

ΗΜΥ01Κ06
Επιστημονικός Προγραμματισμός με Python



6+1 ερωτήσεις επανάληψης για τη Διάλεξη 01
Προβλήματα / Αλγόριθμοι / Γλώσσες Προγραμματισμού

Φθινόπωρο 2026

Ερώτηση 1/7

Πώς ονομάζεται μια *υπολογιστική* διαδικασία που έχει όλες τις βασικές ιδιότητες ενός αλγορίθμου εκτός από την ιδιότητα **Finiteness** (χρήση πεπερασμένων πόρων);

Βασικές Ιδιότητες ενός Αλγορίθμου

Finiteness Χρήση πεπερασμένων πόρων

- Οι ανάγκες ενός αλγορίθμου σε *χώρο* και *χρόνο* πρέπει να είναι πεπερασμένες

Ένας αλγόριθμος πρέπει να υλοποιείται μέσα σε *πεπερασμένο χώρο* και να τερματίζει μετά από *πεπερασμένο αριθμό βημάτων*.

- Υπολογιστικές διαδικασίες που *στερούνται* αυτή την ιδιότητα – *ενώ έχουν όλες τις υπόλοιπες*- ονομάζονται **Υπολογιστικές Μέθοδοι** (*computational methods*)

Θυμόμαστε τις βασικές ιδιότητες ενός αλγορίθμου

- **Finiteness** Χρήση πεπερασμένων πόρων
- **Definiteness** Σαφήνεια / Καθοριστικότητα
- **Effectiveness** Εφικτότητα / Αποτελεσματικότητα
- **Input / Output** Είσοδος / Έξοδος

Ερώτηση 2/7

Ποιες είναι οι τέσσερις εναλλακτικές μορφές διατύπωσης αλγορίθμων;

Εναλλακτικές μορφές διατύπωσης αλγορίθμων

Φυσική γλώσσα

Ο αλγόριθμος εκφράζεται σε **απλή καθομιλούμενη γλώσσα**, στην οποία οι προτάσεις έχουν χωριστεί σε **παραγράφους-βήματα** και έχουν αριθμηθεί.

Τύπος

Ο αλγόριθμος εκφράζεται με τη βοήθεια ενός (ή συνδυασμού περισσότερων) **μαθηματικών τύπων**.

Ψευδοκώδικας

Ο αλγόριθμος εκφράζεται σε μια **υποθετική γλώσσα** με στοιχεία από κάποιες γλώσσες προγραμματισμού, **παραλείποντας λεπτομέρειες** που δεν είναι ουσιαστικές για την ανθρώπινη κατανόηση του αλγορίθμου.

Διαγράμματα ροής

Ο αλγόριθμος εκφράζεται **γραφικά** με μια σειρά σχημάτων που συνδέονται μεταξύ τους με συγκεκριμένο τρόπο. Τα **διαφορετικά περιγράμματα** των σχημάτων υποδηλώνουν **διαφορετικά είδη εντολών / πράξεων**, ενώ ο **τρόπος διασύνδεσης** υποδηλώνει την **σειρά εκτέλεσης** των συγκεκριμένων εντολών / πράξεων.

Ερώτηση 3/7

Τι είναι, που χρησιμοποιούνται, και από τι χαρακτηρίζονται οι γλώσσες προγραμματισμού υψηλού επιπέδου;

Γλώσσες προγραμματισμού υψηλού επιπέδου

- **Τεχνητές** γλώσσες που μπορούν να χρησιμοποιηθούν για τον έλεγχο μιας μηχανής, συνήθως ενός υπολογιστή
- Χρησιμοποιούνται για να διευκολύνουν την οργάνωση και διαχείριση πληροφοριών αλλά και την **ακριβή διατύπωση αλγορίθμων**
- Ακολουθούν τις **βασικές έννοιες και αρχές της γλωσσολογίας** και προσεγγίζουν πολύ περισσότερο τον ανθρώπινο τρόπο σκέψης και διατύπωσης
- Χαρακτηρίζονται από ένα σύνολο **συντακτικών** και **εννοιολογικών** κανόνων, που ορίζουν αντίστοιχα τη **δομή** και το **νόημα** των **προτάσεων** της γλώσσας

Ερώτηση 4/7

Ποια είναι τα τέσσερα στοιχεία που χαρακτηρίζουν μια γλώσσα προγραμματισμού υψηλού επιπέδου;

Μια γλώσσα προγραμματισμού υψηλού επιπέδου προσδιορίζεται από 4 στοιχεία:

Αλφάβητο - Λεξιλόγιο - Συντακτικό - Σημασιολογία

Αλφάβητο. Το σύνολο των *αποδεκτών συμβόλων* - δηλαδή εκείνων που *αναγνωρίζει* και *κατανοεί* η συγκεκριμένη γλώσσα.

