

Αξιολόγηση Γλωσσών Προγραμματισμού για τη Σχολική Εκπαίδευση

Χάρης Β. Γεωργίου¹

hgeorgiou@unipi.gr

¹Μεταδιδακτορικός ερευνητής, Data Science Lab, Πανεπιστήμιο Πειραιώς

ΠΕΡΙΛΗΨΗ

Η διδασκαλία της Πληροφορικής και ειδικότερα η εισαγωγή στις βασικές έννοιες αλγοριθμικής σκέψης και προγραμματισμού αποτελούν μια σύνθετη διαδικασία και εκπαιδευτική πρόκληση σε πολλά επίπεδα. Ένας από τους κρίσιμους παράγοντες επιτυχίας αποτελεί η κατάλληλη επιλογή των προγραμματιστικών εργαλείων, πρωτίστως της γλώσσας προγραμματισμού η οποία θα χρησιμοποιηθεί ως βασικό εργαλείο.

Η παρούσα μελέτη παρουσιάζει ένα γενικό πλαίσιο καθορισμού των πιο σημαντικών παραγόντων αξιολόγησης, καθώς και της «βαθμολόγησης» μερικών από τις πιο δημοφιλείς γλώσσες προγραμματισμού διεθνώς για το σκοπό αυτό. Σκοπός της μελέτης είναι να διερευνήσει τις διαθέσιμες επιλογές και να προτείνει τις καταλληλότερες εναλλακτικές, προτείνοντας τις απαραίτητες τροποποιήσεις και αναπροσαρμογές των σχολικών προγραμμάτων σπουδών.

Σύμφωνα με τα αποτελέσματα, τα μεν κριτήρια αξιολόγησης επιβεβαιώνονται ως προς τις αναμενόμενες προτεραιότητες της εκπαιδευτικής διαδικασίας σε μαθήματα βασικών εννοιών προγραμματισμού και αλγοριθμικής σκέψης, με έμφαση σε Γυμνάσια και Λύκεια, ενώ από τις γλώσσες προγραμματισμού η Python εμφανίζεται ξεκάθαρα ως η καταλληλότερη επιλογή.

Από τα αποτελέσματα της μελέτης κρίνεται σκόπιμη η άμεση αναπροσαρμογή του προγράμματος σπουδών στα σχολικά μαθήματα Πληροφορικής, έτσι ώστε τόσο η θεωρητική διδασκαλία με παραδείγματα κώδικα, όσο και η πρακτική εξάσκηση με συγγραφή απλών προγραμμάτων, να γίνεται στο εξής στη γλώσσα Python.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Πληροφορική, γλώσσες προγραμματισμού, σχολική εκπαίδευση

ΕΙΣΑΓΩΓΗ

Ένα από τα σημαντικότερα ζητήματα στην εκπαιδευτική διαδικασία σε μαθήματα Πληροφορικής σε σχολικό περιβάλλον, ιδιαίτερα σε επίπεδο Δευτεροβάθμιας εκπαίδευσης, αποτελεί η επιλογή και σωστή χρήση κατάλληλων προγραμματιστικών εργαλείων, γλωσσών προγραμματισμού, καθώς και σχετικού διδακτικού υλικού (Bell, 2018). Οι κύριοι στόχοι των μαθημάτων Πληροφορικής, ήδη από το επίπεδο της Πρωτοβάθμιας, ξεκινούν από την ανάπτυξη βασικής αναλυτικής σκέψης και αλγοριθμικής «σχεδίασης» λύσεων σε απλά προβλήματα, καταλήγοντας σταδιακά στη χρήση ψευδοκώδικα, λογικών διαγραμμάτων και απλού προγραμματιστικού συντακτικού για την αποτύπωση του σκελετού ενός πραγματικού προγράμματος Η/Υ με αρκετά ρεαλιστικό βαθμό λεπτομέρειας (Kanemune, 2022; Kalas, 2011).

Παρόλα αυτά, τα σημερινά προγράμματα σπουδών εμφανίζουν σημαντικές ελλείψεις και προβλήματα συνάφειας, τόσο ως προς τις πραγματικές συνθήκες και δομές μιας σύγχρονης γλώσσας προγραμματισμού γενικής χρήσης, όσο και ως προς τις δυνατότητες των σημερινών Η/Υ. Χαρακτηριστικό παράδειγμα αποτελεί η «ΓΛΩΣΣΑ» που διδάσκεται ως πρότυπη γλώσσα προγραμματισμού στα σχολικά μαθήματα Πληροφορικής στην Ελλάδα (ΥΠΕΘ-ΙΕΠ, 2017), η οποία έχει σχεδιαστεί χρησιμοποιώντας ως πρότυπο τη γλώσσα Pascal της δεκαετίας του '80. Η σχεδιαστική αυτή επιλογή είναι γενικά σωστή, όμως οι δυνατότητες της «ΓΛΩΣΣΑΣ» δεν μπορούν να καλύψουν τις σύγχρονες απαιτήσεις των αντίστοιχων μαθημάτων σήμερα, όπως η δυναμική διαχείριση μνήμης (διδασκαλία δομών δεδομένων), η διεπαφή με γραφικό περιβάλλον (διδασκαλία σχεδίασης/διαγραμμάτων), η επικοινωνία μέσω δικτύου (διδασκαλία web apps), κτλ.

Έχοντας ως στόχο τη διερεύνηση των σημερινών επιλογών ως προς τις γλώσσες προγραμματισμού και κυρίως την αξιολόγησή τους ως προς την καταλληλότητά τους στην εκπαιδευτική διαδικασία εισαγωγής στον προγραμματισμό, η παρούσα μελέτη παρουσιάζει ένα γενικό πλαίσιο καθορισμού των πιο σημαντικών παραγόντων αξιολόγησης, καθώς και της «βαθμολόγησης» μερικών από τις πιο δημοφιλείς γλώσσες προγραμματισμού διεθνώς για το σκοπό αυτό. Η μελέτη βασίζεται στη μέθοδο των ανώνυμων ερωτηματολογίων με συμμετοχή κυρίως εκπαιδευτικών στον τομέα της Πληροφορικής, με στόχευση κυρίως τη Δευτεροβάθμια και Μεταλυκειακή εκπαίδευση.

Σκοπός της μελέτης είναι να διερευνήσει τις διαθέσιμες επιλογές και να προτείνει τις καταλληλότερες εναλλακτικές, δείχνοντας έτσι τους άξονες πάνω στους οποίους θα πρέπει να βασιστούν οι απαραίτητες τροποποιήσεις και αναπροσαρμογές των σχολικών προγραμμάτων σπουδών.

Το υπόλοιπο της παρούσας μελέτης περιλαμβάνει την ενότητα «Μεθοδολογία», όπου περιγράφονται οι λεπτομέρειες σχεδίασης και διεξαγωγής της συλλογής απαντήσεων μέσω online ερωτηματολογίου, την ενότητα «Αποτελέσματα» όπου παρουσιάζονται τα σημαντικότερα από αυτά, την ενότητα «Σχολιασμός» όπου επισημαίνονται οι πιο σημαντικές παρατηρήσεις που προκύπτουν από τη μελέτη, και τέλος την ενότητα «Συμπεράσματα» που ολοκληρώνει τη μελέτη.

