

Προσέγγιση βασικών εννοιών της πληροφορικής μέσα από τον διαγωνισμό Astro Pi Mission Space Lab

Ευριπίδης Βραχνός¹, Μαρία Κατσένη²

¹ evripides@aub.gr, ² mkatseni@sch.gr

¹ Διδάσκων, Οικονομικό Πανεπιστήμιο Αθηνών ² Εκπαιδευτικός, Γυμνάσιο Θρακομακεδόνων

Περίληψη

Η μεθοδολογία STEM έχει στόχο την διδακτική προσέγγιση επιστημονικών εννοιών διαφορετικών γνωστικών αντικειμένων μέσα από πειράματα και βιωματικές δραστηριότητες. Ενώ συνήθως γίνεται αναφορά σε έννοιες των φυσικών επιστημών, σε πολλές δραστηριότητες STEM εμπλέκονται περιοχές της πληροφορικής όπως η ρομποτική, η αλγοριθμική ή η αναπαράσταση και κωδικοποίηση πληροφορίας. Στην εργασία αυτή παρουσιάζουμε τα προκαταρκτικά αποτελέσματα από την εφαρμογή μιας δραστηριότητας μέσω της οποίας προσπαθούμε να προσεγγίσουμε επιστημονικές έννοιες της πληροφορικής του προγράμματος σπουδών του Γυμνασίου. Η δραστηριότητα είναι βασισμένη στον διαγωνισμό Mission Space Lab που συνδιοργανώνουν το ίδρυμα Raspberry και ο Ευρωπαϊκός Οργανισμός Διαστήματος. Στον διαγωνισμό αυτό δίνεται η δυνατότητα σε ομάδες μαθητών να προτείνουν, να σχεδιάσουν και υλοποιήσουν το δικό τους πείραμα το οποίο θα εκτελεστεί στον Διεθνή Διαστημικό Σταθμό. Οι μαθητές έρχονται αντιμέτωποι με πραγματικά προβλήματα που πρέπει να λύσουν και τα οποία σχετίζονται με επιστημονικές έννοιες της πληροφορικής που έχουν διδαχθεί. Στο τέλος παρουσιάζονται κάποια προκαταρκτικά ποιοτικά αποτελέσματα και συμπεράσματα που προέκυψαν από την μελέτη των λύσεων που έδωσαν οι μαθητές στα προβλήματα αυτά.

Λέξεις κλειδιά: διαγωνισμός Mission Space Lab, αναπαράσταση πληροφορίας, προγραμματισμός, Python

Abstract

The STEM methodology aims to introduce scientific concepts of different subjects through experimentation and exploratory activities. While natural science concepts are commonly referred to, many STEM activities involve areas of computing such as robotics, algorithmic or information representation and coding. In this work we present the preliminary results from the implementation of an activity through which we try to approach scientific concepts of informatics of the high school curriculum. The activity is based on the Mission Space Lab competition co-organized by the Raspberry Foundation and the European Space Agency. This competition provides students the opportunity to propose, design and implement their own experiment which will be performed on the International Space Station. Students are confronted with real problems that need to be solved and which are related to scientific concepts of computer science that have been taught. Finally we present some preliminary qualitative results and conclusions that have emerged from this activity.

Key Words: Mission Space Lab, programming, Python, data representation.

Εισαγωγή

Ένας σημαντικός παράγοντας ο οποίος συμβάλλει στις δυσκολίες που αντιμετωπίζουν οι μαθητές στο σχολείο είναι η μονόπλευρη προσέγγιση των επιστημονικών εννοιών χωρίς να υπάρχει άμεση σύνδεση με τις εμπειρίες και τα ενδιαφέροντα των μαθητών. Δυο από τα πιο συνηθισμένα ερωτήματα που θέτουν οι μαθητές όταν έρχονται αντιμέτωποι με νέες έννοιες είναι : “Που θα μας χρησιμεύουν όλα αυτά;” και “γιατί τα μαθαίνουμε αυτά;”. Τα ερωτήματα αυτά δεν είναι εύκολο να απαντηθούν με τρόπο που θα πείσουν τους μαθητές για την σημασία των μαθημάτων που διδάσκονται στο σχολείο. Ένας βασικός λόγος για αυτό συνιστά ο κατακερματισμός των γνωστικών αντικειμένων. Η διδασκαλία των μαθηματικών, των φυσικών επιστημών και της πληροφορικής ως ξεχωριστά και αυτόνομα γνωστικά αντικείμενα καθιστά ιδιαίτερα δύσκολη την παρουσίαση

παραδειγμάτων και εφαρμογών τους στον πραγματικό κόσμο ώστε οι μαθητές να αποκτήσουν μια ενιαία και ολοκληρωμένη εικόνα.

Ο Dewey (1990) ίσως έχει εκφράσει το πρόβλημα της έλλειψης διεπιστημονικότητας στην εκπαίδευση καλύτερα: «Δεν έχουμε μια σειρά από χωριστούς κόσμους, ένας από τους οποίους είναι μαθηματικός, άλλος φυσικός, άλλος ιστορικός. Ζούμε σε έναν κόσμο, όπου όλες οι πλευρές συνδέονται, όλες οι σπουδές προέρχονται από σχέσεις του ενός μεγάλου κοινού κόσμου». Αυτό που λέει ο Dewey είναι άλλωστε προφανές. Η γνώση είναι ενιαία, δεν υπάρχει μαθηματική ή φυσική γνώση. Έννοιες του απειροστικού λογισμού όπως η παράγωγος και το ολοκλήρωμα περιγράφουν φυσικές έννοιες όπως η ταχύτητα και το έργο δύναμης αντίστοιχα. Η παρουσίαση αυτών των εννοιών ξεχωριστά και όχι ως μια ενιαία οντότητα είναι λογικό να προκαλέσει σύγχυση, δυσκολίες ή και παρανοήσεις στους μαθητές.

