

**ΑΕΙ ΠΕΙΡΑΙΑ Τ.Τ.
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ Τ.Ε.**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

Βίκτωρ - Ευστάθιος Κωστόπουλος

Εισηγητής: Αναστασία Βελώνη, Καθηγήτρια

**ΑΘΗΝΑ
ΑΠΡΙΛΙΟΣ 2016**

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

**Βίκτωρ - Ευστάθιος Κωστόπουλος
Α.Μ. 34734**

Εισηγητής:

Αναστασία Βελώνη, Καθηγήτρια

Εξεταστική Επιτροπή:

Ημερομηνία εξέτασης / / 2016

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα πτυχιακή εργασία ολοκληρώθηκε ύστερα από επίμονη προσπάθεια, καθότι βασίστηκε σε πολύ προσωπική εργασία, κόπο και μεράκι, μιας και το θέμα ήταν πολύ ενδιαφέρον.

Θα ήθελα να ευχαριστήσω πρωτίστως τους γονείς μου για την πολυετή υπομονή τους κατά την διάρκεια των σπουδών μου, ελπίζοντας ότι θα τους αποζημιώσω από εδώ και πέρα.

Ευχαριστώ επίσης την επιβλέπουσα καθηγήτριά μου, κυρία Αναστασία Βελώνη, που με έκανε να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα.

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία ασχολείται με την δημιουργία μίας διαδικτυακής πλατφόρμας / ερωτηματολογίου για αξιολόγηση ή εξέταση σπουδαστών από κάποιο επιβλέποντα.

Για την επίτευξη αυτού, κατασκευάστηκαν δύο αλγόριθμοι σε γλώσσα προγραμματισμού C, ένας για τον επιβλέποντα, ο οποίος θα ορίζει το πεδίο εξέτασης, κι ένας για τον εξεταζόμενο, που θα εκτελεί το πρόγραμμα προκειμένου να δοκιμάσει τις γνώσεις του.

Η βάση δεδομένων με τις ερωτήσεις και τις απαντήσεις θα αποθηκεύεται διαδικτυακά χρησιμοποιώντας την MySQL.

ABSTRACT

The present thesis concerns the development of an online platform / questionnaire for the test or evaluation of a candidate of any kind.

To accomplish this, two algorithms were constructed in the program language called C, one for the supervisor / professor, who will select the test field and content, and one for the candidate, who will execute the program, so he could test his knowledge over this particular field.

The database with the questions and answers will be stored over the internet via MySQL.

ΕΠΙΣΤΗΜΟΝΙΚΗ ΠΕΡΙΟΧΗ: Προγραμματισμός Ηλεκτρονικών Υπολογιστών

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: C, αλγόριθμος, MySQL, βάση δεδομένων, στατιστική

ΠΕΡΙΕΧΟΜΕΝΑ

1.	ΕΙΣΑΓΩΓΗ	9
1.1	Στόχος και χρησιμότητα	9
1.2	Πρακτικές δυσκολίες στην υλοποίηση.....	10
2.	ΕΙΣΑΓΩΓΗ ΣΤΗΝ MySQL	11
2.1	Πλεονεκτήματα της MySQL	11
2.2	Εντολές.....	15
2.2.1	Τύποι δεδομένων MySQL	15
2.2.2	Εντολές MySQL	15
2.2.3	Συνάρτηση MySQL	16
2.2.4	Συναρτήσεις C για σύνδεση με MySQL.....	17
2.2.5	Εντολή Shell.....	18
3.	ΛΟΙΠΕΣ ΓΝΩΣΕΙΣ	19
3.1	Συναρτήσεις C - συστήματος.....	19
3.2	Συναρτήσεις C - κατασκευασμένες για το παρόν πρόγραμμα	23
3.3	Μαθηματικό υπόβαθρο - Στατιστική.....	24
3.3.1	Παλινδρόμηση (Regression)	24
3.3.2	Μέθοδος ελαχίστων τετραγώνων (M.E.T.)	25
3.3.3	Διακύμανση (s^2)	29
3.3.4	Τυπική απόκλιση (s).....	30
3.3.5	Τυπικό σφάλμα εκτίμησης της εξαρτημένης μεταβλητής (Syx).....	31
3.3.6	Συντελεστής προσδιορισμού (r^2).....	34
3.3.7	Απλή γραμμική συσχέτιση (r)	35
3.3.8	Μέτρηση της συσχέτισης.....	36
4.	ΛΕΠΤΟΜΕΡΗΣ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ	39
4.1	Δημιουργία βάσης δεδομένων	39
4.2	Επεξήγηση επιλογών του λογισμικού	40
5.	ΜΕΛΛΟΝΤΙΚΕΣ ΒΕΛΤΙΩΣΕΙΣ	43
6.	ΠΑΡΑΡΤΗΜΑ	45
	Παράθεση κώδικα C	45
	Οδηγίες εγκατάστασης.....	65
	Οδηγίες εκτέλεσης	66
7.	ΒΙΒΛΙΟΓΡΑΦΙΑ	69

ΚΑΤΑΛΟΓΟΣ ΓΡΑΦΗΜΑΤΩΝ

Γράφημα 1.1: Συγκριτικά τεστ ανάμεσα στα πακέτα DB2, Informix, MS-SQL, MySQL, Solid, Sybase σε υπολογιστή με λειτουργικό NT 4.0.....	13
Γράφημα 1.2: Συγκριτικά τεστ ανάμεσα στα πακέτα DB2, Informix, MS-SQL, MySQL, Solid, Sybase σε υπολογιστή με λειτουργικό NT 4.0.....	14
Γράφημα 3.1: Γραφική παράσταση για την σχέση των y_i και \hat{y}_i	26
Γράφημα 3.2: Κανονική καμπύλη	31
Γράφημα 3.3: Διάγραμμα διασποράς των y_i	32
Γράφημα 3.4: Γράφημα για τον ορισμό του συντελεστή προσδιορισμού	34
Γράφημα 3.5: Διαγράμματα διασποράς με διάφορες συσχετίσεις	37

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο αναλύεται το αντικείμενο της πτυχιακής εργασίας, καθώς και τα προβλήματα και οι δυσκολίες που προέκυψαν κατά την συγγραφή της.

1.1 Στόχος και χρησιμότητα

Θέμα και στόχος της παρούσας πτυχιακής εργασίας, είναι η δημιουργία μίας εφαρμογής με τη χρήση της γλώσσας προγραμματισμού C, με την βοήθεια της οποίας ο εκάστοτε χρήστης θα έχει την δυνατότητα να δοκιμάσει τις γνώσεις του σε ένα συγκεκριμένο θέμα, πράγμα πολύ σημαντικό αφού θα είναι σε θέση να μάθει σε τι επίπεδο βρίσκεται όσον αφορά τον τομέα αυτό.

Η γενικότερη ιδέα είναι η αυτοαξιολόγηση ενός ατόμου μέσω ενός συστήματος ερωτήσεων και πολλαπλών απαντήσεων, από τις οποίες ο χρήστης θα καλείται να βρει την μόνη σωστή, ενώ σε διαφορετική περίπτωση θα του δίνεται, εκτός από την σωστή απάντηση, και μία απόδειξη / υπόδειξη αυτής.

Πιο συγκεκριμένα, η εφαρμογή κατά κύριο λόγο θα χρησιμοποιείται από έναν διαχειριστή, ο οποίος θα έχει διάφορες δυνατότητες, όπως εισαγωγή ερωταπαντήσεων, διαγραφή ερωτηματολόγιου, κ.ά.

Έπειτα, ο εκάστοτε χρήστης αφού ολοκληρώσει την διαδικασία απαντήσεων στις ερωτήσεις, θα μπορεί να βλέπει το ποσοστό επιτυχίας του, ενώ ταυτόχρονα αυτό θα κρατείται σε μια βάση δεδομένων μαζί με τα αποτελέσματα άλλων χρηστών.

Σε δεύτερο στάδιο, μετά την ολοκλήρωση του ερωτηματολόγιου από ένα επαρκές δείγμα χρηστών, ο διαχειριστής θα έχει την δυνατότητα μέσω στατιστικών εξισώσεων και μαθηματικών τύπων, να εξάγει συμπεράσματα ανάλογα με την ποιότητα των χρηστών που κλήθηκαν να πάρουν μέρος στο “τεστ”, με την δυσκολία των ερωτήσεων, αλλά και -αν αυτό καθίσταται δυνατό- να προβλέψει ως ένα βαθμό, ποσοστά επιτυχίας επόμενων χρηστών.

Τα δεδομένα θα αποθηκεύονται διαδικτυακά με την βοήθεια του συστήματος διαχείρισης βάσεων δεδομένων MySQL, επομένως η χρήση της εφαρμογής από τον οποιοδήποτε, μπορεί να πραγματοποιηθεί κι απομακρυσμένα.

1.2 Πρακτικές δυσκολίες στην υλοποίηση

Η κύρια δυσκολία για την ολοκλήρωση της παρούσας εργασίας, ήταν η εύρεση διαδικτυακού χώρου για την αποθήκευση της βάσης δεδομένων με τις ερωτήσεις / απαντήσεις, τα στατιστικά των συμμετεχόντων, και τα στοιχεία τους. Για την πιο ομαλή λειτουργία της εφαρμογής, ο χώρος αυτός πρέπει να διαθέτει γρήγορο server για την όσο το δυνατόν γρηγορότερη εξυπηρέτηση των αιτημάτων προς αυτόν, κι αν είναι δυνατόν, να μην χρειάζεται κάποια συνδρομή για την διασφάλιση των δεδομένων στο διαδίκτυο.

Επομένως, χρησιμοποιήθηκαν οι υπηρεσίες που προσφέρει η ιστοσελίδα www.db4free.net, ενώ φυσικά υπάρχει η δυνατότητα εύκολης αλλαγής της με σκοπό ο εκάστοτε διαχειριστής να μπορεί να προσθέσει την βάση δεδομένων του σε οποιοδήποτε website επιθυμεί.

Εξίσου σημαντική δυσκολία είναι η ανάπτυξη της εφαρμογής σε συγκεκριμένη γλώσσα προγραμματισμού (C), η οποία όσον αφορά την εκτέλεσή της είναι δυσχερής, μιας και δεν υποστηρίζεται από όλα τα λειτουργικά συστήματα. Έτσι, το παρόν πρόγραμμα δημιουργήθηκε σε περιβάλλον Linux Ubuntu, εκτελείται προφανώς σε αυτό, όπως και σε Mac OS, ενώ στο -πιο διαδεδομένο, στην Ελλάδα τουλάχιστον- περιβάλλον Windows είναι δύσκολο να εκτελεστεί. **[1]**

ΚΕΦΑΛΑΙΟ 2

ΕΙΣΑΓΩΓΗ ΣΤΗΝ MySQL

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων ανοικτού κώδικα (Relational Database Management System - RDBMS), που χρησιμοποιεί την Structured Query Language (SQL), την πιο γνωστή γλώσσα για την προσθήκη, την πρόσβαση και την επεξεργασία δεδομένων σε μία Βάση Δεδομένων.

Επειδή είναι ανοικτού κώδικα (open source), οποιοσδήποτε μπορεί να κατεβάσει τη MySQL και να την διαμορφώσει με βάση τις ανάγκες του, σύμφωνα πάντα με την γενική άδεια χρήσης. Είναι γνωστή κυρίως για την ταχύτητα, την αξιοπιστία, και την ευελιξία που παρέχει. Η MySQL αυτή τη στιγμή μπορεί να λειτουργήσει σε περιβάλλον Linux, Unix, και Windows.

