

Αρχές Γλωσσών Προγραμματισμού & Μεταφραστών

Τμήμα Μηχανικών Η/Υ & Πληροφορικής

Πανεπιστήμιο Πατρών

Εαρινό Εξάμηνο 2026

Διδάσκοντες: Ι. Γαροφαλάκης, Σ. Σιούτας, Π. Χατζηδούκας, Ι. Βασιλόπουλος, Γ. Βονιτσάνος, Ι. Σίμου

Εργαστηριακή Άσκηση

Σκοπός της εργαστηριακής άσκησης είναι να εξοικειωθείτε με την περιγραφή μιας γλώσσας σε μορφή BNF, τις βασικές έννοιες των μεταγλωττιστών, και τελικά να υλοποιήσετε δύο από τα τμήματα ενός μεταγλωττιστή, έναν συντακτικό και έναν λεκτικό αναλυτή με τη χρήση των εργαλείων Bison και Flex.

ΔΙΑΔΙΚΑΣΤΙΚΑ**Παραδοτέα**

- ❖ **Γραπτή Αναφορά** που περιλαμβάνει:
 - Τα στοιχεία των μελών της ομάδας (Όνοματεπώνυμο, ΑΜ, Έτος, e-mail)
 - Την περιγραφή της γραμματικής της γλώσσας σε BNF.
 - Τα **ΤΕΛΙΚΑ** αρχεία περιγραφής της γλώσσας, τα οποία δίνονται ως **είσοδος** στα Flex και Bison
 - Screenshots παραδειγμάτων εφαρμογής (τουλάχιστον ένα παράδειγμα για κάθε ερώτημα, για επιτυχημένη και ανεπιτυχή ανάλυση)
 - Σχόλια - Παραδοχές που έγιναν για την ανάπτυξη της εργασίας
- ❖ Συμπιεσμένα σε **ένα αρχείο (zip)** τα:
 - Την παραπάνω γραπτή αναφορά.
 - Τα **ΤΕΛΙΚΑ** αρχεία εισόδου για το flex και το bison.
 - Τον **ΤΕΛΙΚΟ** κώδικα σε C (και το .h αρχείο) που παράγεται από τα δύο προγράμματα και ο εκτελέσιμος κώδικας του parser.
 - Τα **δοκιμαστικά αρχεία** που δόθηκαν σαν είσοδο στον parser για να ελεγχθεί η σωστή λειτουργία του ([δείτε την επισήμανση στο τέλος της εκφώνησης](#)).

Το αρχείο zip πρέπει να έχει όνομα τους αριθμούς μητρώου των ατόμων της ομάδας διαχωρισμένους με _ και από τον μικρότερο στο μεγαλύτερο (π.χ. 1000_1543_2788.zip), και να υποβληθεί (ΥΠΟΧΡΕΩΤΙΚΑ) στο eclass του μαθήματος και στο μήνυμα να αναφέρονται τα ονοματεπώνυμα, το έτος και οι αντίστοιχοι αριθμοί μητρώου των μελών της ομάδας, καθώς και οι e-mail διευθύνσεις όλων των μελών της ομάδας

Διευκρινίσεις

- ❖ Για τη λειτουργία των εργαλείων Flex και Bison μπορείτε να βρείτε πληροφορίες στο eclass του μαθήματος.
- ❖ Η άσκηση θα γίνει σε ομάδες από 1 έως 3 άτομα.
- ❖ Η συμβολή της στον τελικό βαθμό είναι 20%, βάσει της νέας διαδικασίας αξιολόγησης, ή 30%, βάσει της παλαιάς διαδικασίας αξιολόγησης. **Οι φοιτητές/φοιτήτριες του 2ου έτους και του 3ου έτους ακολουθούν τη νέα διαδικασία αξιολόγησης.**
- ❖ Η εργαστηριακή άσκηση είναι υποχρεωτική.
- ❖ Συμβουλευτείτε τα [Διαδικαστικά](#) του μαθήματος.
- ❖ Οι τελικές ημερομηνίες παράδοσης του project Flex/Bison είναι 1) στις **31 Μαΐου 2026** για την εξεταστική Ιουνίου 2026 και 2) στις **30 Αυγούστου 2026** για την εξεταστική Σεπτεμβρίου. Οι προφορικές παρουσιάσεις θα ξεκινήσουν τις εβδομάδες που ακολουθούν κάθε προθεσμία.
- ❖ Τυχόν απορίες ή υποδείξεις αναρτώνται **ΑΠΟΚΛΕΙΣΤΙΚΑ** στο χώρο “Συζητήσεις” στην σελίδα του μαθήματος στο eclass