Για αυτό είναι σημαντικό να παρουσιάζονται οι επιστημονικές έννοιες στους μαθητές με διεπιστημονικό τρόπο και σε άμεση σύνδεση με πραγματικά προβλήματα και εφαρμογές ώστε τα παιδιά να κατανοήσουν την χρησιμότητά και την σημασία τους. Μια καλή στρατηγική προς αυτή την κατεύθυνση αποτελούν οι δραστηριότητες επίλυσης προβλήματος στις οποίες οι μαθητές χρειάζεται να χρησιμοποιήσουν γνώσεις από διαφορετικά γνωστικά αντικείμενα. Ωστόσο η πρόκληση δεν σταματάει εδώ. Τα προβλήματα που θα επιλεγούν θα πρέπει να συνδυάζουν έννοιες και δεξιότητες που ταιριάζουν στην ηλικία των μαθητών. Ο σκοπός εδώ είναι η επίτευξη συνδυαστικών διδακτικών στόχων που αναφέρονται στην σχολική ύλη διαφορετικών γνωστικών αντικειμένων. Ένα τέτοιο παράδειγμα αποτελεί ο διαγωνισμός Mission Space Lab (Honest & Quinlan, 2017) που διοργανώνει το ίδρυμα Raspberry Pi μαζί με τον Ευρωπαϊκό Διαστημικό Οργανισμό (ESA). Οι μαθητές έχουν μια μοναδική ευκαιρία να σχεδιάσουν ένα πείραμα το οποίο θα εκτελεστεί στον Διεθνή Διαστημικό Σταθμό (ISS).

Το πείραμα που θα προτείνουν θα υλοποιηθεί στην γλώσσα προγραμματισμού Python και θα εκτελεστεί στο Astro Pi, το οποίο είναι ένας υπολογιστής Raspberry Pi πάνω στον οποίο υπάρχει η πλακέτα Sense Hat η οποία περιλαμβάνει διάφορους αισθητήρες όπως θερμοκρασίας, υγρασίας, βαρομετρικής πίεσης, γυροσκόπιο και άλλα. Οι μαθητές μπορούν να χρησιμοποιήσουν και δυο κάμερες που έχει το Astro Pi για να τραβήξουν φωτογραφίες από την Γη τις οποίες θα επεξεργαστούν αργότερα.

Στην εργασία αυτή παρουσιάζουμε κάποια προκαταρκτικά αποτελέσματα από την συμμετοχή 3 ομάδων μαθητών στον διαγωνισμό αυτό. Ο σκοπός μας δεν ήταν κάποια διάκριση αλλά η ενασχόληση των μαθητών με τον προγραμματισμό και την επίλυση πραγματικών προβλημάτων. Η εκτέλεση του δικού τους κώδικα στον διεθνή διαστημικό σταθμό αποτέλεσε ένα ισχυρό κίνητρο για τους μαθητές.

Η πανδημία και το κλείσιμο των σχολείων το μεγαλύτερο μέρος της σχολικής χρονιάς μας υποχρέωσε να δουλέψουμε εξ αποστάσεως. Έπρεπε οι μαθητές να σχεδιάσουν τις λύσεις τους σε κάποια γλώσσα προγραμματισμού και να δουλέψουν σε ομάδες. Για αυτό χρησιμοποιήσαμε συνεργατικά εργαλεία για την ανάπτυξη εφαρμογών όπως το replit και ο εξομοιωτής του Astro Pi. Η δυνατότητα να μπορούν οι μαθητές να γράφουν κώδικα και να δοκιμάζουν τα προγράμματά τους μέσα από έναν browser έδρασε καταλυτικά για την επιτυχία της δράσης. Με αυτόν τον τρόπο αν και απομακρυσμένα οι μαθητές κατάφεραν να συνεργαστούν και οι καθηγητές να τους καθοδηγήσουμε. Στην συνέχεια θα παρουσιάσουμε τον υπολογιστή Astro Pi και τις δυνατότητές του, τον διαγωνισμό Mission Space Lab της ESA και τα περιβάλλοντά προγραμματισμού που αξιοποιήθηκαν. Στο τέλος αναλύουμε κάποιες από τις ιδέες των μαθητών, τον τρόπο με τον οποίο τις υλοποίησαν σε Python και την σύνδεση των δραστηριοτήτων με το πρόγραμμα σπουδών.

Ο μικροϋπολογιστής Astro Pi

Το Raspberry Pi είναι ένα σύγχρονος μικροϋπολογιστής σε μέγεθος πιστωτικής κάρτας με πολύ χαμηλό κόστος αγοράς και με πολλές δυνατότητες. Η ανάπτυξη του πρώτου Raspberry Pi έγινε από μια ομάδα επιστημόνων υπό την καθοδήγηση του Eben Upton στο Πανεπιστήμιο του Cambridge. Ο σκοπός τους ήταν η κατασκευή ενός υπολογιστή χαμηλού κόστους που θα μπορούσε να χρησιμοποιηθεί για εκπαιδευτικές εφαρμογές σε σχολεία και πανεπιστήμια (Upton & Halfacree, 2012). Το 2014

ιδρύθηκε το ίδρυμα Raspberry με σκοπό να παρέχει αυτούς τους υπολογιστές σε εκπαιδευτικά ιδρύματα με πολύ χαμηλό κόστος αλλά και να σχεδιάζει εκπαιδευτικές δραστηριότητες για την προώθηση και αναβάθμιση της επιστήμης της πληροφορικής στην εκπαίδευση. Το Raspberry Pi εκτός από ένας απλός μικροϋπολογιστής μπορεί να χρησιμοποιηθεί σε δραστηριότητες ρομποτικής ή εφαρμογές IoT (Internet of Things) λόγω των 40 θυρών (GPIO) εισόδου-εξόδου που έχει και στις οποίες μπορούν να συνδεθούν διάφοροι αισθητήρες ή άλλες συσκευές. Το λειτουργικό σύστημα που χρησιμοποιείται είναι μια παραλλαγή του Linux το Raspbian, αλλά υπάρχουν και άλλες διανομές του Linux που μπορεί κάποιος να χρησιμοποιήσει.