ΜΕΘΟΔΟΛΟΓΙΑ

Το κύριο μέρος της αξιολόγησης των γλωσσών προγραμματισμού για τη σχολική εκπαίδευση βασίστηκε σε ανώνυμες απαντήσεις σε ερωτηματολόγιο, με στόχο την αξιολόγηση μερικών από τις πιο δημοφιλείς γλώσσες προγραμματισμού, κατάλληλες για τη διδασκαλία βασικών αρχών αλγοριθμικής σκέψης και υλοποίησης αντίστοιχων προγραμμάτων-παραδειγμάτων σε αυτό το πλαίσιο.

Συγκεκριμένα, το ερωτηματολόγιο σχεδιάστηκε σε δύο μέρη: (α) Βαθμολόγηση των επιλεγμένων γλωσσών προγραμματισμού σε σχέση με μια σειρά παράγοντες αξιολόγησης και (β) πρόταση συντελεστών βαρύτητας για κάθε έναν από τους παράγοντες αξιολόγησης, πάντοτε σε σχέση με την καταλληλότητα και την αποτελεσματικότητα στην εκπαιδευτική διαδικασία, δηλαδή τη διδασκαλία των αντίστοιχων εννοιών. Η βαθμολόγηση των γλωσσών προγραμματισμού και των συντελεστών βαρύτητας έγινε ξεχωριστά, καθώς το μέρος (β) αποτελεί γενικότερη μελέτη του προβλήματος, ανεξάρτητη από τη λίστα επιλεγμένων γλωσσών προγραμματισμού που χρησιμοποιήθηκε στη συγκεκριμένη μελέτη, κατά συνέπεια εφαρμόσιμη και σε άλλες παρόμοιες μελέτες, όπως για παράδειγμα μέσα διδασκαλίας, απαραίτητη υποδομή σε Η/Υ, κτλ.

Σε ό,τι αφορά την επιλογή των γλωσσών προγραμματισμού στο μέρος (α), αυτή βασίστηκε σε προγράμματα σπουδών και αντίστοιχες επιλογές που χρησιμοποιούνται σε σχολεία παγκοσμίως ως μέσο εισαγωγής στην αλγοριθμική σκέψη και στην πρώτη επαφή των μαθητών με προγραμματιστικά περιβάλλοντα. Η πλήρης λίστα γλωσσών προγραμματισμού που αξιολογήθηκε στο ερωτηματολόγιο είναι η παρακάτω:

- C
- C++
- Pascal
- Java
- Python
- Javascript
- R
- Octave
- Άλλη (σχόλιο)

Στη λίστα συμπεριλαμβάνονται και κάποιες γλώσσες που συναντώνται κατά κανόνα στην Τριτοβάθμια εκπαίδευση και σε ακαδημαϊκά περιβάλλοντα, όπως Octave και R, ακριβώς για να διερευνηθούν οι διαφοροποιήσεις σε σχέση με γλώσσες που τυπικά χρησιμοποιούνται σε σχολική εκπαίδευση επιπέδου Γυμνασίου-Λυκείου.

Οι παράγοντες αξιολόγησης στο μέρος (β) αποτελούν μια πολύ σύντομη συλλογή βασικών διδακτικών επιδιώξεων και παραγόντων επιτυχίας της διδασκαλίας, πάντοτε σε σχέση με την ανάπτυξη αναλυτικής-αλγοριθμικής σκέψης και τη δυνατότητα υλοποίησης σε αντίστοιχα προγράμματα-παραδείγματα. Η πλήρης λίστα των παραγόντων αξιολόγησης (22) που χρησιμοποιήθηκε στο ερωτηματολόγιο είναι η παρακάτω:

- Ευκολία εκμάθησης (learning curve)
- Βασικό συντακτικό – keywords
- Δομές δεδομένων (κυρίως εκτός πινάκων)
- Ανοχή και αντιμετώπιση σφαλμάτων κώδικα
- Βασική είσοδος-έξοδος χρήστη (console)
- Αντικειμενοστραφής προγραμματισμός (OOP)
- Διαχείριση μνήμης
- Έλεγχος τύπων (type checking)
- Γραφικές παραστάσεις (plots)

- Βάσεις δεδομένων
- Δικτυακός προγραμματισμός
- Πολυπρογραμματισμός (concurrency tools/safety)
- Εφαρμογές διαδικτύου (web apps)
- Τεχνητή νοημοσύνη – Μηχανική Μάθηση (AI)
- Προγραμματισμός μικροσυσκευών (SBC/MC)
- Ανάπτυξη γραφικού περιβάλλοντος (GUI)
- Απόδοση – ταχύτητα εκτέλεσης
- Μεταφερσιμότητα κώδικα (portability)
- Σύνδεση με επαγγελματικές δεξιότητες
- Σύνδεση με ακαδημαϊκές σπουδές
- Διαθέσιμες πλατφόρμες (IDEs/compiler)
- Διαθέσιμες πηγές – εκπαιδευτικό υλικό

Ως έννοιες οι παράγοντες αξιολόγησης καλύπτουν θέματα με πολύ μεγάλο εύρος τόσο θεωρητικών όσο και πρακτικών γνώσεων, όπως για παράδειγμα «βάσεις δεδομένων» ή «πολυπρογραμματισμός», όμως στη συγκεκριμένη μελέτη η εστίαση είναι αποκλειστικά στη δυνατότητα διδασκαλίας των πολύ βασικών εννοιών σε μαθητές αυτού του επιπέδου και πάντοτε επιλεκτικά ανάλογα με το σχολικό έτος.

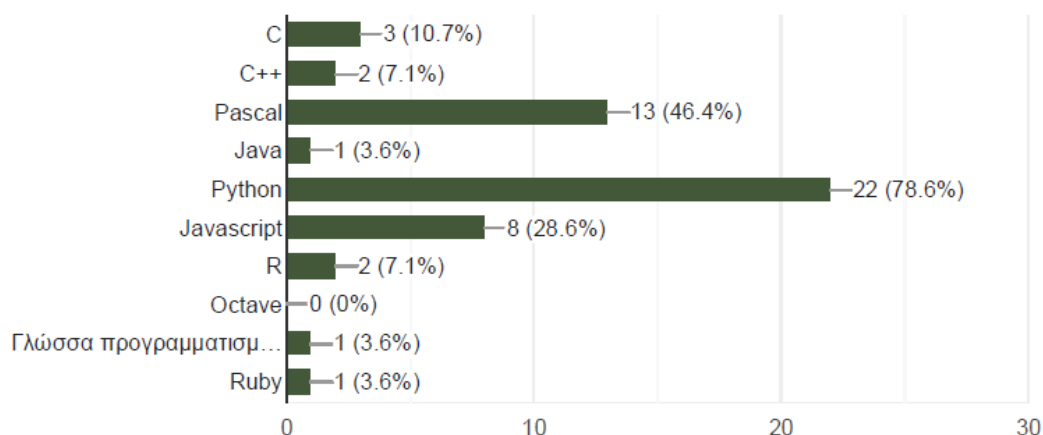
Το ερωτηματολόγιο δημοσιεύτηκε online και έγινε ανοικτή πρόσκληση συμμετοχής σε πτυχιούχους Πληροφορικής που εργάζονται ως διδάσκοντες σε αντίστοιχες εκπαιδευτικές βαθμίδες στο δημόσιο και στον ιδιωτικό τομέα, χωρίς αυτό να αποκλείει και Πληροφορικούς εργαζόμενους σε άλλους τομείς. Η χρονική διάρκεια συλλογής των ανώνυμων απαντήσεων ήταν ένας ημερολογιακός μήνας, συγκεκριμένα ο Φεβρουάριος 2022. Η επιλογή του χρόνου έγινε σύμφωνα και με τα επικαιροποιημένα προγράμματα σπουδών που ανακοινώθηκαν από το Ι.Ε.Π. (Ιαν/2022), έτσι ώστε στη συνέχεια να μπορεί να προταθεί η κατάλληλη επιλογή προγραμματιστικών εργαλείων σε αυτές τις βαθμίδες. Συνεπώς απευθυνόταν κυρίως σε εκπαιδευτικό προσωπικό αυτών των βαθμίδων, με οπωσδήποτε χρήσιμη και τη συμμετοχή αντίστοιχων καθηγητών σε άλλες βαθμίδες, όπως μεταλυκειακής (IEK) και Πανεπιστημιακής (AEI) εκπαίδευσης. Οι ερωτήσεις ήταν πολλαπλής επιλογής και συνολικά απαιτούσαν περίπου 5 λεπτά για να απαντηθούν στο σύνολό τους.