A...Z a...z + - * / = < > ! @ # \$ % ^ & ?

Λεξιλόγιο Το σύνολο των *αποδεκτών συνδυασμών των συμβόλων* της γλώσσας δηλαδή των *λέξεων* που αναγνωρίζει και κατανοεί η συγκεκριμένη γλώσσα.

Οι περισσότερες από αυτές τις λέξεις είναι συνήθως Αγγλικά ρήματα στην προστακτική τους (εντολές της γλώσσας).

read, print, if, else, for, pause...

Συντακτικό (syntax) Το σύνολο των *κανόνων* που διέπουν την *σύνταξη* κειμένων (προγραμμάτων) στην συγκεκριμένη γλώσσα (συντακτικοί κανόνες).

Αφορούν την ορθή σύνταξη των φράσεων και την φυσική διάταξη των συμβόλων της γλώσσας.

var <όνομα μεταβλητής> *as* <τύπος δεδομένων> ;

if <συνθήκη> *then* <έκφραση 1> *else* <έκφραση 2>

Σημασιολογία (semantics) Προσδιορίζει την *έννοια των εκφράσεων* δηλαδή το *τι ακριβώς σημαίνει η κάθε έκφραση* όταν το πρόγραμμα εκτελεστεί σε κάποιον υπολογιστή.

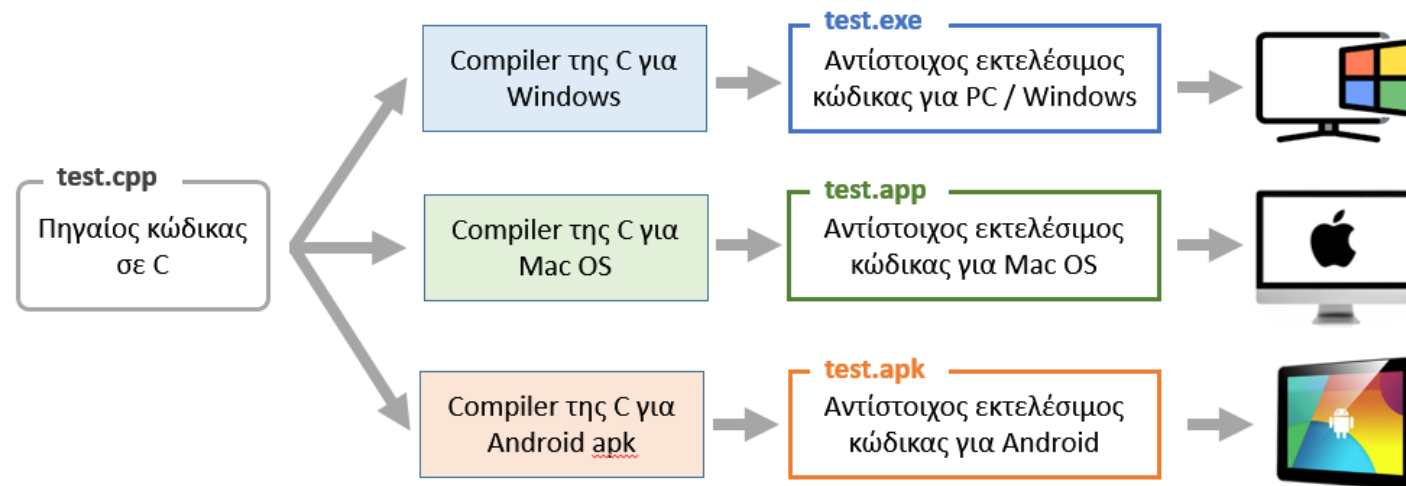
Η έκφραση *var Test as integer*; σημαίνει ότι κατά την εκτέλεση του προγράμματος θα πρέπει να διατεθεί ένα τμήμα της κεντρικής μνήμης του υπολογιστή για την αποθήκευση μιας ακέραιας μεταβλητής που θα ονομάζεται *Test*.