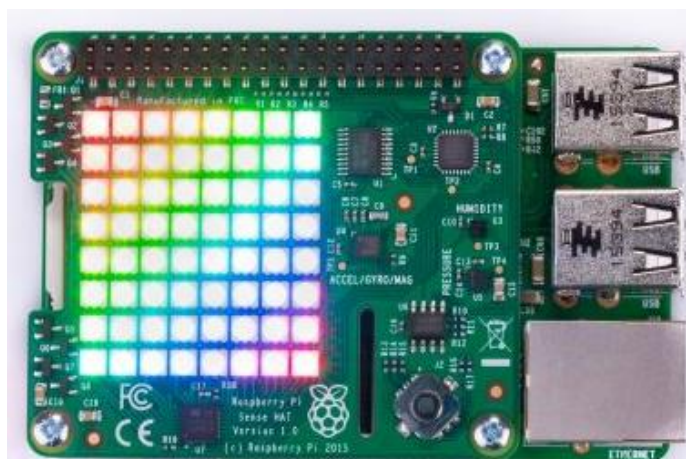


Σχήμα 1. Ο υπολογιστής Raspberry Pi 3 μοντέλο B.

Γενικές οδηγίες

Το Sense Hat είναι ένα πρόσθετο (add-on) του Raspberry Pi, το οποίο περιέχει μια σειρά από αισθητήρες και έναν πίνακα με λαμπάκια LED (8x8), καθώς και ένα μικρό joystick στο πάνω μέρος του. Η πλακέτα αυτή δημιουργήθηκε αρχικά αποκλειστικά για τους διαγωνισμούς του Ευρωπαϊκού Διαστημικού Οργανισμού, αλλά λόγω της μεγάλης επιτυχίας που είχε είναι πλέον διαθέσιμο στην αγορά. Έχει αισθητήρες θερμοκρασίας, υγρασίας, βαρομετρικής πίεσης, προσανατολισμού και πολλούς άλλους. Συνδέεται πάνω στο Raspberry Pi μέσω των θυρών GPIO.

Αυτή η συστοιχία αισθητήρων επιτρέπει στον/στην καθηγητή/τρια τον σχεδιασμό και την υλοποίηση δραστηριοτήτων STEM όπως την δημιουργία ενός μετεωρολογικού σταθμού, ενός σειсмоγράφου ή ενός οχήματος χαρτογράφησης των κλιματικών συνθηκών μιας περιοχής.



Σχήμα 2. Μια πλακέτα Sense Hat πάνω σε ένα Raspberry Pi.

Το Astro Pi είναι ένας υπολογιστής Raspberry Pi πάνω στον οποίο είναι συνδεδεμένη η πλακέτα Sense Hat μέσα σε ένα περίβλημα που είναι ειδικά σχεδιασμένο για τις συνθήκες που υπάρχουν στο διάστημα..

Ο διαγωνισμός Astro Pi Mission Space Lab

Ο διαγωνισμός Mission Space Lab αποτελεί μια καινοτόμο εκπαιδευτική δραστηριότητα του Ευρωπαϊκού Οργανισμού Διαστήματος (ESA) σε συνεργασία με το ίδρυμα Raspberry Pi. Προσφέρει στους μαθητές μια καταπληκτική ευκαιρία για διεξαγωγή επιστημονικών πειραμάτων στο διάστημα υλοποιώντας λογισμικό που θα εκτελεστεί στον Διεθνή Διαστημικό Σταθμό. Η πρόκληση που στην οποία καλούνται να ανταποκριθούν οι μαθητές είναι ο σχεδιασμός και η κωδικοποίηση ενός πειράματος που θα εκτελεστεί σε έναν υπολογιστή Astro Pi. Τα καλύτερα πειράματα θα αναπτυχθούν στο ISS, και οι ομάδες θα έχουν την ευκαιρία να αναλύσουν τα αποτελέσματα που θα λάβουν και να συντάξουν μια τελική έκθεση.

Για να συμμετάσχουν στο Mission Space Lab, οι ομάδες θα πρέπει αρχικά να βρουν μια ιδέα για ένα πείραμα το οποίο θα ανήκει σε μια από τις κατηγορίες και α) Ζωή στο διάστημα β) Ζωή στην Γη. Στην πρώτη περίπτωση το πείραμα καταγράφει και αναλύει δεδομένα που συλλέγονται από τους αισθητήρες μέσα στον διαστημικό σταθμό και εκτελείται στον υπολογιστή Astro Pi Ed.. Στην δεύτερη περίπτωση το πείραμα των μαθητών θα εκτελεστεί στον υπολογιστή Astro Pi με την ονομασία Izzy ο οποίος εκτός από τους αισθητήρες είναι εφοδιασμένος με μια κάμερα υπέρυθρης ακτινοβολίας η οποία είναι τοποθετημένη σε ένα παράθυρο του ISS στραμμένη προς τη Γη. Ο σκοπός εδώ είναι να διερευνηθούν οι συνθήκες ζωής στον πλανήτη Γη. Κάποιες ιδέες που έχουν οι μαθητές είναι να μελετήσουν τις μεταβολές στην επιφάνεια του πλανήτη λόγω της κλιματικής αλλαγής όπως η επιφάνεια των δασών, των λιμνών και των ποταμών.

Οι μαθητές μπορούν να είναι από 11 έως και 19 έτη και οι ομάδες αποτελούνται από 2 έως 6 μαθητές, που επιβλέπονται από έναν καθηγητή-μέντορα. Τουλάχιστον το 50% των μαθητών πρέπει να είναι από χώρες-μέλη του Ευρωπαϊκού Οργανισμού Διαστήματος (ESA) ή του Καναδά, της Λετονίας, της Σλοβενίας ή της Μάλτας.