(<https://eclass.upatras.gr/modules/forum/?course=CEID1091>)

ΠΕΡΙΓΡΑΦΗ ΑΣΚΗΣΗΣ

Η **mySQLq** είναι μια ψευδογλώσσα ερωτημάτων, η οποία ακολουθεί τις βασικές προδιαγραφές της SQL, με σκοπό τη συγγραφή και τον έλεγχο ερωτημάτων πάνω σε σχεσιακά δεδομένα. Η **mySQLq** αποτελείται από **εντολές** (statements), οι οποίες με τη σειρά τους αποτελούνται από **λέξεις-κλειδιά** (keywords), **αναγνωριστικά** (identifiers), **τελεστές** (operators), **κυριολεκτικά** (literals) και **ειδικούς χαρακτήρες** όπως . , ; ().

Κάθε εντολή στην **mySQLq** περιγράφει μια λειτουργία πάνω σε δεδομένα, όπως για παράδειγμα ανάκτηση δεδομένων από έναν πίνακα, φιλτράρισμα, ή ομαδοποίηση. Μια τυπική εντολή ανάκτησης δεδομένων στην **mySQLq** έχει τη μορφή:

```
SELECT <στήλες>  
FROM <όνομα_πίνακα>  
WHERE <συνθήκη>;
```

όπου:

- το **SELECT** καθορίζει ποια πεδία/στήλες του πίνακα θέλουμε να ανακτήσουμε,
- το **FROM** καθορίζει από ποιον πίνακα θέλουμε να πάρουμε τα δεδομένα,
- το **WHERE** καθορίζει την συνθήκη που θα πρέπει να ικανοποιούν τα δεδομένα που θα ανακτηθούν,
- το ; ορίζει το τέλος της εντολής.
- Τα < > στις περιγραφές των γενικών μορφών (παραπάνω και στην υπόλοιπη εκφώνηση) των εντολών και των όρων δεν αποτελούν μέρος της γλώσσας και χρησιμοποιούνται απλώς για ευκρινή ορισμό των ορισμάτων.

Η **mySQLq** είναι αυστηρή ως προς τη σύνταξη, καθώς κάθε εντολή θα πρέπει να είναι πλήρης, βάσει των προδιαγραφών της γραμματικής που δίνονται ακολούθως, και να τερματίζει πάντοτε με ;. Επιπλέον, εάν υπάρχουν παρενθέσεις αυτές θα πρέπει να είναι σωστά ζευγοποιημένες και οι λίστες (π.χ. για την ανάκτηση πολλαπλών στηλών) να χωρίζονται ορθά με κόμματα.

Οι **λέξεις-κλειδιά** (keywords) της **mySQLq** είναι **case-insensitive**. Για παράδειγμα, τα select, SELECT, SeLeCt κ.λπ. είναι ισοδύναμα. Οι λέξεις-κλειδιά της γλώσσας είναι: **select, from, where, limit, group, order, by, in, and, or, not, create, table, int, float, varchar**.

Αντίθετα, τα **αναγνωριστικά** (identifiers) είναι **case-sensitive**. Για παράδειγμα, ένας πίνακας με όνομα Askisi είναι διαφορετικός από έναν πίνακα με όνομα askisi, ή μια στήλη με όνομα age είναι διαφορετική από μια στήλη με όνομα AGE.

Η συνθήκη που ακολουθεί το keyword **WHERE** μπορεί να είναι μια απλή λογική έκφραση ή συνδυασμός περισσότερων λογικών εκφράσεων. Οι λογικοί τελεστές που υποστηρίζει η γλώσσα **mySQLq** είναι οι **=**, **!=**, **>**, **<**, **>=**, **<=**, **AND**, **OR**, **NOT**. Επιπλέον αυτών, υποστηρίζεται και ο τελεστής **IN**, ο οποίος ελέγχει εάν μια τιμή βρίσκεται ή όχι εντός μιας λίστας.