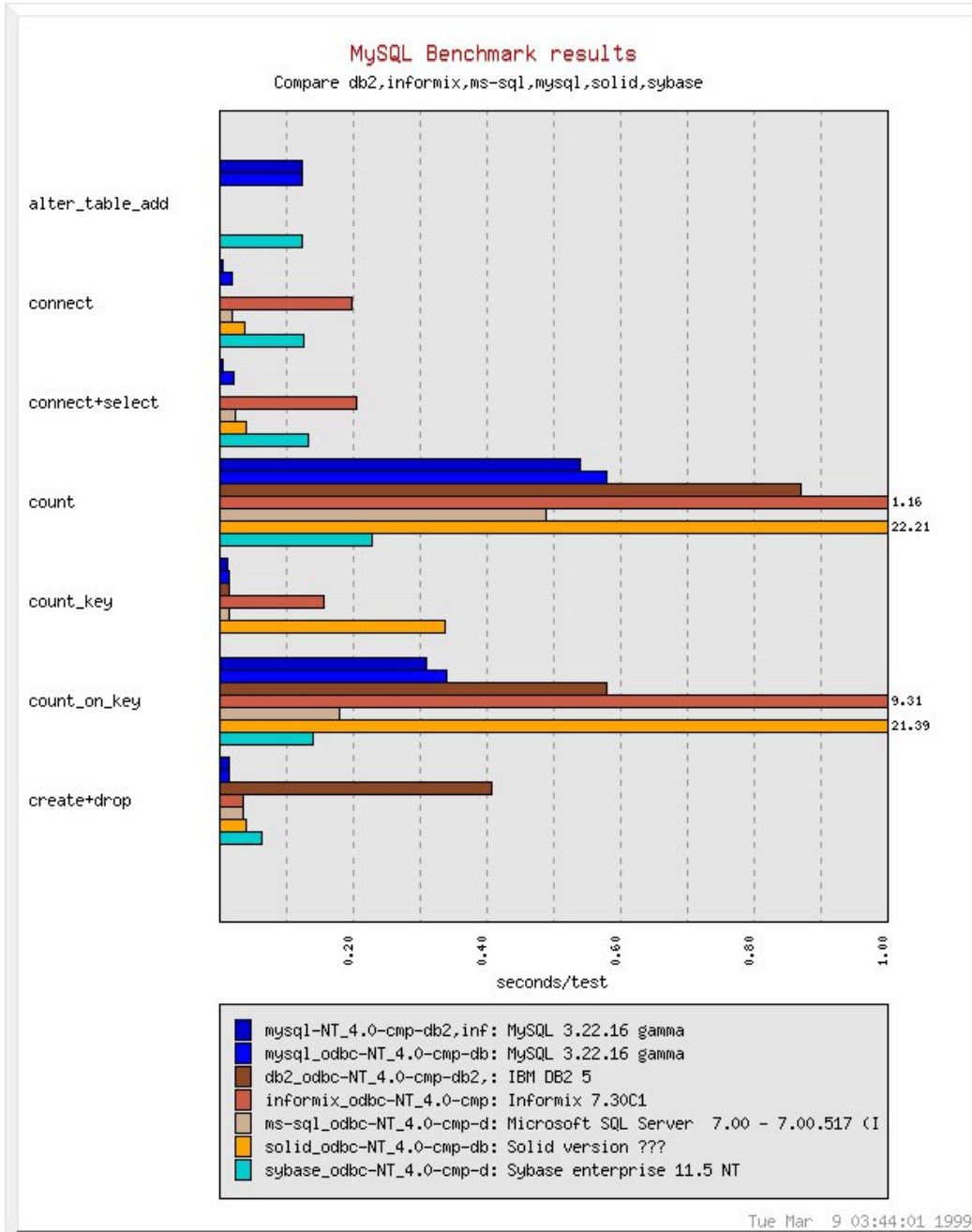
2.1 Πλεονεκτήματα της MySQL

Πρώτα από όλα η MySQL είναι ένα πακέτο βάσης, το οποίο είναι ανοικτού λογισμικού, δηλαδή διανέμεται δωρεάν και δεν χρειάζεται κάποια εταιρία οργανισμός ή κάποιος τελικός χρήστης να δαπανήσει οποιοδήποτε χρηματικό ποσό για να την αγοράσει. Μπορεί πολύ απλά να την λάβει μέσω του διαδικτύου και να την εγκαταστήσει στον υπολογιστή του από το website της MySQL (<http://www.mysql.com/>).

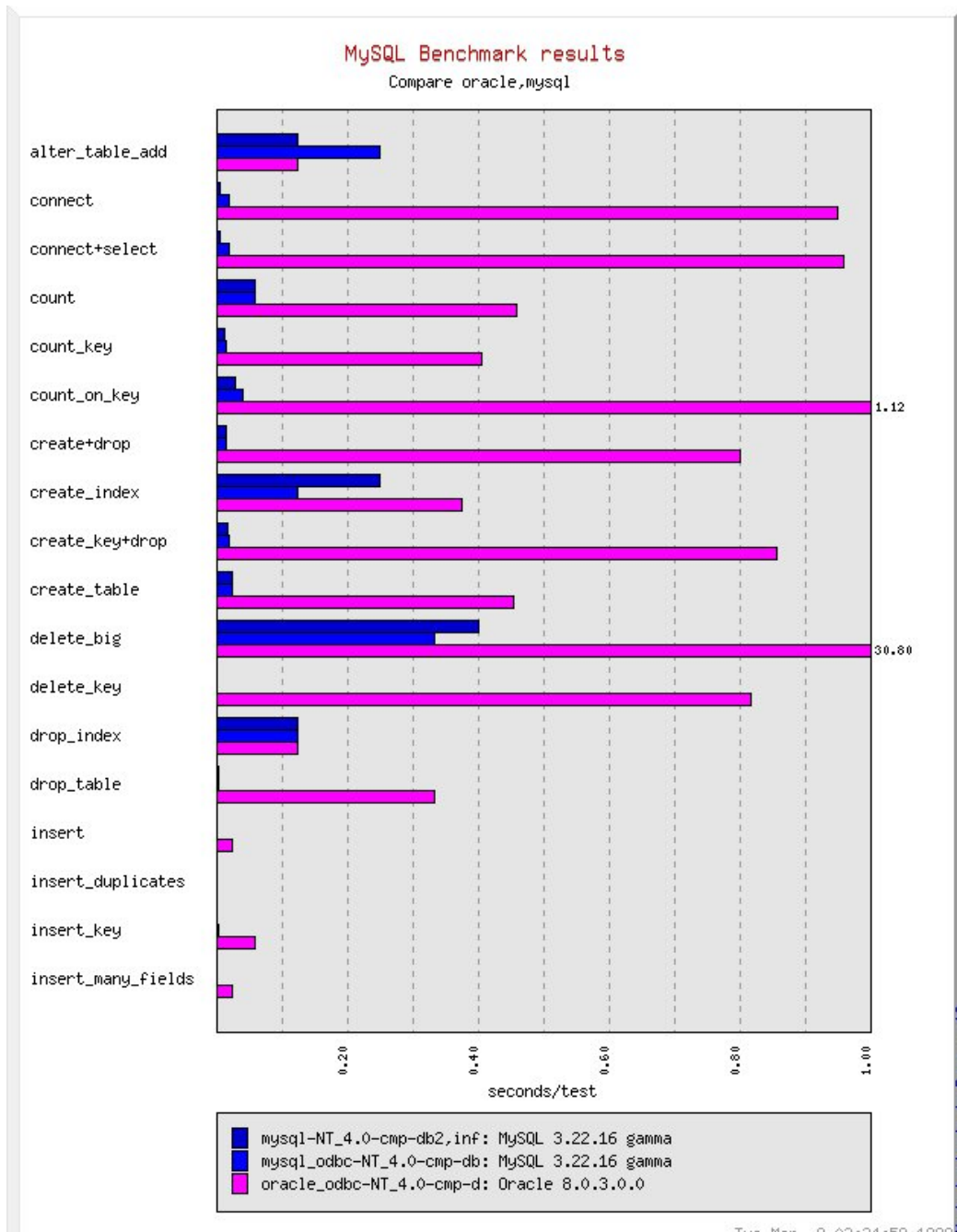
Ένας τομέας στον οποίο υπερτερεί έναντι των άλλων πακέτων βάσης, είναι αυτός της όσο το δυνατόν ομαλότερης και απροβλημάτιστης λειτουργίας, διότι πρόκειται για ένα λογισμικό, το οποίο βρίσκεται στην αγορά πολλά χρόνια, με αποτέλεσμα να έχει δοκιμαστεί σε πάρα πολλές περιπτώσεις από μικρούς οργανισμούς μέχρι εταιρίες κολοσσούς. Η ύπαρξη του πακέτου αυτού τόσα χρόνια, έχει ως αποτέλεσμα να έχει μία πάρα πολύ σταθερή βάση χωρίς προβλήματα, αφού τα περισσότερα έχουν δοκιμαστεί και λυθεί όλον αυτόν τον καιρό.

Ένα άλλο πολύ σημαντικό πλεονέκτημα που έχει φέρει την MySQL ανάμεσα στα πιο χρησιμοποιούμενα πακέτα βάσης, το οποίο ίσως να μην είναι και τόσο γνωστό, είναι ότι πρόκειται για ένα αρκετά ελαφρύ πακέτο, ίσως το πιο ελαφρύ της αγοράς, το οποίο έχει πολύ μικρές απαιτήσεις σε πόρους υπολογιστικής ισχύος από τον υπολογιστή στον οποίο θα στηθεί.

Ίσως όμως ο σημαντικότερος παράγοντας της καθιέρωσης της MySQL να είναι οι πάρα πολύ μικροί χρόνοι απόκρισης του εν λόγω πακέτου σε σύγκριση με άλλα πακέτα βάσεων δεδομένων της αγοράς. Παρακάτω επισυνάπτεται ένας συγκριτικός πίνακας από τεστ που έχουν γίνει από ανώνυμους χρήστες και από τους δημιουργούς της MySQL, στο οποίο φαίνεται η ταχύτητα και η αξιοπιστία του συγκεκριμένου πακέτου. **[2]**



Γράφημα 1.1: Συγκριτικά τεστ ανάμεσα στα πακέτα DB2, Informix, MS-SQL, MySQL, Solid, Sybase σε υπολογιστή με λειτουργικό NT 4.0 [3]



Γράφημα 1.2: Συγκριτικά τεστ ανάμεσα στα πακέτα DB2, Informix, MS-SQL, MySQL, Solid, Sybase σε υπολογιστή με λειτουργικό NT 4.0 [3]

2.2 Εντολές

2.2.1 Τύποι δεδομένων MySQL

MYSQL

Ο τύπος MYSQL χρησιμεύει ως χειριστής (handler) για την σύνδεση μιας βάσης δεδομένων. Χρησιμοποιείται για σχεδόν όλες τις MySQL συναρτήσεις.

MYSQL_RES

Η δομή αυτή αντιπροσωπεύει το αποτέλεσμα ενός αιτήματος που επιστρέφει γραμμές (rows).

Το αίτημα είναι συνήθως κάποιου τύπου από τους ακόλουθους: SELECT, SHOW, DESCRIBE ή EXPLAIN.

MYSQL_ROW

Ο τύπος MYSQL_ROW χρησιμοποιείται για την εμφάνιση μιας γραμμής αποτελεσμάτων, η οποία όμως στην αρχική της μορφή συμπεριφέρεται ως ένας πίνακας (array) αποτελούμενο από bytes τύπου string, τα οποία δεν μπορούν να επεξεργαστούν, αν δεν κληθεί η συνάρτηση `mysql_fetch_row()`.

