασμένο σύνολο χαρακτήρων που επιτρέπονται στην είσοδο. Στο Σ δεν υπάρχει ο χαρακτήρας που υποδηλώνει το κενό.
- Το Γ είναι ένα πεπερασμένο σύνολο χαρακτήρων που επιτρέπονται στην ταινία. Ο χαρακτήρας που υποδηλώνει το κενό ανήκει στο Γ και επιπλέον το Σ είναι υποσύνολο του Γ.
- Η $δ : (Q \setminus \{q_{αποδοχής}, q_{απόρριψης}\}) \times Γ \rightarrow Q \times Γ \times \{Α, Δ\}$ είναι η συνάρτηση μετάβασης, όπου Α σημαίνει ότι ο δείκτης της ταινίας μετακινείται αριστερά και Δ δεξιά.
- Οι καταστάσεις q_0, $q_{αποδοχής}$, $q_{απόρριψης}$ είναι η αρχική κατάσταση, η κατάσταση αποδοχής και η κατάσταση απόρριψης αντίστοιχα. Η κατάσταση αποδοχής δεν είναι ίδια με την κατάσταση απόρριψης.

Μία Καθολική Μηχανή Turing (ΚΜΤ) U, είναι μία ΜΤ η οποία σαν είσοδο δέχεται μία κωδικοποιημένη ΜΤ Μ και μία είσοδο x και προσομοιώνει την Μ με είσοδο το x. Δηλαδή η έξοδος της U είναι ίδια με αυτήν της Μ με είσοδο το στοιχείο x.

2.3 Υπολογισιμότητα

Έστω ένα σύνολο στοιχείων Σ. Ένα πρόβλημα L, λέμε ότι είναι υπολογίσιμο αν και μόνο αν υπάρχει μία ΜΤ, δηλαδή ένας αλγόριθμος, ο οποίος με είσοδο ένα στοιχείο x του Σ, σταματάει και αποφαίνεται θετικά αν το x ανήκει στο σύνολο L και αρνητικά αν το x δεν ανήκει στο σύνολο L.

Για παράδειγμα έστω ότι το Σ είναι το σύνολο όλων των φυσικών αριθμών και L το σύνολο όλων των άρτιων αριθμών. Για να αποδείξουμε ότι το L είναι ένα υπολογίσιμο πρόβλημα, αρκεί να κατασκευάσουμε μία ΜΤ που δεχόμενη ως είσοδο έναν αριθμό x που ανήκει στο Σ σταματάει πάντα και αποφαίνεται θετικά αν ο αριθμός x είναι άρτιος και αρνητικά αν ο αριθμός x είναι περιττός.

ISBN:978-960-578-072-2

3. Η Διδακτική Πρόταση

Οι έννοιες αποτελούν δομικά στοιχεία της σκέψης, διότι μέσα από αυτές οργανώνουμε την εμπειρία μας και συνδέουμε τα νέα γνωστικά στοιχεία με τη γνώση που κατέχουμε (Ματσαγγούρας, 2007). Η διδασκαλία μιας νέας έννοιας θα μπορούσε να γίνει μέσω του ορισμού της και κατάλληλων παραδειγμάτων. Η στρατηγική αυτή, ωστόσο, δεν ενεργοποιεί τον μαθητή και οδηγεί σε επιφανειακή προσέγγιση της νέας έννοιας. Πιο αποτελεσματικές, σύμφωνα και με την βιβλιογραφία, έχουν αποδειχθεί οι στρατηγικές κατευθυνόμενης διερεύνησης, όπου ο εκπαιδευτικός επιλέγει το υλικό που θα δώσει στους μαθητές να επεξεργαστούν και τους προβληματισμούς που θα θέσει. Δεν καθοδηγεί τη σκέψη των μαθητών, αλλά μέσα από διάλογο και διαδικασίες όπως η παρατήρηση, η σύγκριση, η κατηγοριοποίηση, κλπ., οι μαθητές καταλήγουν στα συμπεράσματα και στις γενικεύσεις στις οποίες επιθυμεί ο εκπαιδευτικός να τους οδηγήσει. Η συνεχής αλληλεπίδραση των μαθητών με το εκπαιδευτικό περιβάλλον που έχει προετοιμάσει ο εκπαιδευτικός και με το εκπαιδευτικό υλικό που τους παρέχει, βοηθούν τους μαθητές να χτίσουν σταδιακά τη νέα γνώση και διασαφηνίζει την καινούργια έννοια.