Για παράδειγμα:

```
... WHERE id IN ('id1', 'id2', 'id3');  
... WHERE id NOT IN ('id1', 'id2', 'id3');  
... WHERE age >= 18;
```

Τα **σχόλια** στη **mySQLq** είναι δυο μορφών:

- -- σχόλιο μονής γραμμής (δηλαδή αναγνωρίζεται ό,τι ακολουθεί τις δυο παύλες μέχρι το τέλος της γραμμής)
- /* σχόλιο

πολλαπλών

γραμμών */

Τα σχόλια επιτρέπεται να εμφανιστούν ανάμεσα σε tokens οποιασδήποτε εντολής, αλλά δεν επιτρέπεται να σπάνε ένα token (π.χ. το `Se/*μπλα μπλα*/lect` δεν είναι αποδεκτό).

Η γλώσσα υποστηρίζει **αναγνωριστικά** (ονόματα πινάκων, στηλών) που ακολουθούν τους τυπικούς κανόνες ονοματοδοσίας: 1) Ξεκινούν από γράμμα ή `_`, 2) Συνεχίζουν με γράμματα, νούμερα ή `_`.

Η γλώσσα υποστηρίζει **κυριολεκτικά** (literals):

- ακεραίους (π.χ. 1, 3, -512)
- πραγματικούς (π.χ. 3.14, -0.5)
- αλφαριθμητικά που περιέχονται σε μονά εισαγωγικά (π.χ. 'this is my text').

Εάν δεν αναφέρεται ρητώς κάτι διαφορετικό, οι χαρακτήρες λευκού διαστήματος (whitespaces) και οι αλλαγές γραμμής δεν αποτελούν μέρος της γλώσσας και δεν παίζουν ρόλο στη σύνταξη πέραν του διαχωρισμού των tokens.

Προδιαγραφές γραμματικής

- Κάθε αρχείο **mysql** περιέχει μια ή περισσότερες εντολές (statements). Κάθε εντολή τερματίζει υποχρεωτικά με το σύμβολο ;.
- Οι επιτρεπτές εντολές/statements είναι τύπου **CREATE TABLE** ή **SELECT**. Τα statements αυτά μπορούν να εμφανιστούν με οποιαδήποτε σειρά μέσα στο αρχείο.
- Η εντολή **CREATE TABLE** χρησιμοποιείται για τη δημιουργία ενός πίνακα στη βάση δεδομένων. Η γενική της μορφή είναι:

```
CREATE TABLE <όνομα_πίνακα> (  
    <λίστα_στηλών_πίνακα>  
);
```

όπου το <όνομα_πίνακα> είναι ένα αναγνωριστικό, η <λίστα_στηλών_πίνακα> περικλείεται υποχρεωτικά από παρενθέσεις και οι διαφορετικές στήλες χωρίζονται με κόμμα. Μετά την εμφάνιση της τελευταίας στήλης, δηλαδή πριν το κλείσιμο της παρένθεσης); δεν υπάρχει κόμμα.

Η <λίστα_στηλών_πίνακα> ακολουθεί την εξής μορφή:

```
<όνομα_στήλης> <τύπος> ,
```

όπου το <όνομα_στήλης> είναι ένα αναγνωριστικό και ο <τύπος> ένας από τους επιτρεπτούς τύπους δεδομένων: **int**, **float**, **varchar(n)**. Για παράδειγμα: ID **int**, LastName **varchar(255)**, weight **float**. Η τιμή της μεταβλητής n του τύπου **varchar** είναι αυστηρά ακέραιη και θετική.

- Η εντολή **SELECT** χρησιμοποιείται για την ανάκτηση δεδομένων από έναν ή περισσότερους πίνακες. Η γενική μορφή της είναι:

```
SELECT <λίστα_στηλών>  
FROM <όνομα_πίνακα>  
[WHERE <συνθήκη>]  
[GROUP BY <λίστα_στηλών>]  
[ORDER BY <λίστα_στηλών>]  
[LIMIT <n>];
```

Επισήμανση: Οι αγκύλες [...] παραπάνω φανερώνουν ότι είναι προαιρετική η εμφάνιση των όρων που περιέχουν, δηλαδή υποχρεωτικοί όροι είναι μόνο το **select** και το **from**, ενώ οι υπόλοιποι εμφανίζονται προαιρετικά. Δεν αποτελούν μέρος της γλώσσας οι [].