2.2.2 Εντολές MySQL

CREATE TABLE

Η εντολή CREATE TABLE δημιουργεί έναν πίνακα με το όνομα που ακολουθεί. Ο πίνακας δημιουργείται στην προεπιλεγθείσα βάση δεδομένων.

DROP TABLE

Η εντολή DROP TABLE διαγράφει έναν ή περισσότερους πίνακες. Χρειάζεται προσοχή, μιας και διαγράφεται ο πίνακας και όλα τα περιεχόμενά του. Αν ακολουθείται από την εντολή IF EXISTS, ο πίνακας διαγράφεται εφόσον υπάρχει, για να αποφευχθεί η περίπτωση σφάλματος κατά την προσπάθεια διαγραφής πίνακα που δεν υπάρχει.

INSERT INTO

Η εντολή INSERT INTO όνομα_πίνακα (πεδίο_πίνακα) VALUES (' τιμές ') εισάγει νέες εγγραφές στον πίνακα (όνομα_πίνακα), και πιο συγκεκριμένα μεμονωμένες εισαγωγές (τιμές) στο εκάστοτε πεδίο (πεδίο_πίνακα).

UPDATE

Η εντολή UPDATE χρησιμοποιείται για την αλλαγή ήδη υπαρχόντων εγγραφών σε έναν πίνακα, ενώ με την χρήση του όρου WHERE, δίνεται η δυνατότητα της αλλαγής στοιχείων σε συγκεκριμένα πεδία, όπου δηλαδή ισχύει η συνθήκη που ακολουθεί τον όρο WHERE.

SELECT

Σκοπός της εντολής SELECT είναι η επιλογή στοιχείων από μία βάση δεδομένων. Τα στοιχεία αυτά, θα αποθηκευτούν σε έναν προσωρινό πίνακα αποτελεσμάτων, το λεγόμενο σετ αποτελεσμάτων.

Στο παρόν πρόγραμμα, η εντολή θα χρησιμοποιηθεί με τρεις τρόπους.

```
SELECT * FROM όνομα_πίνακα
```

Θα επιλεχθούν όλα τα στοιχεία από τον εν λόγω πίνακα.

```
SELECT COUNT(*) FROM όνομα_πίνακα
```

Ζητείται ο αριθμός εγγραφών (πλήθος γραμμών) από τον συγκεκριμένο πίνακα.

```
SELECT πεδίο_πίνακα_1, πεδίο_πίνακα_2 FROM όνομα_πίνακα
```

Επιλέγονται τα συγκεκριμένα πεδία (πεδίο_πίνακα_1, πεδίο_πίνακα_2) από τον εκάστοτε πίνακα (όνομα_πίνακα).

Σε ένα αίτημα, μπορεί να υπάρξει χρήση της εντολής SELECT με τον όρο WHERE (εξηγήθηκε παραπάνω).

2.2.3 Συνάρτηση MySQL

NOW()

Όταν καλείται η συνάρτηση NOW(), επιστρέφει την τρέχουσα ημερομηνία και ώρα με την ακόλουθη μορφή: XXXX-MM-HH ΩΩ:ΛΛ:ΔΔ. Οι τιμές είναι ανάλογες με την εκάστοτε χρονική ζώνη.

2.2.4 Συναρτήσεις C για σύνδεση με MySQL

`mysql_init()`

Αρχικοποιεί ένα αντικείμενο τύπου MYSQL, το οποίο θα χρησιμοποιηθεί ως όρισμα στην συνάρτηση `mysql_real_connect()`.

`mysql_real_connect()`

Η συνάρτηση αυτή πραγματοποιεί σύνδεση με μία βάση δεδομένων.

Τα ορίσματά της είναι τα εξής:

Το αντικείμενο τύπου MYSQL (εξηγήθηκε παραπάνω), το website (ή IP διεύθυνση) στο οποίο βρίσκεται η βάση δεδομένων, το όνομα χρήστη με το οποίο συνδέεται στην βάση δεδομένων, ο κωδικός του χρήστη, το όνομα της βάσης δεδομένων, ο αριθμός της θύρας (port) για την TCP/IP σύνδεση.

`mysql_store_result()`

Με την βοήθεια της εν λόγω συνάρτησης, το αποτέλεσμα ενός MySQL αιτήματος (query) μπορεί να αποθηκευτεί σε μία μεταβλητή τύπου MYSQL_RES (εξηγήθηκε παραπάνω) σε μορφή κατάλληλη για ανάγνωση από το πρόγραμμα.

`mysql_free_result()`

Ύστερα από την χρήση της `mysql_store_result()`, κρίνεται ενδεδειγμένη η χρήση της `mysql_free_result()`, ώστε να αποδεσμευτεί η μνήμη που κρατήθηκε για την αποθήκευση του αποτελέσματος -ή σετ αποτελεσμάτων- μέσω του MySQL query.

`mysql_close()`

Κλείνει μία ανοιχτή MySQL σύνδεση.

`mysql_query()`

Χρησιμοποιείται για την εκτέλεση των MySQL αιτημάτων προς τον SQL server. Ως πρώτο όρισμα έχει το αντικείμενο τύπου MYSQL (εξηγήθηκε παραπάνω), ενώ το δεύτερο όρισμα περιλαμβάνει την εντολή -ή τις εντολές διαχωρισμένες με τον χαρακτήρα ';' - σε γλώσσα MySQL.

`mysql_num_fields()`

Το όρισμα αυτής της συνάρτησης είναι το αντικείμενο τύπου MYSQL_RES (εξηγήθηκε παραπάνω), ενώ επιστρέφει το πλήθος των αποτελεσμάτων του (πλήθος των στηλών).

```
mysql_fetch_row()
```

Ανακτά την επόμενη στήλη ενός σετ αποτελεσμάτων. Όταν χρησιμοποιείται μετά την `mysql_store_result()`, η `mysql_fetch_row()` επιστρέφει την τιμή NULL όταν δεν υπάρχουν άλλες στήλες προς ανάκτηση. **[4]**

2.2.5 Εντολή Shell

```
mysqldump
```

Η συγκεκριμένη εντολή καλεί ένα πρόγραμμα, το οποίο συνδέεται με έναν MySQL server με σκοπό να κάνει αποθήκευση (backup) της βάσης δεδομένων, τέτοιο ώστε όταν εκτελεστεί από την MySQL να μπορεί να ξαναδημιουργήσει την βάση δεδομένων που υπήρχε τη στιγμή που δημιουργήθηκε το backup.

Για να αναγνωριστούν από το πρόγραμμα οι συναρτήσεις που αναφέρθηκαν, επιβεβλημένη είναι η προσθήκη των κατάλληλων βιβλιοθηκών που τις περιέχουν. Επομένως, στην αρχή των προγραμμάτων θα εισαχθούν οι παρακάτω γραμμές κώδικα:

```
#include <mysql.h>  
#include <my_global.h>
```

ΚΕΦΑΛΑΙΟ 3

ΛΟΙΠΕΣ ΓΝΩΣΕΙΣ

Το παρόν λογισμικό συνδυάζει τον προγραμματισμό σε γλώσσες C και SQL, καθώς και μαθηματικά, και πιο συγκεκριμένα στατιστική.

Σε αυτό το κεφάλαιο θα γίνει μια προσπάθεια ανάλυσης των προαπαιτούμενων που πρέπει να κατέχει κανείς για την πλήρη κατανόηση της εργασίας.

Πρόκειται να εξηγηθούν επιγραμματικά οι συναρτήσεις που επιλέχθηκαν για την δημιουργία της, καθώς και αυτές που κατασκευάστηκαν γι'αυτή, ενώ θα παρουσιαστεί και η θεωρία στατιστικής που απαιτείται για την ολοκλήρωσή της.

3.1 Συναρτήσεις C - συστήματος

switch - case

Το σετ συναρτήσεων switch-case αντικαθιστά τη χρήση μεγάλων δομών if που συγκρίνουν την τιμή μίας μεταβλητής με διάφορες τιμές "ακεραίων" (τιμές που εκφράζονται με ακέραιους αριθμούς, όπως η τιμή ενός χαρακτήρα - char).

Ανάμεσα σε κάθε περίπτωση (case), υπάρχει η εντολή break για την οριοθέτηση του εκάστοτε case.

Παράδειγμα:

```
switch (metavliti)
{
    case 1:
        printf("Έδωσες τιμή 1.\n");
        break;

    case 2:
        printf("Έδωσες τιμή 2.\n");
        break;
}
```

foopen - fclose

Η συνάρτηση foopen() χρησιμοποιείται για την δημιουργία ενός δείκτη (pointer) τύπου FILE, με σκοπό την διαχείριση ενός αρχείου, στην παρούσα περίπτωση αρχείου κειμένου, ενώ η συνάρτηση fclose() παύει την ταύτιση του εκάστοτε αυτού pointer με το αρχείο το οποίο "έδειχνε".

Η foopen() αποτελείται από δύο ορίσματα, το πρώτο είναι το όνομα του αρχείου, το οποίο πρόκειται να ταυτιστεί με τον δείκτη, ενώ το δεύτερο είναι ο ορισμός των δικαιωμάτων που θα έχει ο χρήστης σε αυτό. Επιστρέφει ένα δείκτη σε ένα αντικείμενο τύπου FILE, ο οποίος θα χρησιμοποιείται στο εξής αντί για το αρχείο στο οποίο αναφέρεται.

Η fclose() δέχεται μόνο ένα όρισμα, τον δείκτη του οποίου η ανάθεση με το εκάστοτε αρχείο πρόκειται να τερματιστεί.

fgetc

Η συνάρτηση fgetc() δέχεται ως όρισμα ένα "ρεύμα" (stream) από το οποίο θα δέχεται δεδομένα, όπως για παράδειγμα έναν δείκτη τύπου FILE, τα οποία επιστρέφονται ως μια ακέραια τιμή σε μία μεταβλητή. Έπειτα από αυτή τη διαδικασία, η θέση του δείκτη προχωράει μία θέση δίπλα, και ούτω καθεξής.

fseek

Η συνάρτηση fseek() θέτει τον εκάστοτε δείκτη σε μία συγκεκριμένη θέση. Τα ορίσματά του είναι τρία, το πρώτο είναι το όνομα του δείκτη, το δεύτερο είναι η απόσταση σε bytes (offset) που θα προχωρήσει από το σημείο αναφοράς, και το τρίτο είναι το σημείο αναφοράς. Αυτό μπορεί να είναι είτε η αρχή του αρχείου (SEEK_SET), είτε το τρέχον σημείο στο οποίο βρίσκεται εκείνη τη στιγμή ο δείκτης (SEEK_CUR), είτε το τέλος του αρχείου (SEEK_END).

fgets

Η συνάρτηση fgets() διαβάζει χαρακτήρες μέσω ενός δείκτη τύπου FILE και τα αποθηκεύει σε έναν πίνακα. Ως ορίσματα δέχεται, πρώτον, τον δείκτη του πίνακα, στον οποίο θα αποθηκευτούν οι χαρακτήρες, δεύτερον το ανώτατο πλήθος χαρακτήρων που θα δεχτεί ο πίνακας, και τρίτον, ο FILE pointer που "δείχνει" τους χαρακτήρες προς αποθήκευση.