ΑΠΟΤΕΛΕΣΜΑΤΑ

Κατά τη χρονική διάρκεια που το ερωτηματολόγιο ήταν ενεργό συγκεντρώθηκαν συνολικά 28 πλήρεις απαντήσεις διαφορετικών ατόμων, εργασιακού περιβάλλοντος και επιπέδου σπουδών. Έγιναν συνολικά τρεις δημόσιες προσκλήσεις σε κατάλληλα μέσα (mailing lists, social media, ανακοινώσεις σε websites), στοχευμένες κατά κύριο λόγο σε εκπαιδευτικούς στο δημόσιο και ιδιωτικό τομέα, χωρίς να αποκλείονται όμως άλλοι επαγγελματικοί τομείς, έτσι ώστε να συγκεντρωθεί στατιστικά σημαντικό και αντιπροσωπευτικό (unbiased) δείγμα απαντήσεων. Για τον ίδιο λόγο, κατά τη διάρκεια συγκέντρωσης απαντήσεων δεν ανακοινώθηκαν ενδιάμεσα αποτελέσματα, παρά μόνο μετά την ολοκλήρωση της έρευνας.

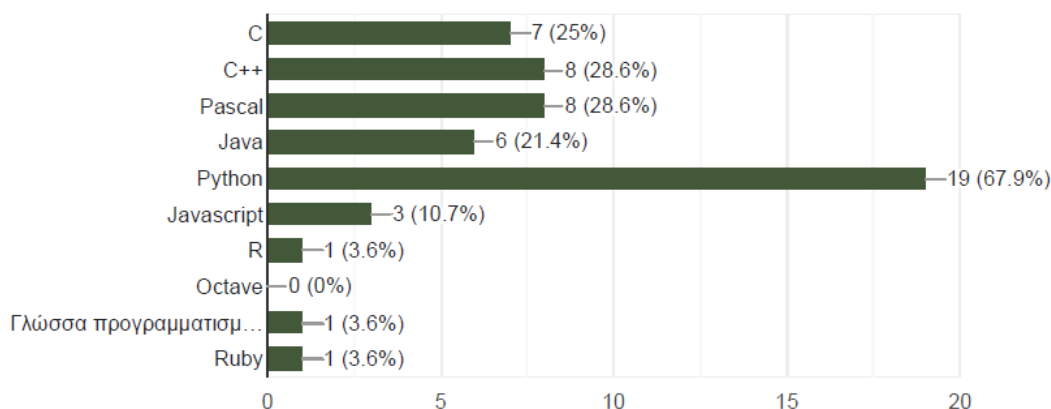
Λόγω του πλήθους των ερωτήσεων και εν γένει του όγκου της έρευνας, παρακάτω παρουσιάζονται μόνο μερικά ενδεικτικά αποτελέσματα που σχετίζονται με τα πιο ενδιαφέροντα συμπεράσματα. Η πλήρης αποτίμηση και όλα τα σχετικά συγκεντρωτικά δεδομένα και διαγράμματα είναι διαθέσιμα online (Γεωργίου, 2022).

Μια από τις πιο δημοφιλείς ερωτήσεις-κριτήρια αξιολόγησης γλωσσών προγραμματισμού για την εκπαίδευση, ειδικά στις χαμηλότερες και μέσες βαθμίδες, είναι η «ευκολία εκμάθησης», η οποία είναι και η πρώτη στη λίστα του ερωτηματολογίου της έρευνας. Από το Σχήμα 1, όπου παρουσιάζεται σε μορφή διαγράμματος (ιστόγραμμα) η σύνοψη των απαντήσεων που συγκεντρώθηκαν, είναι φανερό ότι η γλώσσα Python θεωρείται ξεκάθαρα η καταλληλότερη επιλογή ως προς το κριτήριο αυτό, ακολουθούμενη αρκετά πίσω από τις γλώσσες Pascal και Javascript. Το αποτέλεσμα αυτό συμφωνεί με την τάση των τελευταίων ετών να εισαχθεί όλο και περισσότερο η γλώσσα Python στο σχολικό περιβάλλον. Η γλώσσα Pascal, που εδώ παρουσιάζεται δεύτερη, αποτέλεσε για πολλά χρόνια την πρώτη γλώσσα εκμάθησης βασικών αρχών προγραμματισμού στα εισαγωγικά μαθήματα σε Πανεπιστημιακά προγράμματα σπουδών, όμως σήμερα δεν συνδέεται με κάποιο σημαντικό τομέα ανάπτυξης λογισμικού για πραγματικές εφαρμογές και έτσι έχει σταδιακά εγκαταλειφθεί. Αντίθετα η γλώσσα Javascript, ειδικά με την προσθήκη ευέλικτων πακέτων όπως το «P5», έχει καταστεί πολύ εύχρηστη και διδακτικά χρήσιμη γλώσσα εισαγωγής στον προγραμματισμό.



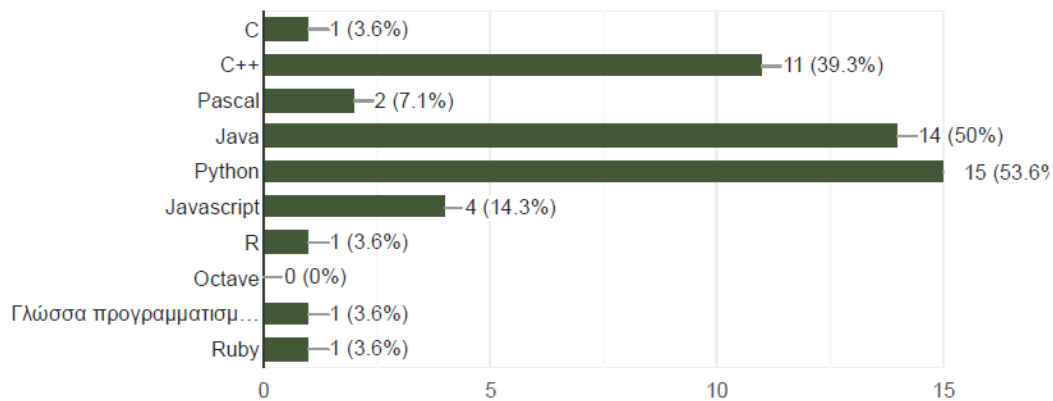
Σχήμα 1: Συνοπτική βαθμολόγηση γλωσσών προγραμματισμού στην ερώτηση «Ευκολία εκμάθησης».