Αρχικά οι μαθητές καταθέτουν τις ιδέες τους. Αν περάσουν την 1^η φάση τότε τους αποστέλλεται ένας υπολογιστής Astro Pi ώστε να συνεχίσουν με την υλοποίηση του πειράματος. Πριν στείλουν τον κώδικά τους για να ελεγχθεί πρέπει οι ομάδες να τον έχουν δοκιμάσει στο δικό τους Astro Pi. Ωστόσο όσοι μαθητές δεν έχουν τον εξοπλισμό μπορούν να αξιοποιήσουν τον εξομοιωτή του Astro Pi που διατίθεται ελεύθερα στο περιβάλλον προγραμματισμού Trinket το οποίο μπορεί να εκτελεστεί μέσα από έναν browser, όπως φαίνεται στο σχήμα .

Αν ο κώδικας των μαθητών περάσει τα τεστ των ειδικών τότε παίρνει κατάσταση πτήσης (flight status), μεταφορτώνεται στον ISS και εκτελείται εκεί για 3 ώρες. Στη συνέχεια τα δεδομένα που συλλέγονται και δεν πρέπει να υπερβαίνουν τα 3GB, αποστέλλονται πίσω στους μαθητές για ανάλυση. Στο τέλος οι ομάδες υποβάλλουν μια αναφορά/έκθεση με την ανάλυση των δεδομένων που έλαβαν.

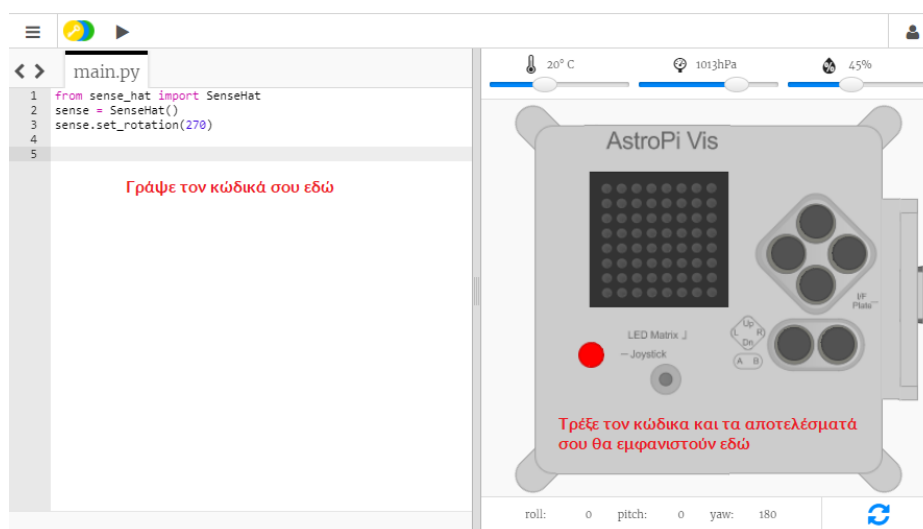
Ο χρόνος των 3 ωρών και το συνολικό μέγεθος των 3GB είναι κάποιος από τους περιορισμούς που πρέπει να προσέξουν οι μαθητές όταν θα αναπτύξουν το λογισμικό που θα εκτελέσει το πείραμα. Ο κώδικας των μαθητών πρέπει να είναι γραμμένος στην γλώσσα Python και να χρησιμοποιεί τις βιβλιοθήκες που έχουν αναπτυχθεί αποκλειστικά για το Astro Pi. Άλλωστε το Pi στο Raspberry Pi παραπέμπει στην γλώσσα Python (Upton, 2012). Αρχικά οι μαθητές μπορούν να χρησιμοποιήσουν τον διαδικτυακό εξομοιωτή και στη συνέχεια τον εξοπλισμό που θα τους στείλει το ίδρυμα Raspberry Pi.

Περιβάλλοντα προγραμματισμού εξ αποστάσεως

Μια δυσκολία που μπορεί να αντιμετωπίσει κάποιος που θέλει να διδάξει προγραμματισμό και σχεδίαση αλγορίθμων σε μαθητές σε περίοδο που τα σχολεία είναι κλειστά είναι η χρήση κάποιου περιβάλλοντος προγραμματισμού που θα υποστηρίζει τον διαμοιρασμό κώδικα μεταξύ των μαθητών και του καθηγητή. Τα διαδικτυακά περιβάλλοντα έχουν τα εξής πλεονεκτήματα:

- Δεν απαιτούν εγκατάσταση στον Η/Υ του μαθητή, άρα δεν υπάρχουν ιδιαίτερες απαιτήσεις σε υπολογιστικούς πόρους.
- Η συγγραφή και εκτέλεση κώδικα μπορεί να γίνει ακόμα και από ένα tablet σε περίπτωση που ο μαθητής δεν έχει υπολογιστή, κάτι που αποτελεί τεράστιο πλεονέκτημα γιατί σε μια τετραμελή οικογένεια με δυο γονείς που εργάζονται με τηλεργασία και δυο παιδιά που παρακολουθούν την τηλεκπαίδευση δεν είναι δυνατόν να υπάρχουν διαθέσιμοι κάθε στιγμή τέσσερις υπολογιστές.
- Ο μαθητής δεν χρειάζεται να στείλει τον κώδικά του σε κάποιο αρχείο στον καθηγητή αν έχει κάποιο πρόβλημα, Του διαμοιράζει απλά τον κώδικά του στο εργαλείο και ο καθηγητής έχει άμεση πρόσβαση
- Αντίστοιχα ο καθηγητής μπορεί να διαμοιράσει κώδικα σε μια ομάδα μαθητών.

Αυτές τις δυνατότητες εκτός από τον διαμοιρασμό κώδικα τις παρέχει ο εξομοιωτής του Astro Pi που διατίθεται ελεύθερα στο περιβάλλον προγραμματισμού Trinket και μπορεί να εκτελεστεί μέσα από έναν browser, όπως φαίνεται στο σχήμα 3.