Η διδακτική πρόταση που παρουσιάζουμε ακολουθεί την στρατηγική της Επαγωγικό-Υποθετικής Διδασκαλίας, όπως περιγράφεται στη συνέχεια, με προτεινόμενη διάρκεια τις 3 διδακτικές ώρες.

Ως κατάλληλο μαθητικό κοινό προτείνουμε τις τρεις τάξεις του Γενικού Λυκείου. Δεν απαιτούνται πρότερες τεχνικές γνώσεις. Η στοιχειώδης γνώση της γλώσσας προγραμματισμού Python και η δυνατότητα διεξαγωγής του μαθήματος σε εργαστήριο Πληροφορικής, προσθέτουν στις τελευταίες φάσεις της διδασκαλίας μια ενδιαφέρουσα πτυχή: την προσομοίωση της Καθολικής ΜΤ σε Python.

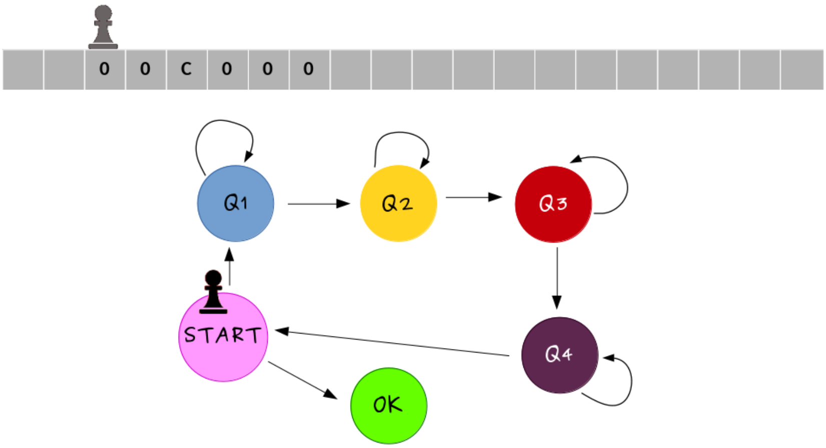
3.1 Ψυχολογική και Γνωσιολογική Προετοιμασία

Υπάρχουν όρια στο τι μπορεί να υπολογιστεί; Ο κύριος, αυτός, προβληματισμός παρακινεί τους μαθητές να ανατρέξουν σε όσα έχουν διαβάσει ή ακούσει για υπερυπολογιστές ή κβαντικούς υπολογιστές. Όσοι χρησιμοποιούν υπολογιστές υψηλών επιδόσεων, πχ για gaming, έχουν επίσης πολύ συγκεκριμένη άποψη για το τι σημαίνει ισχυρός υπολογιστής.
Ως γνωσιολογική προετοιμασία, μια ιστορική αναδρομή γύρω από τον Alan Turing, τον 2ο παγκόσμιο πόλεμο, την αποκρυπτογράφηση του Enigma, κλπ, (Hodges, 2004) είναι χρήσιμη ώστε οι μαθητές να μεταφερθούν νοητικά σε μια εποχή όπου *υπολογιστής* είναι απλώς ένας άνθρωπος που μπορεί να εκτελεί μαθηματικούς υπολογισμούς.

ISBN:978-960-578-072-2

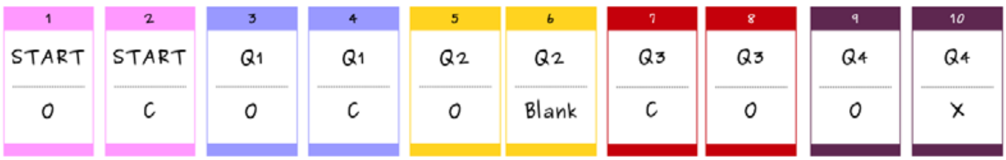
3.2 Επαγωγικό-Υποθετική Επεξεργασία

Στην αρχή της δεύτερης φάσης, ο εκπαιδευτικός παρουσιάζει παραδείγματα. Επειδή η λογική των ΜΤ είναι πρωτόγνωρη, έχουμε κατασκευάσει τις εξής ΜΤ ως επιτραπέζια παιχνίδια: (α) ΜΤ που δέχεται στην είσοδο δύο αριθμούς κωδικοποιημένους ως σειρές μηδενικών και τους προσθέτει, (β) ΜΤ που δέχεται στην είσοδο δύο αριθμούς κωδικοποιημένους ως σειρές μηδενικών και τους αποδέχεται εφόσον είναι ίσοι. Τα παιχνίδια αυτά αποτελούνται από την Άπειρη Ταινία, το Διάγραμμα Καταστάσεων, τις Κάρτες Μεταβάσεων, ένα πιόνι που τοποθετείται στην τρέχουσα κατάσταση κι ένα δεύτερο πιόνι που κινείται πάνω στην Άπειρη Ταινία.