Μια εξίσου χρήσιμη ερώτηση-κριτήριο αξιολόγησης γλωσσών προγραμματισμού είναι το κατά πόσο το συντακτικό και εν γένει η σχεδιάσή τους διευκολύνει τη διδασκαλία δομών δεδομένων, κυρίως επιπρόσθετα των πινάκων που κατά κανόνα υποστηρίζονται από όλες τις γλώσσες. Το κρίσιμο σχεδιαστικό ζήτημα είναι η ισορροπία μεταξύ απόλυτου ελέγχου διαχείρισης μνήμης σε χαμηλό επίπεδο, άρα μεγαλύτερου κινδύνου προγραμματιστικών σφαλμάτων όπως συμβαίνει με τις γλώσσες C και C++, και δυνατοτήτων δυναμικής διαχείρισης μνήμης αλλά με πλήρως αυτοματοποιημένο τρόπο, όπως συμβαίνει με τις γλώσσες Java και Python. Και σε αυτή την ερώτηση, όπως φαίνεται στο Σχήμα 2, η γλώσσα Python είναι με διαφορά η καταλληλότερη επιλογή σύμφωνα με τις απαντήσεις που συγκεντρώθηκαν.



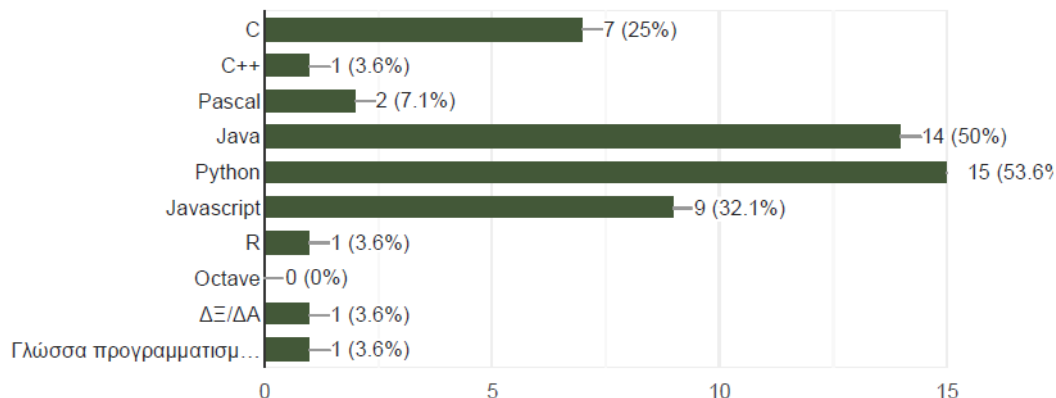
Σχήμα 2: Συνοπτική βαθμολόγηση γλωσσών προγραμματισμού στην ερώτηση «Δομές δεδομένων (κυρίως εκτός πινάκων)».

Το θέμα του αντικειμενοστραφούς προγραμματισμού (Object Oriented Programming – OOP) είναι μια σχετικά πιο πρόσφατη αλλά όλο και πιο σημαντική προσθήκη στα προγράμματα σπουδών στις χαμηλότερες και μέσες βαθμίδες εκπαίδευσης, ενώ για πολλά χρόνια αποτελεί βασικό συστατικό στα προγράμματα σπουδών στα Πανεπιστημιακά Τμήματα Πληροφορικής. Η διδασκαλία OOP αποτελεί συχνά πρόκληση για τον εκπαιδευτικό σε όλες τις βαθμίδες, καθώς πρέπει να συνδυάσει υψηλού επιπέδου έννοιες όπως η κληρονομικότητα (inheritance) και ο πολυμορφισμός (polymorphism) με την παρουσίαση του τρόπου που η εκάστοτε γλώσσα OOP τις υλοποιεί στο συντακτικό της. Συνεπώς, τα αποτελέσματα στο Σχήμα 3 είναι απόλυτα ερμηνεύσιμα με αυτή την οπτική, καθώς και οι τρεις γλώσσες που επιλέγονται ως καταλληλότερες υποστηρίζουν ισχυρές και καλώς ορισμένες δυνατότητες OOP. Συγκεκριμένα, η γλώσσα Python επισημαίνεται ως η πρώτη προτίμηση για εκπαιδευτικό περιβάλλον λόγω του ευκολότερου συντακτικού, ενώ οι γλώσσες Java και C++, που αποτελούν δύο από τις βασικότερες γλώσσες OOP για εφαρμογές μεγάλης κλίμακας και βιομηχανικής παραγωγής, περιλαμβάνουν πιο σύνθετες και αυστηρές δομές υλοποίησης OOP.



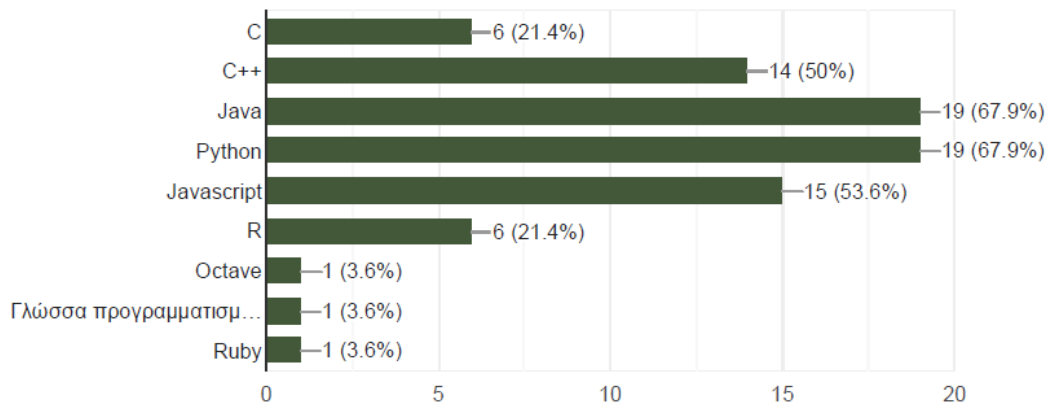
Σχήμα 3: Συνοπτική βαθμολόγηση γλωσσών προγραμματισμού στην ερώτηση «Αντικειμενοστραφής προγραμματισμός (OOP)».