Σχήμα 3. Εικόνα από τον οδηγό του Astro Pi για τον εξομοιωτή στο Trinket.io.

Την δυνατότητα δημιουργίας τάξεων/ομάδων και διαμοιρασμού κώδικα την παρέχει το Replit, Το Replit είναι ένα νέο εργαλείο το οποίο έχει αναπτυχθεί ιδιαίτερα το τελευταίο έτος και παρέχει πολλές δυνατότητες για την δημιουργία και παρακολούθηση ομάδων προγραμματιστών.



```
main.py
1 import random
2
3 dice1 = random.randint(1,6)
4 dice2 = random.randint(1,6)
5
6 roll = dice1 + dice2
7
8 answer = int( input("Δώσε την πρόβλεψη : "))
9 if (roll == answer) :
10     print("κέρδισες")
11 else :
12     print("ΕΧΑΣΕΣ!!!")
13
14 print("Επιξες", dice1, dice2, "=", roll)
15
16
```

Console Shell

```
Δώσε την πρόβλεψη : 4
ΕΧΑΣΕΣ!!!
Επιξες 6 5 = 11
```

Σχήμα 4. Το διαδικτυακό περιβάλλον προγραμματισμού Replit.

Σύνδεση με το πρόγραμμα σπουδών πληροφορικής

Πολλές φορές προσπαθούμε να βρούμε ενδιαφέρουσες δραστηριότητες που θα τραβήξουν το ενδιαφέρον των μαθητών και απομακρυνόμαστε από τον στόχο μας που είναι η επίτευξη των διδακτικών στόχων του προγράμματος σπουδών. Οι δραστηριότητες που προτείνονται θα πρέπει να έχουν κάποια σύνδεση με έννοιες του προγράμματος σπουδών των γνωστικών αντικείμενων που εμπλέκονται.

Οι περισσότερες ενότητες του προγράμματος σπουδών της πληροφορικής στο Γυμνάσιο αναφέρονται στην αξιοποίηση των ΤΠΕ για την επίλυση προβλημάτων. Μια εξαίρεση στον κανόνα αυτό αποτελεί η ενότητα «Ψηφιακός κόσμος» όπου οι μαθητές εισάγονται σε βασικές έννοιες αναπαράστασης και κωδικοποίησης πληροφορίας. Εκτός από την δυαδική αναπαράσταση της πληροφορίας και τις τάξεις μεγέθους (MegaByte, GigaByte κλπ) υπάρχει η ενότητα «Πολυμέσα» στην οποία γίνεται αναφορά στην αναπαράσταση των χρωμάτων των εικονοστοιχείων (pixels) μιας εικόνας με τρία Bytes ένα για κάθε χρώμα του μοντέλου RGB. Κάθε ένα από $8 \times 8 = 64$ LEDs του Astro Pi μπορούν να ανάψουν σε συγκεκριμένο χρώμα που θα ορίσουμε εμείς όπως φαίνεται στο σχήμα 4.

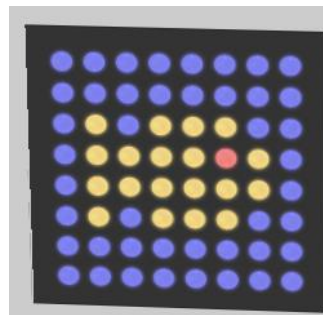
Η εικόνα που εμφανίζεται ορίζεται ως μια λίστα με 64 τιμές. Κάθε τιμή είναι μια τριάδα αριθμών, που αναπαριστούν την φωτεινότητα των τριών χρωμάτων (Red, Green, Blue). Οι τρεις αυτοί αριθμοί σχηματίζουν μια δομή που είναι γνωστή στην Python ως πλειάδα (tuple). Στην συνέχεια όλες αυτές οι πλειάδες (μια για κάθε pixel/Led της εικόνας) τοποθετούνται σε μια λίστα με το όνομα image. Πιο πριν έχουμε ορίσει τα χρώματα που χρειαζόμαστε. Για παράδειγμα το μπλε ορίζεται από την τριάδα των αριθμών (255, 0, 0) αφού έχει την μέγιστη φωτεινότητα σε μπλε και οι άλλοι είναι μηδέν, ενώ το πορτοκαλί είναι ανάμειξη κόκκινου και πράσινου, με το κόκκινο να είναι το πιο ισχυρό. Δίνοντας στους μαθητές την ευκαιρία να ζωγραφίσουν την δική τους εικόνα τους αφήνουμε να πειραματιστούν με διάφορα χρώματα.

Πέρα από τις έννοιες που αναφέρθηκαν παραπάνω εδώ παρουσιάζονται ταυτόχρονα και απλές προγραμματιστικές δομές, όπως η αυτές της μεταβλητής, της πλειάδας και της λίστας.

```

1  from sense_hat import SenseHat
2  sense = SenseHat()
3  sense.set_rotation(270)
4
5  humidity = round( sense.get_humidity(), 2 )
6  sense.show_message( str( humidity ) )
7  o = orange = (255,130,0)
8  b = blue =(0,0,255)
9  r = red =(255,0,0)
10
11 image = [
12     b, b, b, b, b, b, b, b,
13     b, b, b, b, b, b, b, b,
14     b, o, b, o, o, o, b, b,
15     b, o, o, o, o, r, o, b,
16     b, o, o, o, o, o, o, b,
17     b, o, b, o, o, o, b, b,
18     b, b, b, b, b, b, b, b,
19     b, b, b, b, b, b, b, b
20 ]
21 sense.set_pixels(image)

```



Σχήμα 5. Εμφάνιση εικόνας στον εξομοιωτή του Astro Pi.

Ένα σημαντικό πρόβλημα που τέθηκε στους μαθητές ήταν να αναπτύξουν ένα πρόγραμμα το οποίο θα κατέγραφε φωτογραφίες από την επιφάνεια της Γης στην μέγιστη δυνατή ανάλυση, χωρίς όμως να παραβιάζει τους περιορισμούς του διαγωνισμού. Οι περιορισμοί που τέθηκαν είναι το πρόγραμμα να μην εκτελείται πάνω από 3 ώρες και να τα δεδομένα που θα συλλέξει να μην υπερβαίνουν τα 3 GB.