Η σειρά εμφάνισης των όρων σε μια εντολή **select** είναι αυστηρή και ακολουθεί πάντα τη σειρά: **select** -> **from** -> **where** -> **group by** -> **order by** -> **limit**.

Εμφάνιση των όρων σε διαφορετική σειρά θα πρέπει να αναγνωρίζεται ως λάθος.

- Η <λίστα_στηλών> που ακολουθεί το **SELECT** μπορεί να είναι είτε ο ειδικός χαρακτήρας *, είτε μια λίστα από αναγνωριστικά στηλών, χωρισμένα με κόμμα. Για παράδειγμα:

SELECT * ...

SELECT col1, col2, col3 ...

- Ο όρος **FROM** έχει την ακόλουθη γενική μορφή:

FROM <όνομα_πίνακα>

όπου <όνομα_πίνακα> είναι το όνομα του πίνακα της βάσης από τον οποίο θέλουμε να ανακτήσουμε δεδομένα.

- Ο όρος **WHERE** είναι προαιρετικός και ακολουθεί τη γενική μορφή:

WHERE <συνθήκη>

όπου η μορφή της συνθήκης έχει οριστεί στις προδιαγραφές της άσκησης παραπάνω.

- Ο όρος **GROUP BY** είναι προαιρετικός και ακολουθεί τη γενική μορφή:

GROUP BY <λίστα_στηλών>

όπου η <λίστα_στηλών> μπορεί να είναι ένα ή περισσότερα ονόματα στηλών χωρισμένα με κόμμα.

- Ο όρος **ORDER BY** είναι προαιρετικός και ακολουθεί τη γενική μορφή:

ORDER BY <λίστα_στηλών>

όπου η <λίστα_στηλών> μπορεί να είναι ένα ή περισσότερα ονόματα στηλών χωρισμένα με κόμμα.

- Ο όρος **LIMIT** είναι προαιρετικός και ακολουθεί τη γενική μορφή:

LIMIT <n>

όπου <n> είναι ένας αυστηρά θετικός ακέραιος αριθμός.

Ερωτήματα

1. (60%)

- a. Δώστε σε BNF το συντακτικό ορισμό της γραμματικής της γλώσσας.
- b. Χρησιμοποιώντας τα προγράμματα Flex και Bison, υλοποιήστε έναν λεξικό και συντακτικό αναλυτή, ο οποίος θα παίρνει ως είσοδο ένα αρχείο γραμμένο στη ψευδογλώσσα που περιγράφηκε παραπάνω και θα ελέγχει σε ένα πέρασμα αν το πρόγραμμα είναι συντακτικά ορθό. Το πρόγραμμά σας θα καλείται από τη γραμμή εντολών ως εξής:

prompt> myParser.exe file_name

και 1) θα επιστρέφει υποχρεωτικά το ίδιο το πρόγραμμα στην οθόνη και ένα διαγνωστικό μήνυμα για το αν ήταν ορθώς γραμμένο, ή 2) κατάλληλο μήνυμα σφάλματος (πρέπει να φαίνεται η γραμμή όπου υπάρχει το σφάλμα και η εκτύπωση του προγράμματος στην οθόνη και η ανάλυση να διακόπτονται σε εκείνη τη γραμμή).