Η αποθήκευση των χαρακτήρων θα πραγματοποιηθεί είτε όταν αναγνωστεί ο

χαρακτήρας νέας γραμμής ('\n'), είτε όταν τελειώσει το αρχείο (EOF κατάσταση), είτε όταν οι χαρακτήρες φτάσουν το προαναφερθέν ανώτατο πλήθος, ό,τι συμβεί πρώτο.

strtok

Σκοπός της συνάρτησης strtok() είναι ο διαίρεση μίας φράσης σε επιμέρους προτάσεις / λέξεις, για την διαχείριση των τελευταίων. Αυτό επιτυγχάνεται με την ανάθεση ενός ή περισσότερων χαρακτήρων τον ρόλο των διαχωριστικών, με αποτέλεσμα κατά την σάρωση της αρχικής φράσης, το πρόγραμμα να την χωρίζει σε κομμάτια όποτε συναντάει έναν από αυτούς τους χαρακτήρες.

Δέχεται ως ορίσματα τον pointer που "δείχνει" στην προς διαίρεση φράση, και μια σειρά χαρακτήρων (string) που υποδηλώνει ποιοι χαρακτήρες θα οριστούν ως τα διαχωριστικά.

Πιο συγκεκριμένα, κατά την πρώτη κλήση της, η συνάρτηση strtok() επιστρέφει σε έναν δείκτη τύπου χαρακτήρα (char) την αρχή της επιμέρους πρότασης, η οποία ξεκινάει από τον πρώτο χαρακτήρα που δεν έχει οριστεί ως διαχωριστικό, και ολοκληρώνεται όταν θα βρεθεί το πρώτο.

Από εκεί κι ύστερα, κάθε κλήση της συνάρτησης σαν πρώτο όρισμα έχει έναν NULL δείκτη, ώστε ως αρχικό σημείο σάρωσης να χρησιμοποιηθεί η επόμενη θέση από την οποία βρίσκεται ο pointer. Η διαδικασία τερματίζεται όταν εμφανιστεί ο null χαρακτήρας ('\0'). **[5]**

snprintf

Η συνάρτηση snprintf() χρησιμεύει για την μεταφοράς μιας γραμμής χαρακτήρων (string) από έναν πίνακα σε έναν άλλο. (Πίνακας θεωρείται και ένας δείκτης τύπου char / int, ο οποίος "δείχνει" στην αρχή ενός string)

Τα ορίσματά της είναι τα εξής:

Ο pointer που "δείχνει" εκεί που πρόκειται να αποθηκευτεί η γραμμή χαρακτήρων.

Το μέγιστο πλήθος των bytes που θα χρησιμοποιηθούν για την αποθήκευση του string -επομένως, γίνεται κατανοητό πως η μνήμη που πρέπει να βρεθεί, θα πρέπει να είναι τουλάχιστον τόσα bytes.

Ο pointer που "δείχνει" την αρχή του string προς αντιγραφή, ή την ίδια την γραμμή χαρακτήρων.

Παράδειγμα:

```
snprintf(buffer, 30, "Auto einai ena paradeigma");
```

system

Η συνάρτηση `system()` δέχεται ένα όρισμα, το οποίο είναι μία εντολή προς το τερματικό, δηλαδή τον φλοιό του συστήματος (`shell`). Επιστρέφει αφού εκτελεστεί αυτή η εντολή.

atoi

Η συνάρτηση `atoi()` μετατρέπει την γραμμή χαρακτήρων που δέχεται σαν όρισμα, στην αντίστοιχη τιμή της ως γραμμή ακεραίων (`int`). Αν δεν είναι έγκυρη η παράμετρος, θα επιστραφεί η τιμή 0.

atof

Σαν την συνάρτηση `atoi()`, η `atof()` εξυπηρετεί τον ίδιο σκοπό, μα αντί για ακέραιους, μετατρέπει την προς εισαγωγή γραμμή χαρακτήρων, σε δεκαδικούς αριθμούς (`float`). Κι εδώ, αν δεν είναι έγκυρο το όρισμα, θα επιστραφεί η τιμή 0 (αν είναι π.χ. γράμματα).

malloc - free

Οι δύο αυτές συναρτήσεις είναι σχετικές με την εύρεση επαρκούς μνήμης για την αποθήκευση δεδομένων.

Πιο συγκεκριμένα, η συνάρτηση `malloc()` χρησιμεύει στην δυναμική δέσμευση ενός μπλοκ μνήμης μεγέθους ίσο με το όρισμά της. Επιστρέφει τον `pointer` που "δείχνει" στην αρχή αυτού του μπλοκ.

Η συνάρτηση `free()` καλείται αφού έχει ολοκληρωθεί ο σκοπός της `malloc()`, για την αποδέσμευση της μνήμης, ώστε να μπορεί να επαναχρησιμοποιηθεί στο μέλλον. Το όρισμά της είναι ο `pointer` που "έδειχνε" στο μπλοκ μνήμης κατά την κλήση της `malloc()`.

sqrt

Η συνάρτηση `sqrt()` επιστρέφει την τετραγωνική ρίζα του αριθμού που έχει ως όρισμά της.

Στο πρόγραμμα θα πρέπει να έχει προστεθεί η βιβλιοθήκη που περιέχει την εν

λόγω συνάρτησης, η `math.h` (`#include <math.h>`).

`strcpy`

Η συνάρτηση `strcpy()` χρησιμοποιείται για την αντιγραφή μίας γραμμής χαρακτήρων (`string`). Διαθέτει δύο ορίσματα, το πρώτο είναι ο `pointer` που "δείχνει" πού θα αντιγραφεί η γραμμή αυτή, ενώ το δεύτερο είναι ο `pointer` που "δείχνει" στην αρχή της προς αντιγραφή γραμμής. [6]

3.2 Συναρτήσεις C - κατασκευασμένες για το παρόν πρόγραμμα

`int empty_stdin()`

Η συνάρτηση `empty_stdin()` δημιουργήθηκε για να "αδειάζει" την προσωρινή μνήμη στην οποία αποθηκεύονται οι χαρακτήρες που έχει πληκτρολογήσει ο χρήστης (`standard input`).

Η λειτουργία της περιγράφεται ως εξής:

Κατά την κλήση της, αρχίζει η ανάγνωση των εισαγόμενων χαρακτήρων από το πληκτρολόγιο, μέχρι να βρεθεί ο χαρακτήρας του πλήκτρου Enter (`'\n'`). Τότε, η συνάρτηση επιστρέφει την τιμή 1 στο κυρίως πρόγραμμα. Με αυτό τον τρόπο, δεν λαμβάνονται υπόψιν οι χαρακτήρες πριν το πάτημα του πλήκτρου Enter. Πιο συγκεκριμένα, η συνάρτηση αυτή είναι απαραίτητη, διότι χρησιμοποιείται για τον έλεγχο σωστής εισαγωγής αριθμητικού χαρακτήρα από τον χρήστη, επομένως χωρίς την χρήση της, το πρόγραμμα στην επόμενη (πιθανή) επανάληψη θα δεχόταν ως όρισμα την συνέχεια της γραμμής χαρακτήρων που δόθηκε στην προηγούμενη.

Παράδειγμα:

Έστω ότι το πρόγραμμα ζητάει από τον χρήστη να εισάγει έναν αριθμό από το 1 ως και το 6, και ο χρήστης πληκτρολόγησε: "abc83"

Χωρίς την χρήση της `empty_stdin()`, το πρόγραμμα θα διάβαζε αρχικά τον χαρακτήρα 'a', που δεν είναι αποδεκτός, θα έβγαζε μήνυμα λάθους εισαγωγής, και θα ξαναζητούσε από τον χρήστη να εισάγει έναν αριθμό από το 1 ως και το 6. Αυτόματα όμως το πρόγραμμα θα συνέχιζε την ανάγνωση από το `buffer` του πληκτρολογίου, αφού δεν άδειασε. Επομένως, θα έβγαζε μήνυμα λάθους 4 φορές, μέχρι που θα έφτανε στον αριθμό 3, οπότε και θα συνέχιζε παρακάτω η εκτέλεση

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

του προγράμματος, κάτι που προφανώς δεν είναι λειτουργικό.

```
void normalEqu(float x[], float y[])
```

Συνάρτηση, η οποία κατασκευάστηκε με σκοπό τον υπολογισμό και την εμφάνιση στατιστικών στοιχείων σχετικών με τους χρήστες που απάντησαν τις ερωτήσεις της εφαρμογής. Πιο συγκεκριμένα, εμφανίζονται οι κανονικές εξισώσεις για χάραξη ευθείας καθώς και για χάραξη παραβολής μεταξύ δύο μεγεθών, το πλήθος των σωστών απαντήσεων και του εξαμήνου που βρίσκεται ο εκάστοτε χρήστης/φοιτητής (το οποίο προφανώς μπορεί να υποδηλώνει και την ηλικία του) , όπως και οι μέσοι όροι τους, η διακύμανση, η τυπική απόκλιση, οι συντελεστές σχετίσεως και προσδιορισμού, και τέλος το δείγμα (πλήθος των ατόμων που ολοκλήρωσαν το ερωτηματολόγιο).

Η αναλυτικότερη λειτουργία της θα εξεταστεί στο ακόλουθο κεφάλαιο.

3.3 Μαθηματικό υπόβαθρο - Στατιστική

Σκοπός του μαθηματικού σκέλους της παρούσας εργασίας, είναι η εξαγωγή συμπερασμάτων που αφορούν στατιστικούς ελέγχους και περαιτέρω διερεύνηση.

3.3.1 Παλινδρόμηση (Regression)

Για την μελέτη κάθε μιας μεταβλητής χωριστά, πρέπει να εξεταστούν οι μονάδες ενός πληθυσμού ως προς ένα χαρακτηριστικό τους.

Όμως τα διάφορα μεγέθη συνήθως δεν μεταβάλλονται συναρτήσει ενός παράγοντα, αλλά δέχονται την επίδραση κι άλλων μεγεθών. Π.χ. το ύψος των παιδιών εξαρτάται από την ηλικία τους, τον τόπο γέννησης, τις συνθήκες διατροφής. Επίσης, η ποσότητα κατανάλωσης ενός αγαθού εξαρτάται από την τιμή του αγαθού, το εισόδημα των καταναλωτών, από τις τιμές άλλων αγαθών που μπορεί να το αντικαταστήσουν, τις καταναλωτικές συνήθειες, κτλ.

Αν λοιπόν μία μεταβλητή y εξαρτάται από τις μεταβλητές x_1, x_2, \dots, x_k είναι απαραίτητο να μελετηθεί η y σε συνάρτηση με τις μεταβλητές που την επηρεάζουν

ή τουλάχιστον με τις κυριότερες από αυτές.

Με την ανάλυση παλινδρόμησης επιδιώκεται, με βάση τις επιδράσεις που έχουν προκαλέσει διάφορες μεταβλητές x_1, x_2, \dots, x_k σε μία μεταβλητή y , η ποσοτική εκτίμηση της σχέσης εξάρτησης μεταξύ της y και των x_1, x_2, \dots, x_k , δηλαδή της σχέσης:

$$y = f(x_1, x_2, \dots, x_k)$$

Αν η y θεωρηθεί ότι δέχεται την επίδραση μίας μόνο μεταβλητής x , δηλαδή:

$$y = f(x)$$

έχουμε την **απλή** παλινδρόμηση.

Αν μάλιστα η σχέση εξάρτησης είναι γραμμική, δηλαδή:

$$y = \alpha + \beta x$$

έχουμε την απλή γραμμική παλινδρόμηση, με την οποία αρχικά θα ασχοληθούμε.

Οι μεταβλητές x_1, x_2, \dots, x_k είναι ονομάζονται **ερμηνευτικές** ή **ανεξάρτητες** μεταβλητές και η y **εξαρτημένη** μεταβλητή.