Ερώτηση 5/7

Πως μπορούμε να εκτελέσουμε ένα πρόγραμμα που γράψαμε σε διαφορετικούς τύπους υπολογιστών (π.χ. Windows και Mac OS);

Διαφορετικοί compilers για την ίδια γλώσσα προγραμματισμού

Από τον *ίδιο^(*) πηγαίο κώδικα* μπορούμε να δημιουργήσουμε *διαφορετικά εκτελέσιμα προγράμματα* για διαφορετικούς τύπους ηλεκτρονικών υπολογιστών



() ανάλογα με τον τύπο του υπολογιστή ενδέχεται να υπάρχουν μικροδιαφορές στον πηγαίο κώδικα*

Ερώτηση 6/7

Μπορούμε να ξεκινήσουμε να εκτελούμε ένα πρόγραμμα ακόμα και αν έχει συντακτικά λάθη;

Η έννοια του Interpreter (Διερμηνέα)

Interpreter **συγκεκριμένης** γλώσσας προγραμματισμού για **συγκεκριμένο** τύπο Η/Υ

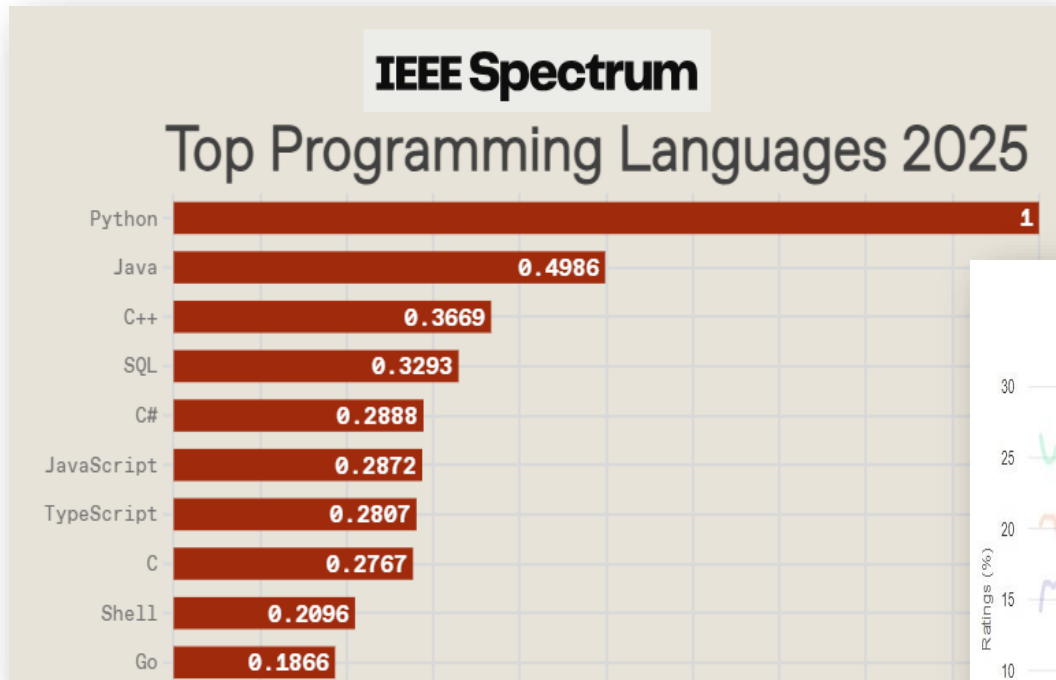
- Ειδικό πρόγραμμα που μεταγλωττίζει / μετατρέπει μία-μία τις εντολές μιας γλώσσας προγραμματισμού υψηλού επιπέδου σε γλώσσα μηχανής και στη συνέχεια τις εκτελεί. *Αν ο interpreter συναντήσει μια εντολή που δεν είναι συντακτικά σωστή, τότε η εκτέλεση του προγράμματος διακόπτεται*
- Τόσο ο compiler όσο και ο interpreter κάνουν βασικά την ίδια δουλειά, δηλαδή τη μετατροπή της γλώσσας προγραμματισμού υψηλού επιπέδου σε γλώσσα μηχανής
- Ωστόσο, ένας **compiler** μετατρέπει συνολικά τον κώδικα σε γλώσσα μηχανής (δημιουργία αρχείου .exe, .app ή .apk ανάλογα με το είδος του υπολογιστή) πριν από την εκτέλεσή του, ενώ ένας **interpreter** μετατρέπει μία-μία τις εντολές σε γλώσσα μηχανής και στη συνέχεια τις εκτελεί



Ναι, εφόσον για την μεταγλώττισή του χρησιμοποιήσουμε interpreter αντί για compiler

Bonus Ερώτηση 7/7

Ποιες είναι οι 10 πρώτες σε χρήση γλώσσες προγραμματισμού σήμερα;
(Οκτώβριος 2025)



www.tiobe.com/tiobe-index/