Εικόνα 1. *ΜΤ που αθροίζει δύο αριθμούς κωδικοποιημένους ως σειρές μηδενικών: Καταστάσεις και Άπειρη Ταινία*

Οι μαθητές παίζουν τα παιχνίδια, ακολουθώντας τις οδηγίες που υπάρχουν στις Κάρτες Μετάβασης. Οι Κάρτες Μετάβασης αναγράφουν στην μια πλευρά την τρέχουσα κατάσταση (εκεί βρίσκεται το 1ο πιόνι – στην εικόνα 1 μαύρο) και την τιμή της ταινίας στην τρέχουσα θέση (2ο πιόνι – στην εικόνα 1 γκρίζο).



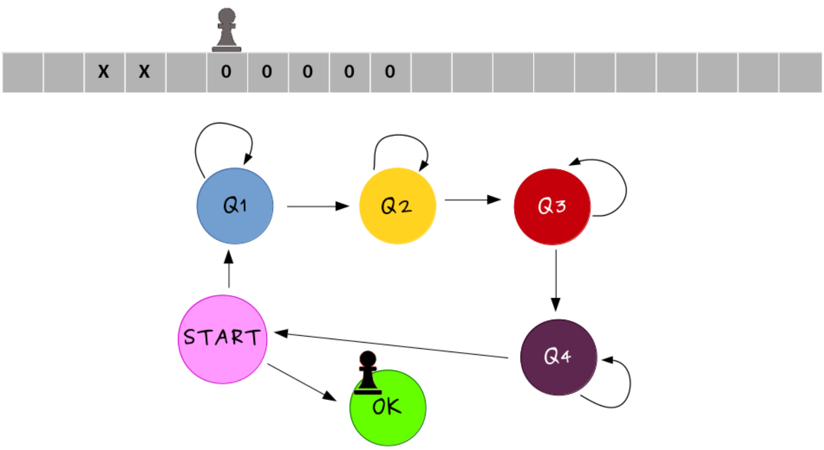
Εικόνα 2. *Κάρτες Μετάβασης – εμπρός πλευρά*

Με βάση αυτά τα δύο στοιχεία οι μαθητές διαλέγουν την αντίστοιχη κάρτα και ακολουθούν τις οδηγίες που αναγράφονται στο πίσω μέρος της: νέα κατάσταση για το 1ο πιόνι (μαύρο), νέα τιμή της τρέχουσας θέσης της ταινίας, νέα θέση για το 2ο πιόνι (γκρίζο).

ISBN:978-960-578-072-2

Εικόνα 3. *Κάρτες Μετάβασης – πίσω πλευρά*

Η λογική σύμφωνα με την οποία η παραπάνω ΜΤ υλοποιεί την πρόσθεση δύο αριθμών είναι η ακόλουθη: Οι δύο αριθμοί κωδικοποιούνται με σειρές μηδενικών και διαχωρίζονται από τον χαρακτήρα C. Η κεφαλή αντικαθιστά τα μηδενικά του πρώτου αριθμού με τον χαρακτήρα X (ως ένδειξη ότι τα επεξεργάστηκε) και γράφει ισάριθμα μηδενικά στα κελιά δεξιά του δεύτερου αριθμού. Προφανώς θα μπορούσε να είχε σχεδιαστεί διαφορετική λογική τόσο για την κωδικοποίηση των αριθμών όσο και για την πρόσθεσή τους. Έγκειται στην ευχέρεια του εκπαιδευτικού να σχεδιάσει την ΜΤ με όποιον τρόπο επιθυμεί, λαμβάνοντας υπ’ όψιν ότι επιλύει ένα πρόβλημα υπολογισμού.