Η ανάλυση συσχέτισης ασχολείται με την ένταση της υφιστάμενης σχέσης εξάρτησης μεταξύ της εξαρτημένης μεταβλητής και των ερμηνευτικών μεταβλητών και η οποία ένταση μετρείται με κατάλληλη στατιστική παράμετρο.

3.3.2 Μέθοδος ελαχίστων τετραγώνων (Μ.Ε.Τ.)

Θα χρησιμοποιηθεί η μέθοδος των ελαχίστων τετραγώνων για να εκτιμηθούν οι παράμετροι α και β του γραμμικού υποδείγματος.

Θεωρείται ένα δείγμα από n τιμές των μεταβλητών x και y :

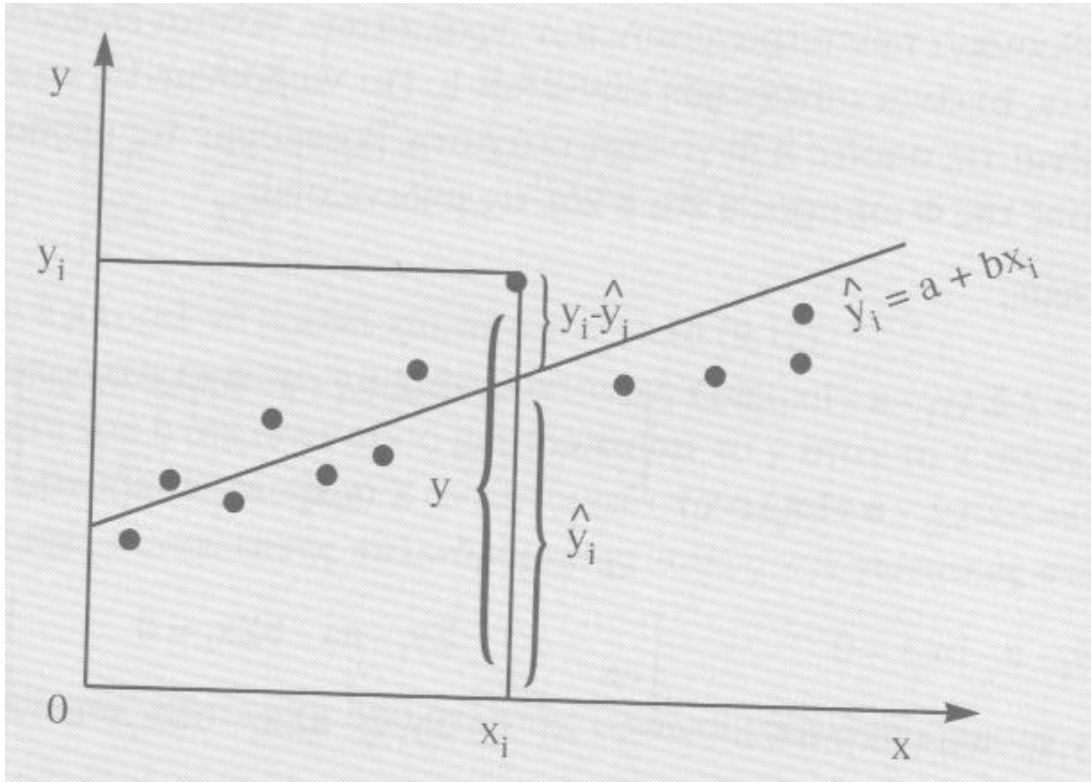
x_i : $x_1, x_2, x_3, \dots, x_n$

y_i : $y_1, y_2, y_3, \dots, y_n$

Πρέπει να υπολογιστεί η γραμμική σχέση εξάρτησης:

$$\hat{y} = \alpha + \beta x_i \text{ (εξίσωση παλινδρόμησης)}$$

που συνδέει τα y_i με τα x_i .



Γράφημα 3.1: Γραφική παράσταση για την σχέση των y_i και \hat{y}_i

Για τυχόν x_i θα έχουμε την αντίστοιχη τιμή y_i που προκύπτει από το δείγμα και εκείνη την τιμή y_i που προκύπτει από την εξίσωση $\hat{y} = \alpha + \beta x_i$.

Για το x_i θα έχουμε την διαφορά της εκ παρατηρήσεως τιμής του y με την εξ υπολογισμού τιμή του y (\hat{y}):

$$y_i - \hat{y}_i$$

Λόγω της περίπτωσης η διαφορά αυτή να είναι αρνητική, το αποτέλεσμα υψώνεται στο τετράγωνο.

Το ολικό σφάλμα βρίσκεται κατά το άθροισμα όλων των επιμέρους αυτών διαφορών:

$$F = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Αντικαθιστώντας το \hat{y}_i με την παράσταση $\alpha + \beta x_i$, έχουμε:

$$F = \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2$$

Η σωστή εκτίμηση των α και β είναι εκείνη για την οποία ο παραπάνω τύπος,

δηλαδή το άθροισμα των τετραγώνων των σφαλμάτων, γίνεται ελάχιστος.

Για να βρεθεί το ελάχιστο της εν λόγω συνάρτησης, πρέπει να βρεθεί η πρώτη παράγωγός της και να εξισωθεί με το μηδέν.

Όμως, οι ανεξάρτητες μεταβλητές είναι δύο, η α και η β , οπότε πρέπει να βρεθούν οι μερικές παράγωγοι της συνάρτησης ως προς α και ως προς β , και να εξισωθούν με το μηδέν.

Άρα:

$$\begin{aligned}\frac{\partial F}{\partial \alpha} &= 2 \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2 (-1) \Rightarrow -2 \sum_{i=1}^n (y_i - \alpha - \beta x_i) = 0 \Rightarrow \\ &\Rightarrow \sum_{i=1}^n (y_i - \alpha - \beta x_i) = 0 \Rightarrow \sum_{i=1}^n y_i - n\alpha - \beta \sum_{i=1}^n x_i = 0 \Rightarrow \\ &\Rightarrow n\alpha + \left(\sum_{i=1}^n x_i \right) \beta = \sum_{i=1}^n y_i\end{aligned}$$

και

$$\begin{aligned}\frac{\partial F}{\partial \beta} &= 2 \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2 (-x_i) \Rightarrow -2 \sum_{i=1}^n (y_i x_i - \alpha x_i - \beta x_i^2) = 0 \Rightarrow \\ &\Rightarrow \sum_{i=1}^n (y_i x_i - \alpha x_i - \beta x_i^2) = 0 \Rightarrow \sum_{i=1}^n y_i x_i - \alpha \sum_{i=1}^n x_i - \beta \sum_{i=1}^n x_i^2 = 0 \Rightarrow \\ &\Rightarrow \left(\sum_{i=1}^n x_i \right) \alpha + \left(\sum_{i=1}^n x_i^2 \right) \beta = \sum_{i=1}^n y_i x_i\end{aligned}$$

Οι δύο τελικές εξισώσεις ονομάζονται **κανονικές εξισώσεις**.

Σημείωση: Κατά την διαδικασία εύρεσης ακρότατου, για την εξαγωγή συμπεράσματος σχετικά με το αν το ακρότατο είναι μέγιστο ή ελάχιστο, γίνεται χρήση του κριτηρίου της δεύτερης παραγωγού, κατά το οποίο αν αυτή είναι θετική, το ακρότατο είναι ελάχιστο, ενώ αν είναι αρνητική, το ακρότατο είναι μέγιστο. Δεύτερος τρόπος, είναι η εξέταση του πρόσημου της πρώτης παραγωγού για τιμή μικρότερη της τιμής στην οποία αυτή μηδενίζεται και για τιμή μεγαλύτερή της. Αν η μικρότερη είναι αρνητική και η μεγαλύτερη θετική, σημαίνει ότι το ακρότατο είναι ελάχιστο, ενώ αν η μικρότερη είναι θετική και η μεγαλύτερη αρνητική, τότε το ακρότατο είναι μέγιστο.

Παρ'όλα αυτά, η φύση του συγκεκριμένου προβλήματος δεν εγείρει αμφισβήτηση σχετικά με το αν το ακρότατο είναι μέγιστο ή ελάχιστο, διότι για

μέγιστο μπορούν να δοθούν αυθαίρετα τιμές, ώστε η τιμή να φτάσει σε δυσθεώρητα μεγέθη. Άρα, το ακρότατο είναι ελάχιστο.

Η εύρεση των α και β , αφού διατίθενται δύο εξισώσεις με δύο αγνώστους, θα γίνει μέσω οριζουσών, και πιο συγκεκριμένα μέσω του κανόνα του Cramer.

Επομένως, θα έχουμε:

$$D_{\alpha} = \begin{vmatrix} \sum_{i=1}^n y & \sum_{i=1}^n x \\ \sum_{i=1}^n xy & \sum_{i=1}^n x^2 \end{vmatrix} = \sum_{i=1}^n y \sum_{i=1}^n x^2 - \sum_{i=1}^n x \sum_{i=1}^n xy$$

$$D_{\beta} = \begin{vmatrix} n & \sum_{i=1}^n y \\ \sum_{i=1}^n x & \sum_{i=1}^n xy \end{vmatrix} = n \sum_{i=1}^n xy - \sum_{i=1}^n y \sum_{i=1}^n x$$

και

$$D = \begin{vmatrix} n & \sum_{i=1}^n x \\ \sum_{i=1}^n x & \sum_{i=1}^n x^2 \end{vmatrix} = n \sum_{i=1}^n x^2 - \left(\sum_{i=1}^n x \right)^2$$

Άρα,

$$\alpha = \frac{D_{\alpha}}{D} = \frac{\sum_{i=1}^n y \sum_{i=1}^n x^2 - \sum_{i=1}^n x \sum_{i=1}^n xy}{n \sum_{i=1}^n x^2 - \left(\sum_{i=1}^n x \right)^2}$$

και

$$\beta = \frac{D_{\beta}}{D} = \frac{n \sum_{i=1}^n xy - \sum_{i=1}^n y \sum_{i=1}^n x}{n \sum_{i=1}^n x^2 - \left(\sum_{i=1}^n x \right)^2}$$

Ένας άλλος τρόπος εύρεσης της άλλης παραμέτρου αφού έχει βρεθεί η μία, είναι ο εξής:

Έστω ότι το β έχει βρεθεί και είναι άγνωστο το α .

$$n\alpha + \left(\sum_{i=1}^n x_i\right)\beta = \sum_{i=1}^n y_i$$

Στην εξίσωση διαιρώντας και τα δύο μέλη με n , έχουμε:

$$\alpha + \beta \frac{\sum_{i=1}^n x_i}{n} = \frac{\sum_{i=1}^n y_i}{n} \Rightarrow \alpha + \beta \bar{x} = \bar{y} \Rightarrow \alpha = \bar{y} - \beta \bar{x}$$

Αντικαθιστώντας, βρίσκεται η τιμή και της άλλης παραμέτρου. [7]

3.3.3 Διακύμανση (s^2)

Ένας τρόπος για να υπολογίσουμε τη διασπορά των παρατηρήσεων x_1, x_2, \dots, x_n μιας μεταβλητής x θα ήταν να αφαιρέσουμε τη μέση τιμή \bar{x} από κάθε παρατήρηση και να βρούμε τον αριθμητικό μέσο των διαφορών αυτών, δηλαδή τον αριθμό:

$$\frac{(x_1 - \bar{x}) + (x_2 - \bar{x}) + \dots + (x_n - \bar{x})}{n} = \frac{\sum_{i=1}^n (x_i - \bar{x})}{n}$$

Ο αριθμός αυτός όμως είναι ίσος με μηδέν, αφού

$$\frac{(x_1 - \bar{x}) + (x_2 - \bar{x}) + \dots + (x_n - \bar{x})}{n} = \frac{x_1 + x_2 + \dots + x_n}{n} - \frac{n\bar{x}}{n} = \bar{x} - \bar{x} = 0$$

Γ'αυτό, ως ένα μέτρο διασποράς, παίρνουμε τον μέσο όρο των τετραγώνων των αποκλίσεων των x_i από την μέση τιμή τους \bar{x} . Το μέτρο αυτό καλείται **διακύμανση** ή **διασπορά** (variance) και ορίζεται από τη σχέση

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$$

Σημείωση: Για τον υπολογισμό της διακύμανσης από τον παραπάνω τύπο, είναι φανερό πως ο πίνακας των δεδομένων θα πρέπει να προσπελαστεί δύο φορές, μία για την εύρεση της μέσης τιμής τους (\bar{x}), και ακόμα μία για την εύρεση της διαφοράς ($x_i - \bar{x}$), οπότε προγραμματιστικά, δεν είναι τόσο αποδοτικός τρόπος.