Αν και η μεταφερσιμότητα (portability) των προγραμμάτων συχνά δεν θεωρείται μια από τις πρώτες προτεραιότητες στο εκπαιδευτικό περιβάλλον. Εντούτοις, η δυνατότητα ανάπτυξης και δοκιμής των απλών παραδειγμάτων-ασκήσεων προγραμματισμού σε οποιονδήποτε Η/Υ και μάλιστα με περιορισμένες απαιτήσεις σε υπολογιστικούς πόρους δείχνει να συνεισφέρει σημαντικά στην εκπαιδευτική διαδικασία. Το γεγονός αυτό επιβεβαιώνεται και από την αντίστοιχη ερώτηση βαρύτητας του κριτηρίου «μεταφερσιμότητα κώδικα (portability)» στην παρούσα μελέτη, όπου οι απαντήσεις δείχνουν σταθερά αυξητική τάση, από 14,3% στη βαρύτητα «1» μέχρι 28,6% στη βαρύτητα «5» (Γεωργίου, 2022). Στο Σχήμα 4 παρουσιάζεται η σύνοψη των αποτελεσμάτων στη συγκεκριμένη ερώτηση, όπου όπως αναμένεται ως καταλληλότερες επιλέγονται οι γλώσσες (κατά σειρά) Python και Java, ακολουθούμενες από Javascript και C.



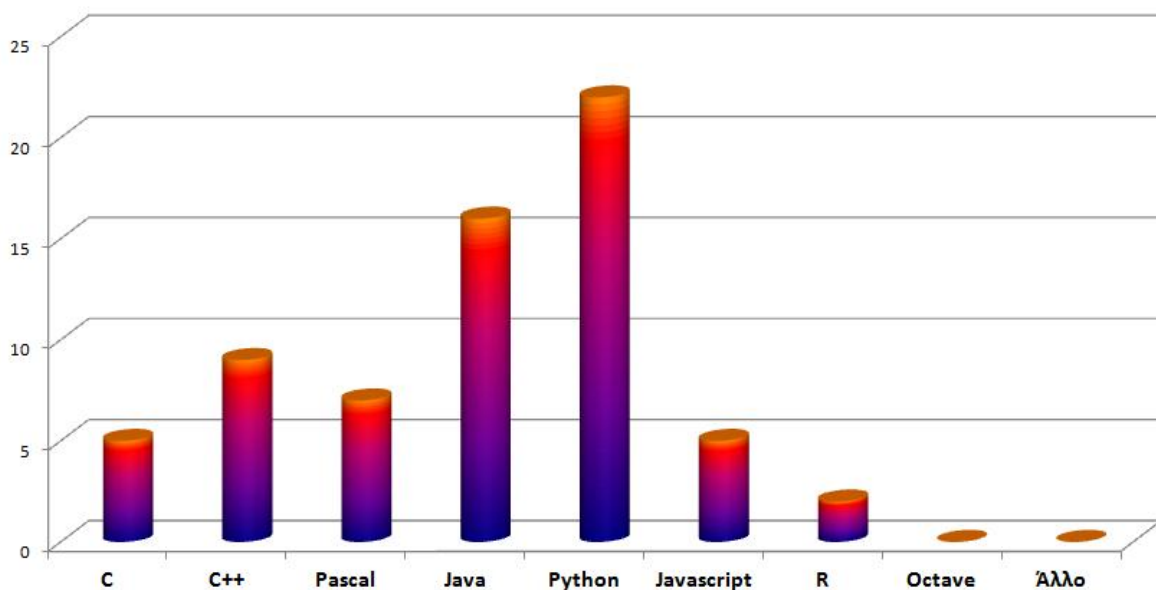
Σχήμα 4: Συνοπτική βαθμολόγηση γλωσσών προγραμματισμού στην ερώτηση «Μεταφερσιμότητα κώδικα (portability)».

Μία ακόμη ενδιαφέρουσα ερώτηση-κριτήριο αξιολόγησης γλωσσών προγραμματισμού στη διδασκαλία είναι η σχέση που έχουν, καθώς και η συνάφεια των γνώσεων που προσφέρουν, με τις αντίστοιχες επαγγελματικές δεξιότητες (e-Skills). Συγκεκριμένα, μία γλώσσα μπορεί να είναι κατάλληλη για εισαγωγή στον δομημένο προγραμματισμό (structured programming) αλλά με ελάχιστη ή καθόλου συσχέτιση με το σημερινό περιβάλλον ανάπτυξης λογισμικού, όπως είναι η περίπτωση της Pascal. Αντίθετα, μια γλώσσα μπορεί να είναι βασικό εργαλείο σύγχρονων εφαρμογών σήμερα αλλά η δομή και η σχεδιάσή της να μη διευκολύνει την εύκολη εισαγωγή στον προγραμματισμό ως πρώτη γλώσσα, όπως είναι η περίπτωση της C. Στο Σχήμα 5 παρουσιάζεται ακριβώς η δύσκολη αυτή ισορροπία, όπου φαίνεται να υπάρχουν διαφορετικές επιλογές κατά περίπτωση, όμως και πάλι οι γλώσσες Python και Java φαίνεται να υπερισχύουν ως καταλληλότερες επιλογές.



Σχήμα 5: Συνοπτική βαθμολόγηση γλωσσών προγραμματισμού στην ερώτηση «Σύνδεση με επαγγελματικές δεξιότητες».

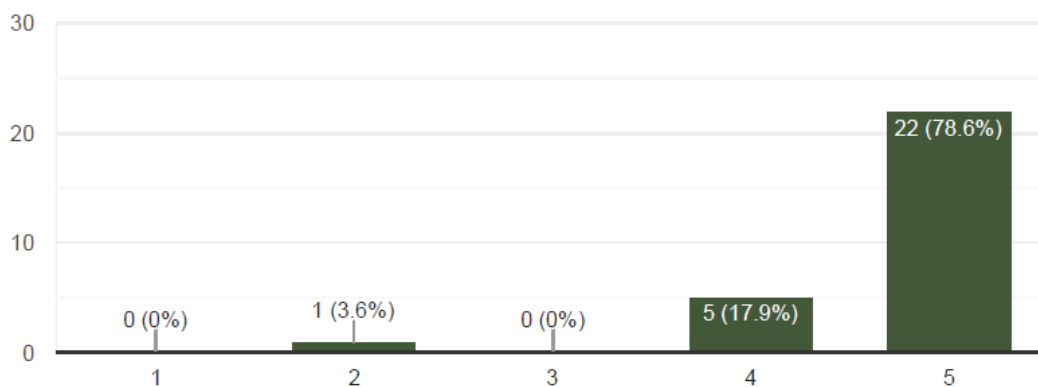
Συνοπτικά, στο Σχήμα 6 παρουσιάζεται η βαθμολόγηση των γλωσσών προγραμματισμού και για τις 22 ερωτήσεις-παράγοντες που εξετάστηκαν. Σε κάθε ερώτηση, οι τρεις πρώτες επιλογές βαθμολογούνται με ένα (1) και όλες οι υπόλοιπες με μηδέν (0). Κάθε ράβδος αντιστοιχεί στο αντίστοιχο άθροισμα ανά γλώσσα προγραμματισμού, δηλαδή πόσες φορές η συγκεκριμένη γλώσσα βρέθηκε μέσα στις πρώτες τρεις θέσεις αξιολόγησης στις απαντήσεις των συμμετεχόντων στην έρευνα. Από το διάγραμμα είναι φανερή η καθαρή «καθολική» επιλογή της Python έναντι κάθε άλλης, καθώς σε όλες τις ερωτήσεις-κριτήρια βρίσκεται εντός της πρώτης τριάδας των απαντήσεων (22/22). Ακολουθεί ως δεύτερη «καθολική» επιλογή η Java (16/22) και στην τρίτη θέση η C++ (9/22).