Θέσαμε λοιπόν το ερώτημα αυτό στους μαθητές. Τους ζητήσαμε καταρχάς να υπολογίσουν με ποιον ρυθμό πρέπει να φωτογραφίζει η κάμερα ώστε να συλλέξουμε όσο το δυνατόν περισσότερες φωτογραφίες χωρίς να ξεπεράσουμε τα 3 GB. Οι μαθητές είχανε δυο προβλήματα να λύσουν.

- Τι μέγεθος έχει μια φωτογραφία στην μέγιστη δυνατή ανάλυση
- Με τι ρυθμό πρέπει να τραβάει φωτογραφίες το πρόγραμμα

Η απάντηση στο πρώτο ερώτημα δόθηκε πειραματικά. Εδώ είμασταν πάρα πολύ τυχεροί, γιατί τα σχολεία άνοιξαν τον Ιανουάριο του 2021 για λίγες εβδομάδες και επίσης είχαμε λάβει τον εξοπλισμό Astro Pi. Έτσι στις λίγες μέρες που είχαν οι μαθητές ακολουθώντας τις οδηγίες από την ιστοσελίδα της δραστηριότητας συνέδεσαν το Raspberry Pi με το Sense Hat και την κάμερα και δοκίμασαν να τραβήξουν μερικές φωτογραφίες. Το αποτέλεσμα ήταν ότι οι φωτογραφίες ήταν κατά μέσο όρο 3 GB. Το πιθανότερο όμως είναι ότι δεν θα ήταν όλες σε αυτό το μέγεθος γιατί οι φωτογραφίες που θα τραβήξει το Astro Pi τη νύχτα θα είναι σίγουρα μικρότερες. Εδώ να σημειωθεί ότι ο Διεθνής Διαστημικός Σταθμός θέλει 1,5 ώρα για να εκτελέσει μια πλήρη τροχιά γύρω από την Γη.

Στην επόμενη ενότητα δίνουμε τρεις διαφορετικές λύσεις που έδωσαν οι μαθητές για αυτό το πρόβλημα και προσπαθούμε να αναλύσουμε τον τρόπο σκέψης τους. Οι μαθητές που συμμετείχαν σε αυτή την δραστηριότητα είναι μέλη ενός ομίλου με αντικείμενο τον προγραμματισμό με τη γλώσσα Python που γίνεται μετά το σχολείο, άρα ήταν εξοικειωμένοι με τις βασικές δομές ελέγχου της Python.

Μελέτη των λύσεων των μαθητών

Παρακάτω δίνουμε τμήματα του κώδικα των μαθητών και όχι ολόκληρη την λύση λόγω του περιορισμένου χώρου του άρθρου. Στον πρώτο αλγόριθμο που φαίνεται στο σχήμα 5 οι μαθητές ελέγχουν δυο συνθήκες. Η μια είναι ο χρόνος και η δεύτερη το πλήθος των αρχείων. Δεδομένου ότι $3GB/3MB = 1000$ αρχεία έβαλαν ένα όριο γύρω στις 700 φωτογραφίες ώστε να

σιγουρευτούν ότι δεν θα είναι κοντά στα όρια. Ο δεύτερος έλεγχος αφορά τον χρόνο. Η μεταβλητή `now` περιέχει σε κάθε επανάληψη την ακριβή ώρα ενώ η `start` την ώρα που ξεκίνησε το πείραμα. Το `delta` είναι μια απόσταση ασφαλείας σε περίπτωση που υπάρξει κάποια απόκλιση σε υπολογισμούς ή στον υπολογισμό της ώρας. Άλλωστε το πείραμα θα γίνει στο διάστημα σε συνθήκες έλλειψης βαρύτητας, και για αυτό οι μαθητές δεν ήθελαν να ρισκάρουν. Άλλωστε 700 φωτογραφίες είναι αρκετές. Η εντολή `sleep` παγώνει την εκτέλεση του αλγορίθμου για 15 δευτερόλεπτα. Τόσο υπολόγισαν οι μαθητές το μεσοδιάστημα μεταξύ δυο διαδοχικών λήψεων. Είναι φανερό ότι η συνθήκη του χρόνου θα μπορούσε να λείπει αλλά πάλι οι μαθητές επέμεναν να προσθέσουν και αυτή για ασφάλεια.

```
while (now < start + delta and FileNo < 700):
    image_file = str(dir_path) + "/images_" +
                 str(FileNo).zfill(3) + ".jpg"
    capture(camera, image_file)

    sleep(15)
    now = datetime.datetime.now()
    FileNo = FileNo + 1
```

Σχήμα 6. Αλγόριθμος 1ης ομάδας για την λήψη φωτογραφιών της επιφάνειας της Γης.

Στην επόμενη λύση στο σχήμα 6 οι μαθητές το ρίσκαραν περισσότερο και έθεσαν χρονικό διάστημα μεταξύ δυο λήψεων τα 10 δευτερόλεπτα.

```
delta = datetime.timedelta(minutes=TIME_LIMIT)
while (now < start+delta and total_size < SIZE_LIMIT):
    photos = photos + 1;

    image_file = str(dir_path) + "/image_" +
                 str(photos).zfill(3) + ".jpg"
    capture(camera, image_file)

    image = Image.open(image_file)
    size_in_bytes = str(len(image.fp.read()))
    total_size = total_size + int(size_in_bytes)

    sleep(10)
    now = datetime.datetime.now()
```

Σχήμα 7. Αλγόριθμος 2ης ομάδας για την λήψη φωτογραφιών της επιφάνειας της Γης.