Εικόνα 4. *Η τελική κατάσταση της ΜΤ που αθροίζει δύο αριθμούς*

Παρομοίως και για το παιχνίδι της σύγκρισης δύο αριθμών, ο εκπαιδευτικός μπορεί μέσω των Καρτών Μετάβασης να σχεδιάσει τον τρόπο λειτουργίας της ΜΤ, λαμβάνοντας υπ’ όψιν ότι επιλύει ένα πρόβλημα απόφασης. Στα δύο αυτά παιχνίδια, επιδίωξή μας ήταν Μηχανές Turing μετρίου μεγέθους και όχι πολύ απλές, ώστε οι μαθητές να έχουν χρόνο να κατανοήσουν τους κανόνες των παιχνιδιών.

Γενικά, η κατάληξη κάθε παιχνιδιού μπορεί να είναι είτε η Κατάσταση Αποδοχής είτε η Κατάσταση Απόρριψης. Το αν το ίδιο παιχνίδι (δηλ. η ίδια ΜΤ) θα καταλήξει σε κατάσταση απόρριψης ή αποδοχής, εξαρτάται από τα δεδομένα εισόδου τα οποία

ISBN:978-960-578-072-2

γράφονται αρχικά στην Άπειρη Ταινία. Ο εκπαιδευτικός παροτρύνει τους μαθητές να ξαναπαίξουν τα ίδια παιχνίδια αλλάζοντας τα αρχικά δεδομένα.

3.3 Ορισμός Έννοιας

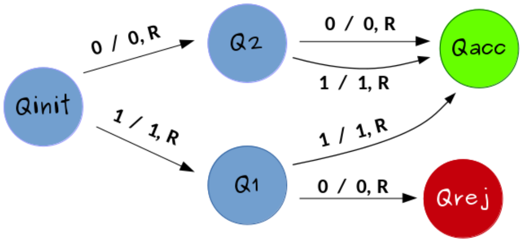
Βασιζόμενοι στην παραπάνω επεξεργασία, οι μαθητές μπορούν πλέον να ορίσουν περιγραφικά την έννοια της Υπολογισιμότητας. Ένα πρόβλημα είναι υπολογίσιμο όταν υπάρχει μια ΜΤ η οποία, αφού λειτουργήσει με προκαθορισμένο τρόπο, να μπορεί να αποφανθεί σωστά για τα στοιχεία του προβλήματος. Δηλαδή να αποφανθεί θετικά για τα στοιχεία που ανήκουν στο πρόβλημα και αρνητικά για αυτά που δεν ανήκουν.

Μέσα από συζήτηση γίνεται φανερό πως για κάθε αλγόριθμο υπάρχει μια αντίστοιχη ΜΤ. Δεν είναι η ίδια μηχανή που τρέχει όλους τους αλγορίθμους, είναι διαφορετικές μηχανές, μια για κάθε αλγόριθμο, που όλες όμως διαθέτουν το ίδιο απλό σύνολο χαρακτηριστικών: άπειρη ταινία, διάγραμμα καταστάσεων και κανόνες μετάβασης. Οι μαθητές θα πρέπει να είναι σε θέση να διακρίνουν πως όλες οι ΜΤ διαθέτουν την ίδια άπειρη ταινία, της οποίας το περιεχόμενο μπορούν να ελέγξουν (διαβάσουν) και να τροποποιήσουν, αλλά αυτό που διαφοροποιεί τις ΜΤ μεταξύ τους είναι το διάγραμμα καταστάσεων και οι κανόνες μετάβασης. Καθώς, λοιπόν, η ιδέα αυτών των παράξενων μηχανών εξελίχθηκε στη διάρκεια δεκαετιών σε αυτό που σήμερα γνωρίζουμε ως Η/Υ, η στοιχειώδης αντιστοίχιση που υπάρχει είναι: (α) η Άπειρη Ταινία αντιστοιχεί στην μνήμη ενός Η/Υ (υλικό), (β) το Διάγραμμα Καταστάσεων και οι - άμεσα σχετιζόμενοι με αυτό - Κανόνες Μετάβασης, αντιστοιχούν στο λογισμικό.