Σχήμα 6: Συνοπτική βαθμολόγηση γλωσσών προγραμματισμού αθροιστικά για τις 22 ερωτήσεις-παράγοντες που εξετάστηκαν. Κάθε ράβδος αντιστοιχεί στο πόσες φορές η συγκεκριμένη γλώσσα βρέθηκε στις πρώτες τρεις θέσεις αξιολόγησης στις απαντήσεις των συμμετεχόντων στην έρευνα.

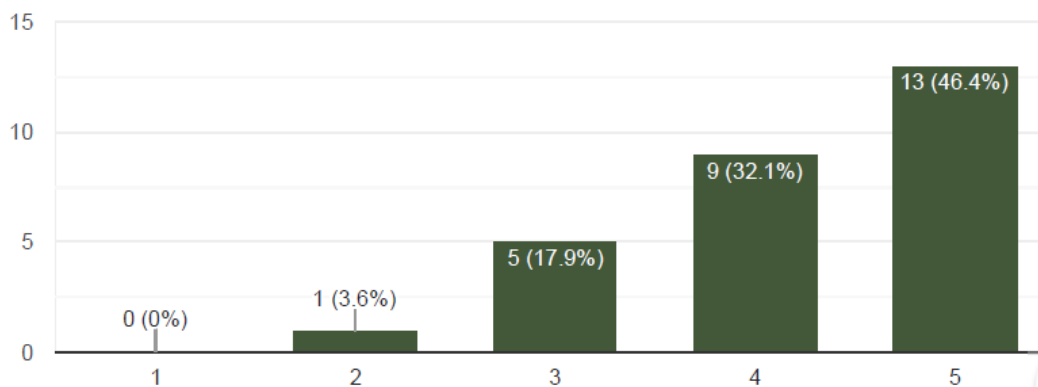
Σε σχέση με την αξιολόγηση της βαρύτητας (σημαντικότητας) κάθε παράγοντα αξιολόγησης, αντίστοιχα εδώ παρουσιάζονται μόνο μερικά ενδεικτικά αποτελέσματα που σχετίζονται με τα πιο ενδιαφέροντα συμπεράσματα. Όπως και για το μέρος (α) της μελέτης, η πλήρης αποτίμηση και όλα τα σχετικά συγκεντρωτικά δεδομένα και διαγράμματα είναι διαθέσιμα online (Γεωργίου, 2022).

Η ερώτηση-κριτήριο «ευκολία εκμάθησης», που όπως αναφέρθηκε παραπάνω είναι και η πρώτη στη λίστα του ερωτηματολογίου της έρευνας, παρουσιάζεται ως προς την αξιολόγηση της βαρύτητάς της στο Σχήμα 7. Όπως είναι αναμενόμενο, σε διδακτική διαδικασία εισαγωγής σε βασικές έννοιες προγραμματισμού, ειδικά σε σχολικό περιβάλλον χαμηλού ή μέσου επιπέδου εκπαίδευσης, η απλότητα και η προσβασιμότητα του αρχάριου χρήστη-προγραμματιστή στη δομή της γλώσσας προγραμματισμού αξιολογείται σχεδόν απόλυτα στον υψηλότερο βαθμό σημαντικότητας.



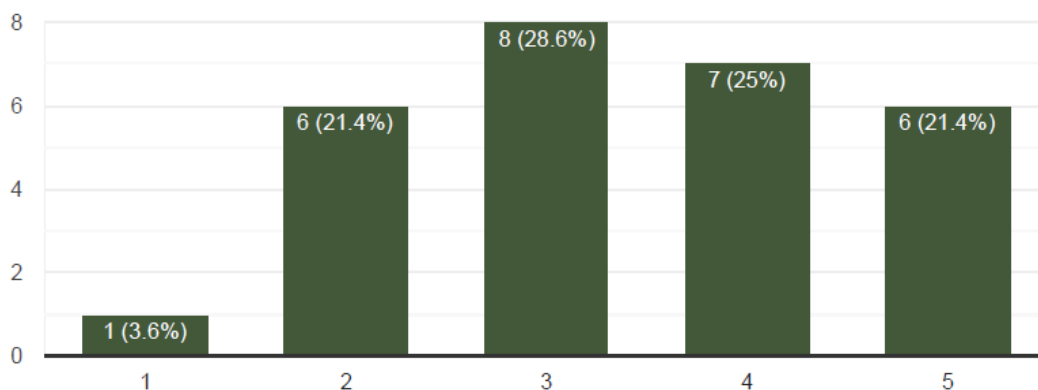
Σχήμα 7: Συνοπτική βαθμολόγηση συντελεστή βαρύτητας στην ερώτηση «Ευκολία εκμάθησης».

Ανάλογο είναι το αποτέλεσμα, αλλά με πιο σταδιακή κλιμάκωση της βαρύτητας, για την ερώτηση-κριτήριο «βασικό συντακτικό – keywords», όπως παρουσιάζεται στο Σχήμα 8. Η διαφοροποίηση εδώ κρίνεται πως οφείλεται κυρίως στο ότι ακόμα και μια γλώσσα προγραμματισμού με κάπως πιο σύνθετη δομή μπορεί να διδαχθεί επιτυχώς και με υψηλό βαθμό κατανόησης από τους μαθητές αν δοθεί ο απαραίτητος χρόνος και το κατάλληλο πλάνο διδασκαλίας της, πάντοτε φυσικά σε σχέση και με το επίπεδο εκπαίδευσης.



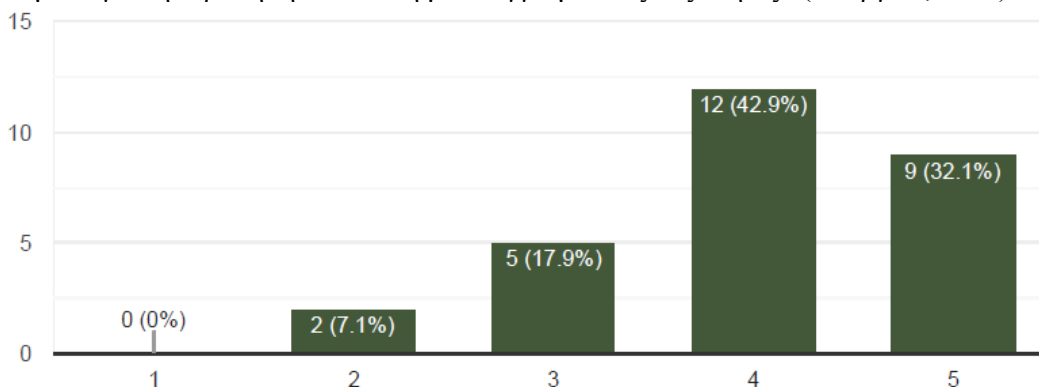
Σχήμα 8: Συνοπτική βαθμολόγηση συντελεστή βαρύτητας στην ερώτηση «Βασικό συντακτικό – keywords».