Τώρα δεν χρησιμοποιούν άνω φράγμα για το πλήθος των φωτογραφιών αλλά ελέγχουν απευθείας το συνολικό μέγεθος και τον χρόνο, διατηρώντας πάντα μικρή απόσταση ασφαλείας. Για παράδειγμα το `SIZE_LIMIT` είχε δηλωθεί παραπάνω ίσο με `2800 MegaBytes = 2,8 GB < 3 GB`.

Η τελευταία λύση της τρίτης ομάδας που δίνεται στο σχήμα 7 έχει περισσότερο ενδιαφέρον γιατί ενώ δεν έχει μεγάλη διαφορά από την λύση της πρώτης ομάδας στους ελέγχους εδώ οι μαθητές έχουν επιλέξει να χρησιμοποιήσουν την δομή επανάληψης `for` σε συνδυασμό με την εντολή `break` και όχι την `while`. Το πλήθος των φωτογραφιών είναι μικρότερο γιατί το διάστημα μεταξύ δυο διαδοχικών λήψεων είναι μεγαλύτερο (20 δευτερόλεπτα). Όταν ρωτήσαμε τους μαθητές γιατί χρησιμοποίησαν την `for` μαζί με την `break` και όχι την `while` με διπλή συνθήκη μας απάντησαν ότι αυτή η προσέγγιση τους φάνηκε πιο απλή. Πιο

συγκεκριμένα μας είπαν ότι με αυτόν τον τρόπο έχουν την αίσθηση ότι ελέγχουν καλύτερα την εξέλιξη του προγράμματος, γιατί ξέρουν ότι αν το πρόγραμμα ξεπεράσει τον χρόνο δεν θα εκτελεστούν άλλες εντολές αλλά η εκτέλεση θα τερματιστεί άμεσα.

```
now = datetime.datetime.now()
TIME_LIMIT = 175
total_size = 0
delta = datetime.timedelta(minutes=TIME_LIMIT)
for photo_counter in range(1, 521):

    image_file = str(dir_path) + "/image_" +
                str(photo_counter).zfill(3) + ".jpg"
    capture(camera, image_file)

    sleep(20)
    now = datetime.datetime.now()
    if (now - start >= delta) :
        break
```

Σχήμα 8. Αλγόριθμος 3ης ομάδας για την λήψη φωτογραφιών της επιφάνειας της Γης.

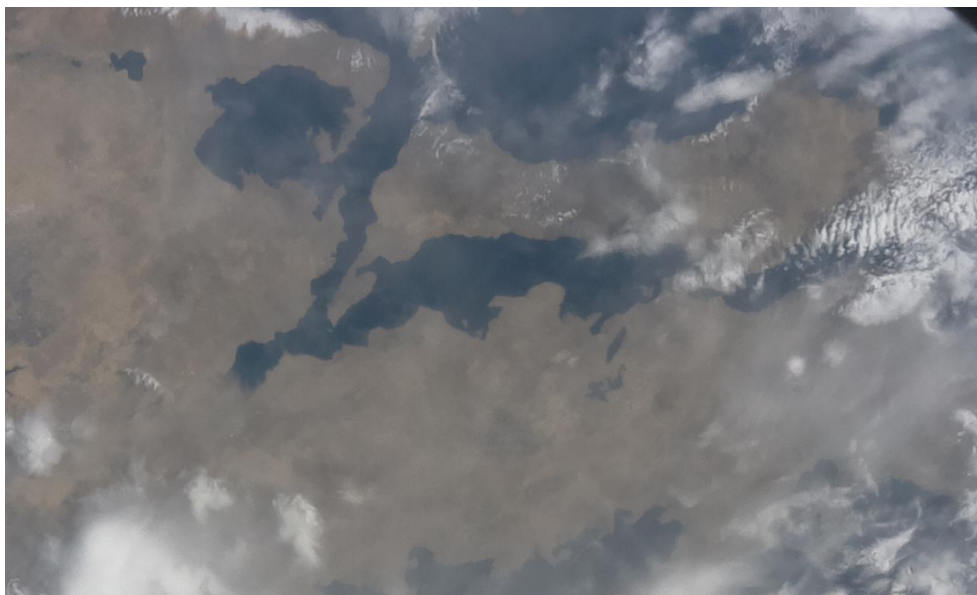
Παρόλο που δεν ωθούμε τους μαθητές να χρησιμοποιούν την εντολή break κάποιοι μαθητές την θεωρούν πολύ χρήσιμη. Αυτό ταιριάζει και με τα αποτελέσματα σχετικών ερευνών στην ελληνική και τη διεθνή βιβλιογραφία για αυτό το θέμα (Βραχνός, 2019; Sorva & Vihavainen, 2016).