Ο A. Turing περιέγραψε την Καθολική ΜΤ (στο εξής ΚΜΤ), μια μηχανή η οποία μπορεί να προσομοιώσει άλλες μηχανές. Να λάβει δηλ. ως είσοδο την περιγραφή μιας οποιασδήποτε άλλης ΜΤ Μ μαζί με ένα στοιχείο x και να προσομοιώσει την Μ με είσοδο το x. Το αποτέλεσμα της ΚΜΤ θα είναι ίδιο με αυτό που προκύπτει όταν η Μ τρέχει αυτόνομα με είσοδο το x. Η ιδέα της ΚΜΤ ενέπνευσε αργότερα τον von Neumann να δημιουργήσει την αρχιτεκτονική von Neumann. Οι δύο πρωτοπόρες αυτές ιδέες οδήγησαν, μέσω της συνεισφοράς πολλών ακόμη ανθρώπων και κατά τη διάρκεια πολλών χρόνων, στον σημερινό Η/Υ, ο οποίος μέσω του λειτουργικού συστήματος τρέχει άλλα προγράμματα. Μια ιστορική αναδρομή της εξέλιξης αυτής παρουσιάζει ενδιαφέρον.

3.4 Εφαρμογές

Στη φάση των Εφαρμογών ο εκπαιδευτικός ζητάει από τους μαθητές να ελέγξουν αν τα ακόλουθα προβλήματα είναι ή όχι υπολογίσιμα, φτιάχνοντας αντίστοιχες ΜΤ που να τα επεξεργάζονται: (α) αντιστροφή ενός bit, (β) μη-αντιστροφή ενός bit, (γ) έλεγχος αν σε μια σειρά 2 bits το πρώτο είναι μεγαλύτερο ή ίσο του δεύτερου. Οι μαθητές θα πρέπει να σχεδιάσουν τις καταστάσεις των ΜΤ και να καταγράψουν τους κανόνες μετάβασης μεταξύ των καταστάσεων, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 5. *MT που συγκρίνει 2 bit*

Επόμενο στάδιο είναι η σύνθεση των απλών ΜΤ που μόλις κατασκεύασαν, ώστε να προκύψει μια πιο σύνθετη μηχανή, η οποία θα υπολογίζει το πρόβλημα "Σε μια σειρά 3 bits, να γίνει αντιστροφή του 3ου bit αν και μόνο αν το πρώτο bit είναι μικρότερο ή ίσο του δεύτερου, διαφορετικά το 3ο bit να παραμείνει ως έχει.".

Άλλες ΜΤ που θα μπορούσαν οι μαθητές να υλοποιήσουν σε αυτή τη φάση, ως δραστηριότητες εμπέδωσης της νέας γνώσης:

(α) ΜΤ που δέχεται σαν είσοδο μία δυαδική συμβολοσειρά και την αποδέχεται αν και μόνο αν οι μονάδες είναι περισσότερες των μηδενικών. (Σενάριο παρακίνησης για τους μαθητές: Η δυαδική συμβολοσειρά είναι το μήνυμα που έχουμε υποκλέψει. Ως άγγλοι κρυπτογράφοι έχουμε ανακαλύψει με την βοήθεια του A.Turing ότι αν οι μονάδες είναι περισσότερες των μηδενικών τότε το μήνυμα προέρχεται από τους γερμανούς ναζί, ενώ σε διαφορετική περίπτωση προέρχεται από τους σοβιετικούς).

(β) ΜΤ που αποφασίζει τη γλώσσα (ή το πρόβλημα) L = {w#w | w ανήκει στο Σ*}, όπου Σ το αλφάβητο της L και Σ* το λεξιλόγιο της L (δηλ. κάθε δυνατός συνδυασμός των στοιχείων του Σ). Η ΜΤ αποδέχεται συμβολοσειρές που αποτελούνται από δύο ίδιες λέξεις (w) οι οποίες διαχωρίζονται από τον χαρακτήρα #. Αν θεωρήσουμε ως αλφάβητο το Σ = {a, b, c}, αποδεκτές συμβολοσειρές είναι π.χ. οι abc#abc, aa#aa, bbac#bbac, ενώ μη αποδεκτά παραδείγματα: a#aa, a#b, ac#ba. (Σενάριο παρακίνησης για τους μαθητές: Κατασκευάζουμε ένα site και ζητάμε από τον χρήστη να εισάγει έναν κωδικό. Στην συνέχεια του ζητάμε να τον εισάγει ξανά για επιβεβαίωση. Επομένως θα θέλαμε έναν αλγόριθμο για να ελέγξουμε αν οι δύο κωδικοί ταυτίζονται).