2. (15%)

Τροποποιήστε τον αναλυτή σας ώστε να εκτελεί τους εξής ελέγχους και να εμφανίζει σχετικά διαγνωστικά μηνύματα σε περίπτωση σφάλματος:

- a. Τα ονόματα των πινάκων που δημιουργούνται με την εντολή **CREATE** πρέπει να είναι μοναδικά.
- b. Ένας πίνακας που εμφανίζεται στον όρο **FROM** θα πρέπει να έχει δημιουργηθεί προηγουμένως με μια εντολή **CREATE**.
- c. Τα ονόματα των στηλών ενός πίνακα πρέπει να έχουν οριστεί στην εντολή **CREATE** του συγκεκριμένου πίνακα και είναι διαφορετικά μεταξύ τους. Ωστόσο, σε διαφορετικούς πίνακες μπορούν να υπάρχουν στήλες με το ίδιο όνομα.
- d. Οι στήλες που εμφανίζονται στους όρους **SELECT**, **WHERE**, **GROUP BY**, **ORDER BY** θα πρέπει να υπάρχουν στον πίνακα από τον οποίο γίνεται ανάκτηση δεδομένων.
- e. Στον όρο **WHERE**:
 - i. αν γίνεται σύγκριση κάποιας στήλης με κάποιο κυριολεκτικό, θα πρέπει να εξασφαλίζεται ότι ο τύπος δεδομένων της στήλης είναι συμβατός με τον τύπο δεδομένων του κυριολεκτικού.
 - ii. οι επιτρεπτές συγκρίσεις είναι INT με ακέραιο κυριολεκτικό, FLOAT με ακέραιο ή πραγματικό κυριολεκτικό, VARCHAR(n) με αλφαριθμητικό κυριολεκτικό.
 - iii. στην περίπτωση χρήσης του τελεστή (NOT) IN, ο έλεγχος θα επιτρέπεται μόνο όταν ο τύπος δεδομένων της στήλης είναι συμβατός με κάθε τύπο δεδομένων των κυριολεκτικών που περιέχονται στις παρενθέσεις.

3. (25%)

Επεκτείνετε τον αναλυτή σας ώστε να καλύπτει τα ακόλουθα:

- a. Η εντολή **SELECT** υποστηρίζει τη χρήση του όρου **JOIN**. Ο όρος **JOIN** χρησιμοποιείται για να ενωθούν δυο πίνακες, από τους οποίους θέλουμε να ανακτήσουμε δεδομένα από κοινού. Η εντολή **SELECT** ακολουθεί πλέον τη γενική μορφή:

```
SELECT <λίστα_στηλών>  
FROM <όνομα_πίνακα1>  
[JOIN <όνομα_πίνακα2> ON <όνομα_πίνακα1>.<όνομα_στήλης> =  
<όνομα_πίνακα2>.<όνομα_στήλης>]  
[WHERE <συνθήκη>]  
[GROUP BY <λίστα_στηλών>]  
[ORDER BY <λίστα_στηλών>]  
[LIMIT <n>];
```

Ο όρος **JOIN** μπορεί να εμφανίζεται 0, 1 ή περισσότερες φορές. Τα ονόματα των πινάκων και των στηλών που εμφανίζονται στον όρο θα πρέπει να εξασφαλίζεται ότι υπάρχουν, δηλαδή έχουν προηγηθεί κατάλληλες εντολές **CREATE**.

Η σήμανση <όνομα_πίνακα>.<όνομα_στήλης> δείχνει ότι επιλέγουμε μια συγκεκριμένη στήλη ενός συγκεκριμένου πίνακα την οποία συγκρίνουμε με μια άλλη στήλη συγκεκριμένου πίνακα. Στην περίπτωση ύπαρξης πολλαπλών εμφανίσεων του όρου **JOIN**, οι πίνακες που χρησιμοποιούνται σε αυτούς θα πρέπει να έχουν οριστεί στο **FROM** ή σε προηγούμενο **JOIN**. Οι σημασιολογικοί έλεγχοι θα πρέπει να εξασφαλίζουν ότι ικανοποιούνται όλα αυτά.