Αποτελέσματα – Εικόνες από τον Διεθνή Διαστημικό Σταθμό

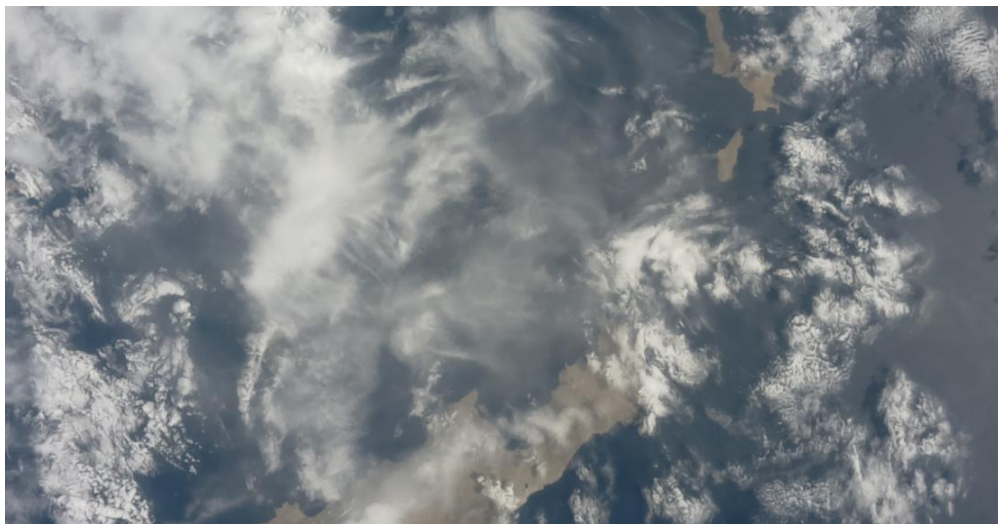
Μετά την πρόκριση και των τριών ομάδων στην τελική φάση του διαγωνισμού και την εκτέλεση του κώδικά τους στον Διεθνή Διαστημικό Σταθμό λάβαμε τα δεδομένα των προγραμμάτων μας, δηλαδή συνολικά 7 GB (2.500 φωτογραφίες). Σε κάθε εικόνα έχουμε ορίσει έναν μοναδικό αριθμό ο οποίος παραπέμπει σε ένα αρχείο – ευρετήριο όπου έχουν καταχωρηθεί οι συντεταγμένες της εικόνας. Έτσι οι μαθητές ανέλαβαν να μελετήσουν όλες τις εικόνες και να βρουν γεωγραφικά την περιοχή στην οποία ανήκουν μέσω του Google Maps. Παραθέτουμε μερικές από αυτές οι οποίες έχουν ενδιαφέρον. Μια ομάδα μας ήταν πολύ τυχεροί γιατί σε αυτή την περίπτωση ο ΔΔΣ πέρασε πάνω από περιοχές της Ελλάδας τις οποίες αναγνωρίσαμε. Παρακάτω παραθέτουμε κάποιες από τις χιλιάδες φωτογραφίες που κατέγραψαν τα προγράμματα των μαθητών.



Σχήμα 9. Κέρκυρα.



Σχήμα 10. Μαγνησία, Εύβοια, Αττική



Σχήμα 11. Κρήτη, Κάρπαθος με συννεφιά



Σχήμα 12. Η διώρυγα του Σουέζ

Συμπεράσματα

Στην εργασία αυτή παρουσιάσαμε τον σχεδιασμό και την υλοποίηση δραστηριοτήτων αναφορικά με τον διαγωνισμό Mission Space Lab που διοργανώνει το ίδρυμα Raspberry Pi μαζί με τον Ευρωπαϊκό Διαστημικό Οργανισμό (ESA). Οι μαθητές έχουν μια μοναδική ευκαιρία να σχεδιάσουν ένα πείραμα το οποίο θα εκτελεστεί στον Διεθνή Διαστημικό Σταθμό (ISS). Η πανδημία και το κλείσιμο των σχολείων το μεγαλύτερο μέρος της σχολικής χρονιάς μας υποχρέωσε να δουλέψουμε εξ αποστάσεως. Έπρεπε οι μαθητές να σχεδιάσουν τις λύσεις τους σε κάποια γλώσσα προγραμματισμού και να δουλέψουν σε ομάδες. Για αυτό χρησιμοποιήσαμε συνεργατικά εργαλεία για την ανάπτυξη εφαρμογών όπως το replit και ο εξομοιωτής του Astro Pi. Η δυνατότητα να μπορούν οι μαθητές να γράφουν κώδικα και να δοκιμάζουν τα προγράμματά τους μέσα από έναν browser έδρασε καταλυτικά για την επιτυχία της δράσης. Με αυτόν τον τρόπο αν και απομακρυσμένα οι μαθητές κατάφεραν να συνεργαστούν και οι καθηγητές να τους καθοδηγήσουμε.

Στα προκαταρκτικά αποτελέσματα αυτής της δράσης κάτι που παρουσιάζει ενδιαφέρον είναι ο τρόπος με τον οποίο αντιμετώπισαν οι μαθητές την ανάπτυξη αλγορίθμων που δεν θα παραβιάζουν τα χρονικά και χωρικά όρια που τέθηκαν από τον διαγωνισμό. Πέρα από τις δομημένες λύσεις μια ομάδα χρησιμοποίησε την εντολή break σε συνδυασμό με την for κάτι που έχουμε συναντήσει και άλλες φορές σε παρόμοια προβλήματα και αναφέρεται στην βιβλιογραφία. Αυτό δείχνει ότι η χρήση της break διευκολύνει αρκετά τους αρχάριους προγραμματιστές και όχι μόνο στην ανάπτυξη τμημάτων κώδικα με πολλαπλές συνθήκες τερματισμού της επανάληψης.

Αυτή την στιγμή οι μαθητές έχουν λάβει τις φωτογραφίες του δικού τους κώδικα που εκτελέστηκε στον διαστημικό σταθμό και είναι στην φάση της επεξεργασίας τους.

Αναφορές

Dewey, J. (1990). *The School and Society and the Child and the Curriculum*. Chicago: The University of Chicago Press

Honess, D. & Quinlan, O. (2017). Astro Pi: Running your code aboard the International Space Station, *Acta Astronautica*, Volume 138, 43-52.

Sorva, J. & Vihavainen, A. (2016). Break Statement Considered. *ACM Inroads*, 7(3), 36-41.

Upton, E. & Halfacree, G. (2012). *Meet the Raspberry Pi*. Wiley.

Βραχνός, Ε. (2019). *Υπερβαίνοντας τα στερεότυπα του δομημένου προγραμματισμού*. 11th Conference on Informatics in Education, σελ. 99-108.

Replit, <https://replit.com/>, Ανακτήθηκε στις 07/02/2022.

Raspberry Pi, <https://www.raspberrypi.org/>, Ανακτήθηκε στις 07/02/2022.

ESA, <https://www.esa.int/>, Ανακτήθηκε στις 07/02/2022.

Trinket, <https://trinket.io/>, Ανακτήθηκε στις 07/02/2022.

Urton, E. (2012). Συνέντευξη του Eben Urton ενός εκ των ιδρυτών του ιδρύματος Raspberry Pi. <https://www.techspot.com/article/531-eben-urton-interview/>, Ανακτήθηκε στις 07/02/2022.