Αν όμως αναπτυχθεί το τετράγωνο μέσα στο άθροισμα:

$$\frac{\sum_{i=1}^n (x^2 - 2x\bar{x} + \bar{x}^2)}{n} = \frac{\sum_{i=1}^n x^2 - 2\bar{x} \sum_{i=1}^n x + n\bar{x}^2}{n} = \frac{\sum_{i=1}^n x^2}{n} - \frac{2\bar{x} \sum_{i=1}^n x}{n} + \frac{n\bar{x}^2}{n} = \frac{\sum_{i=1}^n x^2}{n} - 2\bar{x}^2 + \bar{x}^2 = \frac{\sum_{i=1}^n x^2}{n} - \bar{x}^2$$

Έτσι, ο πίνακας των x θα προσπελαστεί μόνο μία φορά. **[8]**

3.3.4 Τυπική απόκλιση (s)

Τυπική απόκλιση ονομάζεται η θετική τετραγωνική ρίζα της διακύμανσης, δηλαδή

$$s = \sqrt{s^2}$$

Αν η καμπύλη συχνοτήτων για το χαρακτηριστικό που εξετάζεται είναι κανονική ή περίπου κανονική, τότε η τυπική απόκλιση s έχει τις παρακάτω ιδιότητες:

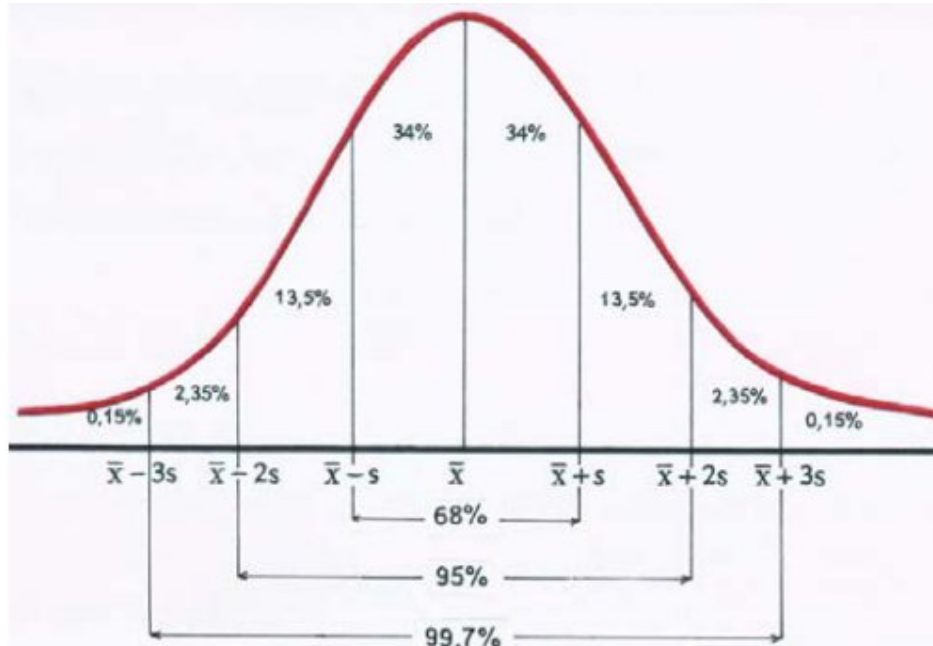
Στο διάστημα $(\bar{x}-s, \bar{x}+s)$ ανήκει το 68% περίπου των παρατηρήσεων, οπότε σε καθένα από τα διαστήματα $(\bar{x}-s, \bar{x})$ και $(\bar{x}, \bar{x}+s)$ ανήκει το 34 % περίπου των παρατηρήσεων.

Στο διάστημα $(\bar{x}-2s, \bar{x}+2s)$ ανήκει το 95 % περίπου των παρατηρήσεων, οπότε σε καθένα από τα διαστήματα $(\bar{x}-2s, \bar{x}-s)$ και $(\bar{x}+s, \bar{x}+2s)$ ανήκει το $\frac{95-68}{2}$, δηλαδή το 13,5 % περίπου των παρατηρήσεων.

Στο διάστημα $(\bar{x}-3s, \bar{x}+3s)$ ανήκει περίπου το 99,7 % των παρατηρήσεων, οπότε

σε καθένα από τα διαστήματα $(\bar{x}-3s, \bar{x}-2s)$ και $(\bar{x}+2s, \bar{x}+3s)$ ανήκει περίπου το $\frac{99,7-95}{2}$, δηλαδή το 2,35 % των παρατηρήσεων.

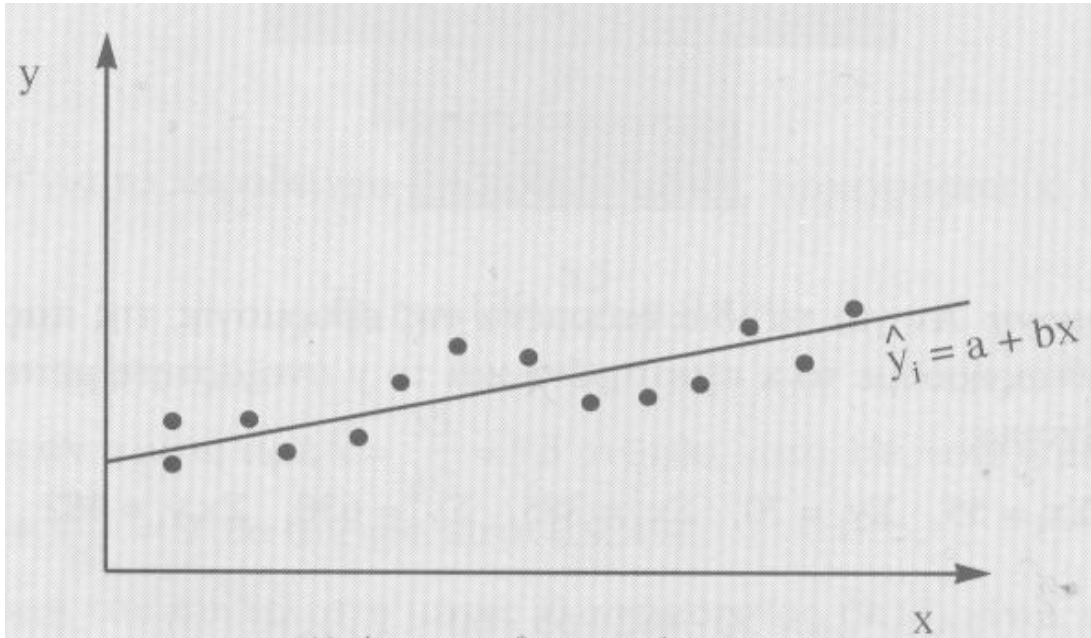
Το εύρος των παρατηρήσεων είναι περίπου ίσο με έξι τυπικές αποκλίσεις, δηλαδή $R \approx 6s$.



Γράφημα 3.2: Κανονική καμπύλη [8]

3.3.5 Τυπικό σφάλμα εκτίμησης της εξαρτημένης μεταβλητής (S_{yx})

Η εξίσωση παλινδρόμησης $\hat{y}_i = \alpha + \beta x_i$ μπορεί να χρησιμοποιηθεί σαν ένα στατιστικό μέσο για να προβλέψουμε τις τιμές της εξαρτημένης μεταβλητής y σε δεδομένες μελλοντικές τιμές της ανεξάρτητης μεταβλητής x . Η προγνωστική ικανότητα της εξίσωσης παλινδρόμησης εξαρτάται από τον **βαθμό διασποράς** των τιμών της y που έχουν προκύψει από παρατηρήσεις, γύρω από την γραμμή παλινδρόμησης.



Γράφημα 3.3: Διάγραμμα διασποράς των y_i

Δηλαδή αν οι τιμές y_i που προέρχονται από παρατηρήσεις βρίσκονται κοντά στην γραμμή παλινδρόμησης $\hat{y}_i = \alpha + \beta x_i$, τότε η εξίσωση παλινδρόμησης θα προσαρμόζεται πολύ καλά στα εμπειρικά δεδομένα και έτσι μπορεί να χρησιμεύει σαν αξιόπιστο όργανο για να γίνει πρόβλεψη των τιμών της εξαρτημένης μεταβλητής. Αντίθετα, αν οι τιμές της y που προέρχονται από παρατηρήσεις, απέχουν από την γραμμή παλινδρόμησης, τότε μειώνεται η αξιοπιστία και επομένως η προγνωστική ικανότητα του εκτιμηθέντος υποδείγματος. Επομένως, θα πρέπει να υπάρχει ένα στατιστικό μέτρο που να μας παρέχει τον βαθμό της διασποράς της εξαρτημένης μεταβλητής y_i γύρω από την γραμμή παλινδρόμησης. Το μέτρο αυτό ονομάζεται **τυπικό σφάλμα εκτίμησης** της εξαρτημένης μεταβλητής και είναι:

$$S_{yx} = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y})^2}{n-2}}$$

Όσο μικρότερες είναι οι διαφορές $y_i - \hat{y}$, δηλαδή όσο οι τιμές y_i , που προέρχονται από παρατηρήσεις, βρίσκονται πιο κοντά στις τιμές \hat{y} που δίνονται από την

εξίσωση παλινδρόμησης, τόσο μικρότερο θα είναι το τυπικό σφάλμα εκτίμησης της y . Έτσι, πολύ μικρό S_{yx} δηλώνει πολύ μικρό βαθμό διασποράς των y_i γύρω από τα \hat{y}_i και άρα μεγάλη αξιοπιστία και προγνωστική ικανότητα της εξίσωσης παλινδρόμησης.

$$\sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Στον παραπάνω τύπο το άθροισμα $\sum_{i=1}^n (y_i - \hat{y}_i)^2$ διαιρείται με $n-2$, διότι κατά τον υπολογισμό των θεωρητικών τιμών \hat{y}_i υπολογίστηκαν οι σταθερές α και β της εξίσωσης και χάθηκαν δύο βαθμοί ελευθερίας, με αποτέλεσμα να απομένουν $n-2$ βαθμοί ελευθερίας.