Ένα ζήτημα που συχνά γίνεται αντικείμενο συζήτησης σε σχέση με την αξία του στην εισαγωγική εκπαίδευση στον προγραμματισμό είναι το αν η γλώσσα που χρησιμοποιείται πρέπει ή όχι να επιβάλλει αυστηρό «έλεγχο τύπων (type checking)». Τα επιχειρήματα υπέρ αυτού είναι ότι εκπαιδεύει από νωρίς τους αρχάριους προγραμματιστές σε αυτό το πολύ σημαντικό δομικό εργαλείο, που ευνοεί την ανάπτυξη σωστά ορισμένων δομών δεδομένων και υψηλής ποιότητας κώδικα, με αποτελεσματικότερο έλεγχο αντίστοιχων σφαλμάτων. Τα επιχειρήματα κατά αυτού είναι ότι οι σύγχρονες γλώσσες προγραμματισμού είναι πολύ αποτελεσματικές στην αυτόματη διαχείριση τύπων δεδομένων εσωτερικά με διάφανο τρόπο, εξαιρετικά χρήσιμο για τον αρχάριο προγραμματιστή, ενώ για τους σύγχρονους Η/Υ η επιπλέον χρήση μνήμης για το σκοπό αυτό θεωρείται πλέον αμελητέα (Meyerovich, 2014). Στο Σχήμα 9 παρουσιάζονται τα συνοπτικά αποτελέσματα της αξιολόγησης του συγκεκριμένου κριτηρίου, όπου επιβεβαιώνεται χαρακτηριστικά η ευρεία αυτή διαφοροποίηση των απόψεων.



Σχήμα 9: Συνοπτική βαθμολόγηση συντελεστή βαρύτητας στην ερώτηση «Ελεγχος τύπων (type checking)».

Τέλος, μία ακόμη εξαιρετικά ενδιαφέρουσα αξιολόγηση της βαρύτητας είναι η ερώτηση-κριτήριο σε σχέση με τη «σύνδεση με ακαδημαϊκές σπουδές», τα αποτελέσματα της οποίας παρουσιάζονται στο Σχήμα 10. Όπως αναμενόταν, είναι φανερό ότι η συσχέτιση της διδασκαλίας εισαγωγικών εννοιών προγραμματισμού σε χαμηλό και μέσο επίπεδο εκπαίδευσης, ειδικότερα στο Γυμνάσιο και Λύκειο, με τα αντίστοιχα προγράμματα σπουδών σε υψηλό επίπεδο, δηλαδή στην Τριτοβάθμια εκπαίδευση, αξιολογείται ως πολύ σημαντική. Αξίζει να σημειωθεί ότι πανομοιότυπα είναι και τα αποτελέσματα για την ερώτηση «Σύνδεση με επαγγελματικές δεξιότητες» (Γεωργίου, 2022).



Σχήμα 10: Συνοπτική βαθμολόγηση συντελεστή βαρύτητας στην ερώτηση «Σύνδεση με ακαδημαϊκές σπουδές».

ΣΧΟΛΙΑΣΜΟΣ

Από την παρούσα έρευνα μπορούν να εξαχθούν δύο ομάδες συμπερασμάτων, αντίστοιχα με τα μέρη (α) και (β) του ερωτηματολογίου, όπως περιγράφηκε παραπάνω.

Σε σχέση με την αξιολόγηση των γλωσσών προγραμματισμού στο μέρος (α), επιβεβαιώνεται πως η κατάλληλη επιλογή είναι καθοριστική για την επιτυχημένη διδασκαλία και την εισαγωγή στις βασικές έννοιες του προγραμματισμού. Οι γλώσσες που επιλέχθηκαν φαίνεται πως σε μεγάλο ποσοστό κρίθηκαν τουλάχιστον σχετικές με αυτό το πεδίο εφαρμογής, ενώ κάποιες συγκέντρωσαν πολύ υψηλή συνολική βαθμολογία στην αξιολόγηση.

Η γλώσσα Python φαίνεται να είναι η ξεκάθαρη προτίμηση στην αξιολόγηση και συνεπώς στην επιλογή της ως πρώτη για τη διδασκαλία εισαγωγής στον προγραμματισμό σε σχολικό περιβάλλον. Πέρα από την «ευκολία εκμάθησης» (Σχήμα 1) και το «βασικό συντακτικό – keywords» (Γεωργίου, 2022), εμφανίζει μια σειρά ακόμη πλεονεκτήματα που αξιολογούνται ως σημαντικά στην εκπαιδευτική διαδικασία, τόσο σε εισαγωγικό επίπεδο όπως οι «δομές δεδομένων» και η «ανοχή και αντιμετώπιση σφαλμάτων κώδικα» (Γεωργίου, 2022), όσο και πιο σύνθετα ζητήματα και πιο εξειδικευμένες γνώσεις όπως «δικτυακός προγραμματισμός» και «τεχνητή νοημοσύνη – μηχανική μάθηση (AI)» (Γεωργίου, 2022).

Ως δεύτερη καταλληλότερη επιλογή εμφανίζεται η γλώσσα Java, με σημαντική όμως διαφορά από την πρώτη. Για τη Java τα πιο ισχυρά χαρακτηριστικά της που σχετίζονται με τη διδασκαλία φαίνεται να είναι ο αντικειμενοστραφής προγραμματισμός (Σχήμα 3) και φυσικά η μεταφερσιμότητα του κώδικα (Σχήμα 4) που αποτελεί εν γένει ένα από τα βασικότερα σχεδιαστικά πλεονεκτήματά της. Επιπλέον, εμφανίζεται πρώτη στην «ανάπτυξη γραφικού περιβάλλοντος (GUI)» και στις «διαθέσιμες

πλατφόρμες (IDEs/compiler)», ενώ ισοβαθεί με την Python στην πρώτη θέση στην ερώτηση-κριτήριο «σύνδεση με επαγγελματικές δεξιότητες» (Σχήμα 5).

Στην τρίτη θέση και αρκετά χαμηλότερα στη συνολική βαθμολογία (Σχήμα 6) εμφανίζεται η γλώσσα C++, η οποία τα τελευταία 15 περίπου χρόνια έχει διευρύνει το πεδίο εφαρμογής της, από εστιασμένη σε προγραμματισμό συστημάτων και βιομηχανική παραγωγή (όπως και η γλώσσα C) σε πραγματική γλώσσα γενικής χρήσης και υποστήριξης μεγάλου εύρους εφαρμογών, συμπεριλαμβανομένης και της εκπαίδευσης. Στην παρούσα έρευνα επιτυγχάνει πολύ υψηλή κατάταξη στην ερώτηση-κριτήριο «αντικειμενοστραφής προγραμματισμός (OOP)» (Σχήμα 3), στη «σύνδεση με επαγγελματικές δεξιότητες» (Σχήμα 5), καθώς επίσης φυσικά και στη «διαχείριση μνήμης», «πολυπρογραμματισμός (concurrency tools/safety)» και «απόδοση – ταχύτητα εκτέλεσης» (Γεωργίου, 2022).