Νέες λέξεις-κλειδιά (case-insensitive): **JOIN, ON**

- b. Η **mysql** υποστηρίζει τα ψευδώνυμα (aliases). Ένα alias είναι ένα αναγνωριστικό και χρησιμοποιείται εντός των ερωτημάτων για να αναφερθούμε σε έναν πίνακα πιο γρήγορα. Ένα alias μπορεί να εμφανιστεί μέσα σε έναν όρο **FROM** ή σε έναν όρο **JOIN** ακολουθώντας μια από τις παρακάτω μορφές και εάν γίνει χρήση alias, τότε υποχρεωτικά εμφανίζεται και στους υπόλοιπους όρους που γίνεται χρήση στηλών του πίνακα που αφορά το alias:

```

SELECT <λίστα_στηλών>
      FROM <όνομα_πίνακα1> AS <alias>
      [JOIN <όνομα_πίνακα2> ON <alias>.<όνομα_στήλης> =
      <όνομα_πίνακα2>.<όνομα_στήλης>]
      [WHERE <συνθήκη>]
      [GROUP BY <λίστα_στηλών>]
      [ORDER BY <λίστα_στηλών>]
      [LIMIT <n>];

```

ή

```

SELECT <λίστα_στηλών>
FROM <όνομα_πίνακα1>
      [JOIN <όνομα_πίνακα2> AS <alias> ON
      <όνομα_πίνακα1>.<όνομα_στήλης> = <alias>.<όνομα_στήλης>]
      [WHERE <συνθήκη>]
      [GROUP BY <λίστα_στηλών>]
      [ORDER BY <λίστα_στηλών>]
      [LIMIT <n>];

```

ή

```

SELECT <λίστα_στηλών>
FROM <όνομα_πίνακα1> AS <alias1>
      [JOIN <όνομα_πίνακα2> AS <alias2> ON <alias1>.<όνομα_στήλης>
      = <alias2>.<όνομα_στήλης>]
      [WHERE <συνθήκη>]
      [GROUP BY <λίστα_στηλών>]
      [ORDER BY <λίστα_στηλών>]
      [LIMIT <n>];

```

Οι σημασιολογικοί έλεγχοι θα πρέπει να εξασφαλίζουν την ορθή χρήση των aliases όταν χρησιμοποιούνται, ώστε να αντιστοιχίζονται στον κατάλληλο πίνακα και να ελέγχονται οι κατάλληλες στήλες αυτού. Φυσικά, τα ονόματα των πινάκων και των στηλών θα πρέπει να εξασφαλίζεται ότι υπάρχουν.

Στον όρο **select** και στους όρους **where**, **group by**, **order by**, εάν υπάρχουν στην υπό έλεγχο εντολή, θα πρέπει να διασφαλίζεται ότι γίνεται σωστή χρήση του/των alias/aliases στις στήλες που απαιτούνται. Σε αυτή την περίπτωση η εκάστοτε στήλη θα πρέπει να παρουσιάζεται ως <alias>.<όνομα_στήλης>.

Νέα λέξη-κλειδί (case-insensitive): **AS**

Σημαντικές επισημάνσεις:

- 1) Οι ζητούμενοι σημασιολογικοί έλεγχοι για τα ερωτήματα 2 και 3 θα πρέπει να υλοποιηθούν αποκλειστικά με κώδικα C/C++.
- 2) Στα παραδοτέα σας και για την προφορική παρουσίαση θα πρέπει να έχετε διαθέσιμα δοκιμαστικά αρχεία που να καλύπτουν όλες τις περιπτώσεις που υλοποιεί η γραμματική σας. Η παρουσία ή έλλειψη αυτών θα ληφθεί υπόψιν στη βαθμολόγηση.

Ενδεικτικό παράδειγμα:

```
CREATE TABLE Students (  
    student_id    int,  
    first_name    varchar(50),  
    last_name     varchar(50),  
    gpa           float  
);  
  
CREATE TABLE Courses (  
    course_id     int,  
    title         varchar(100),  
    credits       int  
);  
  
CREATE TABLE Enrollments (  
    enrollment_id int,  
    student_id    int,  
    course_id     int,  
    grade         varchar(2)  
);  
  
select * from Students;  
  
SELECT s.student_id,  
       s.first_name,  
       s.last_name,  
       c.title,  
       e.grade,  
       s.gpa  
FROM Students      AS s  
JOIN Enrollments   AS e ON student_id = e.student_id  
JOIN Courses       AS c ON e.course_id = c.course_id  
WHERE s.gpa >= 3.0  
      AND c.credits IN (3, 4)  
      AND e.grade NOT IN ('F', 'D')  
GROUP BY s.student_id, c.course_id  
ORDER BY s.gpa , c.title  
LIMIT 20;
```