Έχουμε:

$$\begin{aligned} \sum_{i=1}^n (y_i - \hat{y}_i)^2 &= \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2 = \sum_{i=1}^n (y_i - \alpha - \beta x_i) (y_i - \alpha - \beta x_i) = \\ &= \sum_{i=1}^n y_i (y_i - \alpha - \beta x_i) - \alpha \sum_{i=1}^n (y_i - \alpha - \beta x_i) - \beta \sum_{i=1}^n x_i (y_i - \alpha - \beta x_i) = \\ &= \sum_{i=1}^n y_i^2 - \alpha \sum_{i=1}^n y_i - \beta \sum_{i=1}^n x_i y_i \end{aligned}$$

διότι

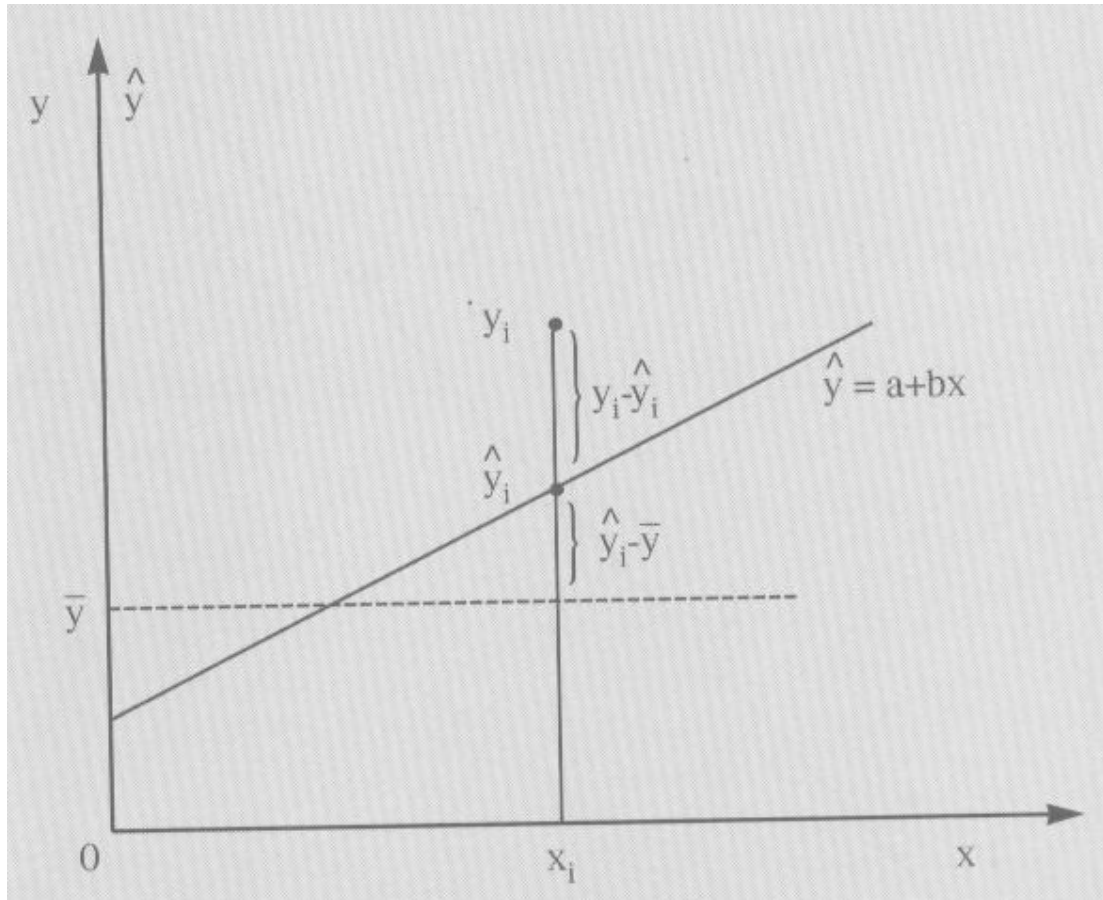
$$\sum_{i=1}^n y_i - n\alpha - \beta \sum_{i=1}^n x_i = 0 \quad \text{και} \quad \sum_{i=1}^n x_i y_i - \alpha \sum_{i=1}^n x_i - \beta \sum_{i=1}^n x_i^2 = 0$$

όπως προκύπτει από το σύστημα κανονικών εξισώσεων που περιγράφηκε παραπάνω.

Άρα, πλέον έχουμε:

$$S_{yx} = \sqrt{\frac{\sum_{i=1}^n y_i^2 - \alpha \sum_{i=1}^n y_i - \beta \sum_{i=1}^n x_i y_i}{n-2}} \quad [7]$$

3.3.6 Συντελεστής προσδιορισμού (r²)



Γράφημα 3.4: Γραφική παράσταση για τον ορισμό του συντελεστή προσδιορισμού

Όπως παρατηρούμε από το παραπάνω σχήμα, έχουμε:

$$y_i - \bar{y} = (\hat{y}_i - \bar{y}) + (y_i - \hat{y}_i)$$

Δηλαδή για την τιμή x_i , η διαφορά του γενικού μέσου \bar{y} από την πραγματική τιμή y_i της y (δηλαδή η $y_i - \hat{y}_i$) διασπάται σε άθροισμα δύο διαφορών, της διαφοράς του γενικού μέσου \bar{y} από την θεωρητική τιμή \hat{y}_i και της διαφοράς της θεωρητικής τιμής \hat{y}_i από την πραγματική τιμή y_i .

$$\begin{aligned} y_i - \bar{y} &= (\hat{y}_i - \bar{y}) + (y_i - \hat{y}_i) \Rightarrow (y_i - \bar{y})^2 = ((\hat{y}_i - \bar{y}) + (y_i - \hat{y}_i))^2 = \\ &= (\hat{y}_i - \bar{y})^2 + (y_i - \hat{y}_i)^2 + 2(\hat{y}_i - \bar{y})(y_i - \hat{y}_i) \Rightarrow \\ &\Rightarrow \sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n (y_i - \hat{y}_i)^2 + 2 \sum_{i=1}^n (\hat{y}_i - \bar{y})(y_i - \hat{y}_i) \end{aligned}$$

Στο δεύτερο μέλος της εξίσωσης ο τελευταίος όρος είναι ίσος με μηδέν.

Διότι αν αντικατασταθεί το \hat{y}_i με το $\alpha x + \beta$, έχουμε:

$$\begin{aligned} \sum_{i=1}^n (\hat{y}_i - \bar{y})(y_i - \hat{y}_i) &= \sum_{i=1}^n (\alpha + \beta x - \bar{y})(y - \alpha - \beta x) = \\ &= \alpha \sum_{i=1}^n (y - \alpha - \beta x) + \beta \sum_{i=1}^n x(y - \alpha - \beta x) - \bar{y} \sum_{i=1}^n (y - \alpha - \beta x) = 0 \end{aligned}$$

λόγω των κανονικών εξισώσεων:

$$\sum_{i=1}^n (y - \alpha - \beta x) = 0 \quad \text{και} \quad \sum_{i=1}^n x(y - \alpha - \beta x) = 0$$

Έτσι, θα προκύψει τελικά:

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Ομοίως συμβαίνει και στην περίπτωση της παραβολής.

Σημείωση: Ο συντελεστής προσδιορισμού r^2 μπορεί να υπολογιστεί και μέσω του τύπου $r^2 = \beta * \beta'$, κάτι που ισχύει μόνο στην περίπτωση της ευθείας και όχι της παραβολής. [7]

3.3.7 Απλή γραμμική συσχέτιση (r)

Υπάρχει το πρόβλημα του προσδιορισμού της έντασης που τυχόν δημιουργείται μεταξύ των μεταβλητών y και x που συσχετίζονται.

Η συσχέτιση μεταξύ δύο μεταβλητών μπορεί να είναι πλήρης (ή τέλεια), μερική ή μηδενική.

Πλήρη συσχέτιση έχουμε όταν ολόκληρη η μεταβλητικότητα των τιμών της y εξηγείται από την μεταβλητικότητα των τιμών της x . Π.χ. η συσχέτιση μεταξύ της τάσης U και της έντασης I του ηλεκτρικού ρεύματος. Στην περίπτωση αυτή, οι μεταβολές της τάσης U οφείλονται στις μεταβολές μόνο της έντασης I .

Μερική συσχέτιση έχουμε όταν μέρος μόνο της μεταβλητικότητας της y οφείλεται στην μεταβλητικότητα των τιμών της x , ενώ το υπόλοιπο μέρος της μεταβλητικότητας της y οφείλεται σε άλλους παράγοντες που δεν περιελήφθηκαν

στην εξίσωση παλινδρόμησης.

Τέλος, στην **μηδενική συσχέτιση** η μεταβλητικότητα των τιμών της y δεν οφείλεται στην μεταβλητικότητα των τιμών της x . [9]

3.3.8 Μέτρηση της συσχέτισης

Η συσχέτιση, και συγκεκριμένα ο βαθμός αυτής μετριέται με τον συντελεστή συσχέτισης r , ο οποίος δίνεται από τον τύπο:

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{nS_x S_y}$$

Ο συντελεστής συσχέτισης r έχει τις εξής ιδιότητες:

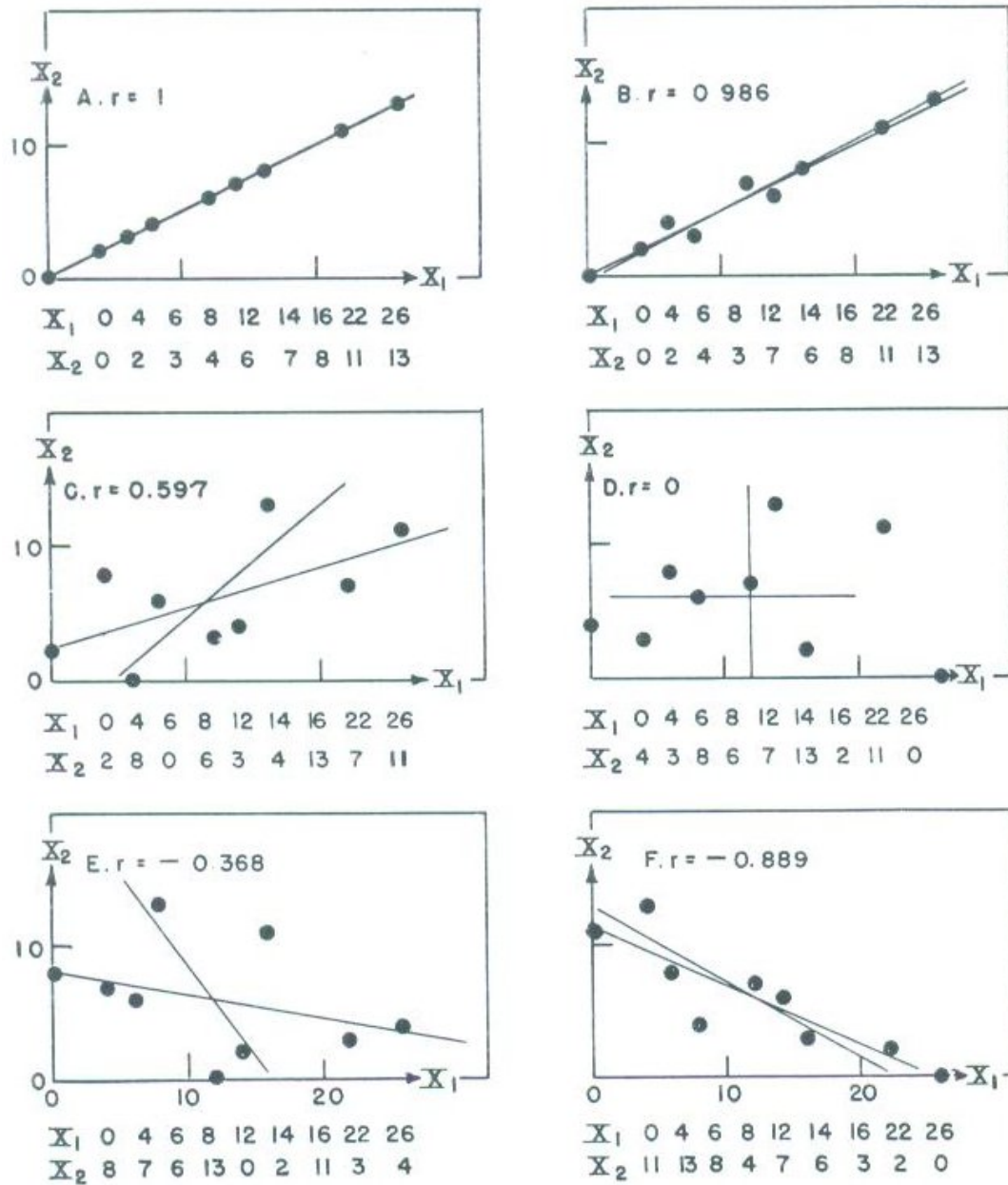
- Είναι ανεξάρτητος από την μονάδα μέτρησης καθεμιάς των μεταβλητών x και y .
- Είναι ανεξάρτητος από την αρχή μέτρησης των μεταβλητών, που σημαίνει ότι αν π.χ. συσχετισθούν αντί των αρχικών τιμών x_i και y_i των δύο μεταβλητών, οι αντίστοιχες αποκλίσεις $x_i - \bar{x}$ και $y_i - \bar{y}$, δεν θα αλλάξει η τιμή του.