Εάν το μάθημα γίνεται σε σχολικό εργαστήριο πληροφορικής και εφόσον οι μαθητές γνωρίζουν προγραμματισμό σε Python, μπορούν να προσομοιώσουν μια ΜΤ σε Python. Η λογική των ΜΤ βασίζεται σε πολλές δομές επιλογής (if) .

ISBN:978-960-578-072-2

```
1.  head = 0
2.  state = 'qinit'

3.  tape = list(raw_input())
4.  tape.append('_')

5.  while state != 'qacc' and state != 'qrej':
6.     print tape, head, state

7.     if state == 'qinit' and tape[head] == '0':
8.         state = 'q2'
9.         tape[head] = '0'
10.        head += 1
11.        continue

12.    if state == 'qinit' and tape[head] == '1':
13.        state = 'q1'
14.        tape[head] = '1'
15.        head += 1
16.        continue

17.    if state == 'q1' and tape[head] == '0':
18.        state = 'qrej'
19.        tape[head] = '0'
20.        head += 1
21.        continue

22.    if state == 'q1' and tape[head] == '1':
23.        state = 'qacc'
24.        tape[head] = '1'
25.        head += 1
26.        continue

27.    if state == 'q2' and tape[head] == '0':
28.        state = 'qacc'
29.        tape[head] = '0'
30.        head += 1
31.        continue

32.    if state == 'q2' and tape[head] == '1':
33.        state = 'qacc'
34.        tape[head] = '1'
35.        head += 1
36.        continue

37. print tape, head, state
```

Εικόνα 6. *Παράδειγμα προσομοίωσης ΜΤ σε Python*

Η παραπάνω ΜΤ συγκρίνει δύο bit και τα αποδέχεται εφόσον το 1ο είναι μικρότερο ή ίσο του 2ου (σχηματική αναπαράσταση στην Εικόνα 5). Η μεταβλητή head (γρ.1) είναι η κεφαλή της ΜΤ που κινείται πάνω στην άπειρη ταινία. Η άπειρη ταινία προσομοιώνεται με την λίστα tape στην οποία αρχικά γράφεται η είσοδος του χρήστη (γρ.3-4). Η τρέχουσα κατάσταση της ΜΤ εκφράζεται από την μεταβλητή state (γρ.2). Η λογική της ΜΤ υλοποιείται επαναληπτικά (γρ.5) όσο η μηχανή δεν έχει φτάσει σε κατάσταση αποδοχής (‘qacc’) ή απόρριψης (‘qrej’) της εισόδου. Σε κάθε επανάληψη εκτελείται μια από τις έξι δομές if (γρ. 7, 12, 17, 22, 27, 32) οι οποίες ανάλογα με την τρέχουσα κατάσταση της μηχανής (state) και την τιμή της ταινίας στη θέση όπου βρίσκεται η κεφαλή (tape[head]) εκτελούν τρεις ενέργειες: (α) μεταβαίνουν σε νέα κατάσταση (γρ.8, 13, 18, 23, 28, 33), (β) γράφουν στο τρέχον σημείο της ταινίας μία τιμή (γρ. 9, 14, 19, 24, 29, 34) και (γ) μετακινούν την κεφαλή κατά μια θέση πάνω στην ταινία (γρ.10, 15, 20, 25, 30, 35). Σε κάθε επανάληψη εμφανίζεται η συνολική εικόνα της ΜΤ (γρ.6, 37).

Με παρόμοιο τρόπο προσομοιώνεται σε Python κάθε ΜΤ που θα σχεδιάσουν οι μαθητές. Εάν ο εκπαιδευτικός το κρίνει πρόσφορο και βέβαια ανάλογα με τη σύνθεση της τάξης και το ενδιαφέρον των μαθητών, μπορεί να προχωρήσει στην προσομοίωση της Καθολικής Μηχανής Turing, η οποία μπορεί να προσομοιώσει η ίδια κάθε άλλη ΜΤ. Η σύλληψη αυτή του A. Turing, εκπληκτική αν αναλογιστούμε την εποχή στην οποία διατυπώθηκε και αποδείχθηκε, αποτελεί ένα ανάλογο του σημερινού Η/Υ από την πλευρά του λογισμικού.