Από τις υπόλοιπες γλώσσες προγραμματισμού, αξίζει να σημειωθεί η πρώτη θέση που καταλαμβάνει η Javascript στην ερώτηση-κριτήριο «εφαρμογές διαδικτύου (web apps)», καθώς και η πρώτη θέση της γλώσσας C στις ερωτήσεις-κριτήρια «προγραμματισμός μικροσυσκευών (SBC/MC)» και «απόδοση – ταχύτητα εκτέλεσης» (Γεωργίου 2022). Επισημαίνεται επίσης η δεύτερη θέση στις γλώσσας R στην ερώτηση-κριτήριο «γραφικές παραστάσεις (plots)», κάτι που φαίνεται να είναι εξαίρεση στον κανόνα, καθώς οι γλώσσες R και Octave εμφανίζονται σταθερά στο χαμηλότερο δεύτερο μισό όλων των υπολοίπων κατατάξεων στην παρούσα έρευνα. Μεταξύ των δύο, η γλώσσα R φαίνεται να υπερτερεί ξεκάθαρα ως εκπαιδευτική επιλογή με βάση τα αποτελέσματα (Γεωργίου, 2022).

Σε σχέση με την αξιολόγηση της βαρύτητας (σημαντικότητας) των κριτηρίων αξιολόγησης στο μέρος (β), επισημαίνεται και πάλι πως τα συγκεκριμένα αποτελέσματα είναι εν γένει ανεξάρτητα της λίστας των επιλεγμένων γλωσσών προγραμματισμού του ερωτηματολογίου, συνεπώς μπορούν να εφαρμοστούν στην αξιολόγηση οποιασδήποτε άλλης γλώσσας προγραμματισμού ως διδακτικό εργαλείο.

Όπως αναφέρθηκε και παραπάνω, οι ερωτήσεις-κριτήρια «ευκολία εκμάθησης» (Σχήμα 7) και «βασικό συντακτικό – keywords» (Σχήμα 8) αξιολογούνται ξεκάθαρα ως εξαιρετικής σημασίας στην εκπαιδευτική διαδικασία. Επιπλέον, η «ανοχή και αντιμετώπιση σφαλμάτων κώδικα» εμφανίζεται ως σημαντικός (βαρύτητα 4) ή πολύ σημαντικός (βαρύτητα 5) παράγοντας, φτάνοντας αθροιστικά το 75% των σχετικών απαντήσεων (Γεωργίου, 2022).

Η ερώτηση-κριτήριο που συγκεντρώνει σαφέστατα πολύ χαμηλή σημαντικότητα (βαρύτητα 1) είναι «πολυπρογραμματισμός (concurrency tools/safety)» (Γεωργίου, 2022), κάτι που επιβεβαιώνει πως η καταλληλότητα της εκάστοτε γλώσσας προγραμματισμού στη διδασκαλία κρίνεται πρωτίστως ως προς τους σκοπούς και το πρόγραμμα σπουδών, αντί των πιο «επαγγελματικών» δυνατοτήτων της γλώσσας σε πραγματικές εφαρμογές. Αντίστοιχα, προγραμματιστικές γνώσεις και δυνατότητες της γλώσσας που αφορούν πιο σύνθετα ζητήματα βελτιστοποίησης και απόδοσης του υπολογιστικού περιβάλλοντος εμφανίζονται πράγματι στη μελέτη ως μέτριας σημαντικότητας (βαρύτητα 3), όπως ο «έλεγχος τύπων (type checking)» (Σχήμα 9) και «διαχείριση μνήμης» (Γεωργίου, 2022).

Τόσο η «σύνδεση με επαγγελματικές δεξιότητες» όσο και η «σύνδεση με ακαδημαϊκές σπουδές» εμφανίζονται με υψηλό (βαρύτητα 4) ή πολύ υψηλό (βαρύτητα 5) βαθμό σημαντικότητας, φτάνοντας αθροιστικά σε κάθε περίπτωση 71,4% και 75% των σχετικών απαντήσεων, αντίστοιχα. Τέλος, με υψηλό (βαρύτητα 4) ή πολύ υψηλό (βαρύτητα 5) βαθμό σημαντικότητας εμφανίζονται αντίστοιχα οι «διαθέσιμες πλατφόρμες (IDEs/compiler)» και «διαθέσιμες πηγές – εκπαιδευτικό υλικό», αντίστοιχα, όπως φυσικά είναι αναμενόμενο.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα μελέτη παρουσιάστηκαν τα σημαντικότερα αποτελέσματα έρευνας που διεξήχθη με τη βοήθεια online ερωτηματολογίων, για την αξιολόγηση τόσο συγκεκριμένων γλωσσών προγραμματισμού, όσο και της βαρύτητας των ίδιων των κριτηρίων αξιολόγησής τους. Σύμφωνα με τα αποτελέσματα, τα μεν κριτήρια αξιολόγησης επιβεβαιώνονται ως προς τις αναμενόμενες προτεραιότητες της εκπαιδευτικής διαδικασίας σε μαθήματα βασικών εννοιών προγραμματισμού και αλγοριθμικής σκέψης, με έμφαση σε Γυμνάσια και Λύκεια, ενώ από τις γλώσσες προγραμματισμού η Python εμφανίζεται ξεκάθαρα ως η καταλληλότερη επιλογή.

Από τα αποτελέσματα της μελέτης κρίνεται σκόπιμη η άμεση αναπροσαρμογή του προγράμματος σπουδών στα σχολικά μαθήματα Πληροφορικής, ειδικότερα σε Γυμνάσια και Λύκεια, έτσι ώστε τόσο

η θεωρητική διδασκαλία με παραδείγματα κώδικα, όσο και η πρακτική εξάσκηση με συγγραφή απλών προγραμμάτων, να γίνεται στο εξής στη γλώσσα Python.

ΕΥΧΑΡΙΣΤΙΕΣ

Ο συγγραφέας ευχαριστεί θερμά όλους τους συμμετέχοντες στην έρευνα, με την ενδελεχή και έγκαιρη συμπλήρωση του online ερωτηματολογίου. Παρά το μικρό σχετικά χρονικό διάστημα διεξαγωγής της έρευνας, η συμμετοχή τους ήταν καθοριστική για την καταγραφή στατιστικά σημαντικού και αντιπροσωπευτικού συνόλου απαντήσεων.

ΑΝΑΦΟΡΕΣ

- Γεωργίου Χάρης, (2022). Αξιολόγηση γλωσσών προγραμματισμού για σχολική διδασκαλία (αποτελέσματα έρευνας). *Zenodo*, 10/8/2022, <https://doi.org/10.5281/zenodo.6979256>
- Υπουργείο Παιδείας, Έρευνας και Θρησκευμάτων – Ινστιτούτο Εκπαιδευτικής Πολιτικής. (2017). Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον – Βιβλίο Καθηγητή. *Ινστιτούτο Τεχνολογίας Υπολογιστών και Εκδόσεων «Διόφαντος»*.
- Bell, Tim. (2018). Computer Science in K-12 Education: The Big Picture. *Olympiads in Informatics*, 2018, Vol. 12, 3–11.
- Kalas, I., Mittermeier, R. (Eds.). (2011). Informatics in Schools Contributing to 21st Century Education. *In: 5th International Conference on Informatics in Schools: Situation, Evolution and Perspectives, (ISSEP 2011)*. Springer LNCS.
- Kanemune, S., Shirai, S., Tani, S. (2022). Informatics and Programming Education at Primary and Secondary Schools in Japan. *Technical report*. Osaka Electro-Communication University.
- Meyerovich, L., Rabkin, A. (2014). The Sociology of Programming Languages. *2014 StrangeLoop Conference*, 17/4/2022, <https://www.youtube.com/watch?v=M4Gsl8sVgdQ>