ISBN:978-960-578-072-2

```
1.  transitions = [trans.split(',') for trans in raw_input().split()]
2.  tape = list(raw_input())
3.  tape.append('_')

4.  head = 0
5.  state = 'qinit'

6.  while state != 'qacc' and state != 'qrej':
7.      print tape, head, state
8.      found = False

9.      for trans in transitions:
            # (state, tape[head]) -> (new_state, (new) tape[head],
            #                         (new) position of tape-head)
10.         if state == trans[0] and tape[head] == trans[1]:
11.             state = trans[2]
12.             tape[head] = trans[3]
13.             head += int(trans[4])

                # tape-head must stay in the bounds
14.             if head < 0:
15.                 head = 0

                # tape is infinite and so we should add
                # extra space if it is needed
16.             if head == len(tape):
17.                 tape.append('_')

18.             found = True
19.             break

        # if no valid intruction was found then we should
        # terminate the execution
20.     if not found:
21.         break

22. print tape, head, state
```

Εικόνα 7. *Προσομοίωση Καθολικής ΜΤ σε Python*

Στον παραπάνω κώδικα, η μία από τις δύο εισόδους της ΚΜΤ είναι η περιγραφή μιας ΜΤ η οποία αποθηκεύεται στην λίστα transitions (γρ.1). Αν για παράδειγμα επιθυμούμε, η ΚΜΤ να τρέξει την ΜΤ σύγκρισης 2 bit, της οποίας η σχηματική αναπαράσταση και ο κώδικας υπάρχουν στις εικόνες 5 και 6 αντίστοιχα, τότε η είσοδος που πρέπει να δοθεί στην ΚΜΤ ως περιγραφή της παραπάνω ΜΤ είναι: qinit,0,q2,0,1 qinit,1,q1,1,1 q1,0,qrej,0,1 q1,1,qacc,1,1 q2,0,qacc,0,1 q2,1,qacc,1,1. Κάθε 5-άδα συμβόλων εκφράζουν: τρέχουσα κατάσταση της μηχανής, τιμή της ταινίας στην τρέχουσα θέση της κεφαλής, νέα κατάσταση της μηχανής, νέα τιμή της ταινίας στην τρέχουσα θέση της κεφαλής, πλήθος θέσεων μετακίνησης της κεφαλής πάνω στην ταινία. Η δεύτερη είσοδος της ΚΜΤ είναι η είσοδος που προορίζεται για την ΜΤ (γρ. 2) και την οποία λαμβάνει βέβαια η ΚΜΤ για να επιτελέσει τη λειτουργία της συγκεκριμένης μηχανής (γρ. 3-22).

Η παρουσίαση του κώδικα της ΚΜΤ μπορεί να γίνει από τον εκπαιδευτικό - ο κώδικάς της να προϋπάρχει στους Η/Υ του εργαστηρίου - και να δοθεί χρόνος στους μαθητές να τρέξουν στην ΚΜΤ τις δικές τους απλές ΜΤ τις οποίες κατασκεύασαν νωρίτερα και για τις οποίες έχουν τις σχηματικές περιγραφές.

ISBN:978-960-578-072-2

Στο τέλος των παραπάνω εφαρμογών, οι οποίες προωθούν ακόμη περισσότερο την μαθητική συμμετοχή, ο εκπαιδευτικός ευελπιστεί πως παρείχε αποτελεσματικές ευκαιρίες για την εμπέδωση και ενεργοποίηση της μάθησης.

3.5 Ανακεφαλαίωση - Αξιολόγηση

Η διδασκαλία κλείνει με ανακεφαλαίωση, σχηματική και απολογιστική, και ακολουθεί η μαθησιακή και μεταγνωστική αξιολόγηση. Ιδιαίτερη σημασία δίνει ο εκπαιδευτικός στην μεταγνωστική αξιολόγηση, ώστε οι μαθητές να συνειδητοποιήσουν τον τρόπο σκέψης τους και να τον αναλύσουν.

4. Συμπεράσματα

Στην διδακτική πρόταση που παρουσιάσαμε, οι μαθητές ήρθαν αντιμέτωποι με το ερώτημα “Υπάρχουν όρια στο τι μπορεί να υπολογιστεί;” και μέσω των Μηχανών Turing ορίστηκε η έννοια της Υπολογισιμότητας. Το κυρίως τμήμα της διδασκαλίας υλοποιείται χωρίς χρήση Η/Υ και με παιγνιώδεις δραστηριότητες, ώστε να εμπλακεί το σύνολο των μαθητών και να παραμεριστούν οι όποιες προκαταλήψεις τους σχετικά με την δυσκολία κατανόησης εννοιών Πληροφορικής.

Η Υπολογισιμότητα αποτελεί θεωρητικό κλάδο της Πληροφορικής και παρ’ όλο που δεν συσχετίζεται άμεσα με πρακτικές εφαρμογές, εντούτοις η παρουσίασή της μέσω κατάλληλου διδακτικού υλικού και διδακτικής μεθοδολογίας (α) δίνει την δυνατότητα ανασκευής παρανοήσεων (πχ οι υπολογιστές στο μέλλον θα κάνουν τα πάντα), (β) αναδεικνύει την ειδοποιό διαφορά μεταξύ υλικού και λογισμικού, (γ) απομυθοποιεί την υποτιθέμενη ικανότητα των υπολογιστών να λειτουργούν μαγικά αναδεικνύοντας τον ρόλο των Λειτουργικών Συστημάτων, και (δ) συνεισφέρει στην γενικότερη πληροφορική παιδεία των μαθητών καθώς ιστορικά συμπίπτει με τις πρώτες προσπάθειες θεμελίωσης της Επιστήμης των Υπολογιστών.

Ως διδασκαλία υλοποιήθηκε πειραματικά σε μικτή ομάδα 20 μαθητών Α και Β τάξης του 5ου ΓΕΛ Βύρωνα. Η ομάδα αυτή των μαθητών παρακολούθησε την ίδια χρονική περίοδο κι άλλες διδασκαλίες που δεν συνδέονται άμεσα με το Αναλυτικό Πρόγραμμα Σπουδών των τάξεών τους, όπως για παράδειγμα οι Αλγόριθμοι DFS και BFS, μια Διασκεδαστική Εισαγωγή στους Γράφους, Προσομοιώσεις Νευτώνειας Φυσικής με Python, κ.α. Οι πειραματικές αυτές διδασκαλίες είναι μια γενικότερη τακτική που ακολουθούμε όταν επιθυμούμε να παρατηρήσουμε τον βαθμό ανταπόκρισης των μαθητών μας σε διάφορα θέματα Πληροφορικής, πριν προχωρήσουμε, σε επόμενη σχολική χρονιά, στην διδασκαλία τους σε αμιγείς τάξεις. Επιλέγουμε θέματα τα οποία θεωρούμε ενδιαφέροντα, που δεν έχουν ακόμη ενταχθεί στο ΑΠΣ και τα οποία παρουσιάζουμε στους μαθητές μας στον χρόνο που απομένει μετά την ολοκλήρωση της διδακτέας ύλης.

Η Υπολογισιμότητα και οι Μηχανές Turing χαρακτηρίστηκαν από τους μαθητές ως ασυνήθιστη διδασκαλία και ψηφίστηκαν ως πολύ ενδιαφέρουσα.

ISBN:978-960-578-072-2

Αναφορές

Hodges, A. (2004). *Άλαν Τιούρινγκ: Το Αίνιγμα.* Αθήνα: ΤΡΑΥΛΟΣ.

Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2006). Introduction to Automata Theory, Languages, and Computation. 3rd Edition. Boston: Pearson Addison Wesley.

Lewis, H. R., & Papadimitriou, C. H. (1998). Elements of the Theory of Computation. Prentice-Hall International.

Sipser, M. (2006). Introduction to the Theory of Computation, 2nd Edition. Boston: Thomson Course Technology.

Μηχανή Τούρινγκ - Βικιπαίδεια. (2020). Ανάκτηση 7 31, 2020, από https://el.wikipedia.org/wiki/Μηχανή_Τούρινγκ

Ματσαγγούρας, Η. Γ. (2007). *Στρατηγικές Διδασκαλίας*. Αθήνα: Gutenberg.

Abstract

In this paper we submit an educational proposal, the content of which we deem a necessary part of a high school student's larger education in the field of computer science, in order to present the concept of Computation. The students, via teacher-directed investigation activities, most of which resemble easy-to-grasp, simple games and do not require use of a computer, are challenged to come up with an answer to the question: "Are there limits as to what can be computed?". The answer comes from the Turing Machine mathematical model, which also helps define the term Computation. The teaching process is completed by implementing, using Python, the Universal Turing Machine, which is able to simulate any other Turing Machine. The Universal Turing Machine constitutes what could be referred to as the greatest contribution of Alan Turing to the field of Computer Science.

Keywords: Computer Science, Alan Turing, Computation, Turing Machine, Universal Turing, Python.