Υπάρχουν λόγοι που επιβάλλουν στον παραπάνω τύπο αντί n , να θέτουμε $n - 1$.

Ο συντελεστής συσχέτισης παίρνει τιμές από -1 μέχρι 1 , δηλαδή $-1 \leq r \leq 1$.

Όταν τα σημεία (x_i, y_i) βρίσκονται πάνω σε ευθεία γραμμή, τότε η τιμή του r είναι 1 (περίπτωση όπου αυξανόμενου του x αυξάνεται και το y), ή είναι -1 (περίπτωση όπου αυξανόμενου του x μειώνεται το y). Στην πρώτη περίπτωση έχουμε πλήρη θετική συσχέτιση και στην δεύτερη πλήρη αρνητική συσχέτιση. Οι αντίθετες τιμές του r (π.χ. $r = 0.75$ και $r = -0.75$) δηλώνουν την ίδια ένταση συσχέτισης, με την διαφορά ότι η πρώτη είναι θετική συσχέτιση (δηλαδή αυξανόμενου του x αυξάνεται και το y), και η δεύτερη είναι αρνητική.

Κάποιες περιπτώσεις φαίνονται στο ακόλουθο σχήμα:



Γράφημα 3.5: Διαγράμματα διασποράς με διάφορες συσχετίσεις [10]

Επίσης, θα πρέπει να σημειωθεί ότι τα r και β είναι πάντα ομόσημα.

ΚΕΦΑΛΑΙΟ 4

ΛΕΠΤΟΜΕΡΗΣ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ

Για την βαθύτερη κατανόηση του παρόντος λογισμικού, θα γίνει μία αναφορά στα βήματα για την δημιουργία μίας βάσης δεδομένων, όπως και μία προσπάθεια επίδειξης των χαρακτηριστικών του προγράμματος καθώς ο εκτελεστής του προγράμματος θα κάνει την εκάστοτε επιλογή από τις διαθέσιμες.

4.1 Δημιουργία βάσης δεδομένων

Καταρχάς, θα πρέπει να δημιουργηθεί λογαριασμός σε ένα website με υποστήριξη βάσης δεδομένων, και πιο συγκεκριμένα MySQL server. Για τις ανάγκες του παρόντος παραδείγματος επιλέχθηκε η ιστοσελίδα <http://www.db4free.net/>.

Μετά την ολοκλήρωση της εγγραφής στο εκάστοτε website, ο χρήστης διαθέτει τα απαιτούμενα για την σύνδεση στην βάση δεδομένων, δηλαδή όνομα χρήστη (username), κωδικό για να την προσπελάσει (password), και φυσικά το όνομα της βάσης δεδομένων της επιλογής του και την διεύθυνση του εξυπηρετητή (hostname).

Στο παρόν παράδειγμα, αυτά είναι τα εξής:

```
hostname: db4free.net  
username: ptuxiak  
password: slide13  
database: ptuxiakidb  
port: 3307
```

Σε περίπτωση που η βάση δεδομένων είναι απομακρυσμένη και όχι τοπική, αλλάζει και ο αριθμός της θύρας (port) που θα χρησιμοποιηθεί. Για απομακρυσμένες συνδέσεις, είναι 3306 ή 3307, ενώ για τοπικές 0 (μηδέν).

Σημειώνεται πως στο πρόγραμμα, τα παραπάνω χαρακτηριστικά (credentials) έχουν εισαχθεί χρησιμοποιώντας την εντολή/οδηγία προς τον προεπεξεργαστή define στην αρχή του κώδικά του, με σκοπό την όσο το δυνατόν πιο δυναμική λειτουργία του.

4.2 Επεξήγηση επιλογών του λογισμικού

Όταν θα εκτελέσει ο Εξεταστής το πρόγραμμα (το οποίο θα προκύπτει από την μεταγλώπτιση του admint.c), θα του εμφανίζονται 6 επιλογές:

1) Δημιουργία βάσης δεδομένων με ερωτήσεις τύπου Multiple Choice

Επιλογή για την δημιουργία νέου ερωτηματολόγιου -είτε για την πρώτη φορά που θα εκτελεστεί το πρόγραμμα, είτε για εισαγωγή νέων ερωτήσεων.

Εμφανίζεται το ακόλουθο μήνυμα:

Παρακαλώ πληκτρολογήστε το όνομα του αρχείου, στο οποίο βρίσκονται οι ερωτήσεις

(Πρέπει να βρίσκεται στο ίδιο directory με το παρόν εκτελέσιμο):

Προφανώς θα πρέπει να έχει δημιουργηθεί ένα αρχείο με ερωτήσεις, τις αντίστοιχες πιθανές απαντήσεις τους, το γράμμα που αντιστοιχεί στην σωστή απάντηση, αλλά και μια υποδείξη/απόδειξη αυτής, με την ακόλουθη μορφή:

Ερώτηση1 <πλήκτρο Tab> ΑπάντησηΑ <Tab> ΑπάντησηΒ <Tab> ΑπάντησηC
<Tab> ΑπάντησηD <Tab> Γράμμα Σωστής Απάντησης <Tab> Απόδειξη/Υπόδειξη
<Enter>

Ερώτηση2 <πλήκτρο Tab> ΑπάντησηΑ <Tab> ΑπάντησηΒ <Tab> ΑπάντησηC
<Tab> ΑπάντησηD <Tab> Γράμμα Σωστής Απάντησης <Tab> Απόδειξη/Υπόδειξη
<Enter>

...

ΤελευταίαΕρώτηση <πλήκτρο Tab> ΑπάντησηΑ <Tab> ΑπάντησηΒ <Tab>
ΑπάντησηC <Tab> ΑπάντησηD <Tab> Γράμμα Σωστής Απάντησης <Tab>

Απόδειξη/Υπόδειξη

Αφού λοιπόν ο χρήστης πληκτρολογήσει το όνομα του αρχείου, το περιεχόμενο αυτού θα εισαχθεί στην βάση δεδομένων που δημιουργήθηκε με σκοπό να χρησιμοποιηθεί ως εξεταστικό υλικό για τους Εξεταζόμενους.

2) Δημιουργία πίνακα Απαντήσεων Φοιτητών - στατιστικών στοιχείων

Επιλογή για την δημιουργία νέου πίνακα με τα στατιστικά στοιχεία των Εξεταζόμενων που κλήθηκαν να απαντήσουν το ερωτηματολόγιο. Ομοίως με το 1), θα επιλεγεί είτε για την πρώτη φορά που θα εκτελεστεί το πρόγραμμα, ώστε να κατασκευαστεί νέος άδειος πίνακας, ο οποίος θα γεμίζει καθώς οι Εξεταζόμενοι θα ολοκληρώνουν το ερωτηματολόγιο, είτε στην περίπτωση που ο Εξεταστής θέλει να σβήσει τον προηγούμενο πίνακα (λόγου χάρη, γιατί έληξε το τρέχον εξάμηνο, άρα δεν χρειάζεται να έχει καταγεγραμμένα τα στατιστικά αυτά).

Επειδή θα διαγραφεί ο προηγούμενος πίνακας, αν υπάρχει, όταν ο χρήστης επιλέξει το 2), θα εμφανιστεί μήνυμα επιβεβαίωσης.

3) Εμφάνιση πίνακα Απαντήσεων Φοιτητών - στατιστικών στοιχείων

Επιλογή για την εμφάνιση του παραπάνω πίνακα.

Εμφανίζονται τα στοιχεία του Φοιτητή, δηλαδή ο αριθμός μητρώου του και το εξάμηνό του, καθώς και το πλήθος των σωστών απαντήσεων του, αλλά και η ημερομηνία και η ώρα που ολοκλήρωσε το ερωτηματολόγιο.

4) Εξαγωγή πίνακα Απαντήσεων Φοιτητών - στατιστικών στοιχείων (Back up)

Επιλογή για την τοπική αποθήκευση του εν λόγω πίνακα (ο Εξεταστής ίσως χρειάζεται την βάση δεδομένων που έχει δημιουργηθεί κατά την ολοκλήρωση του ερωτηματολογίου από τους Εξεταζόμενους).

Αποθηκεύεται σε τέτοια μορφή, που αργότερα να μπορεί να αντληθεί μέσω της MySQL.

5) Εμφάνιση στατιστικών στοιχείων

Επιλογή για την εμφάνιση των μαθηματικών συναρτήσεων με σκοπό την εξαγωγή συμπερασμάτων από τον Εξεταστή, όπως για παράδειγμα θα μπορούσαν οι Εξεταζόμενοι μικρότερων εξαμήνων να μην είχαν τόσο καλά στατιστικά, ενώ εκείνοι των μεγαλύτερων εξαμήνων να έχουν πάει καλύτερα, ή ότι ο μέσος όρος των σωστών απαντήσεων ήταν χαμηλός συγκριτικά με άλλα εξάμηνα, άρα το ερωτηματολόγιο είναι αυξημένης δυσκολίας, κ.τ.λ.

Ακόμα, ο Εξεταστής θα μπορούσε να προβλέψει σε ένα βαθμό τον βαθμό επιτυχίας των Εξεταζόμενων με βάση το εξάμηνό τους μέσω μαθηματικού μοντέλου.

6) Έξοδος

Έξοδος από το πρόγραμμα.

Για να υπάρξει μια βάση δεδομένων με τα στατιστικά στοιχεία των Εξεταζόμενων, θα πρέπει βεβαίως εκείνοι να ολοκληρώσουν το ερωτηματολόγιο. Αυτό θα γίνει όταν εκτελέσουν το αρχείο που θα προκύψει από την μεταγλώττιση του `user.t.c`.

Όταν γίνει αυτό, το πρόγραμμα θα ζητήσει την εισαγωγή του αριθμού μητρώου του Εξεταζόμενου και το εξάμηνό του.

Μετά θα ξεκινήσει το ερωτηματολόγιο, ενώ αφού ολοκληρωθούν οι ερωτήσεις, θα εμφανιστεί το πλήθος των σωστών απαντήσεων του τρέχοντα Εξεταζόμενου, και τέλος, τα ανωτέρω στοιχεία του θα εισαχθούν στην βάση δεδομένων ως μία νέα εγγραφή της.

Σημειώνεται πως σε περίπτωση λάθος εισαγωγής σε οποιοδήποτε σημείο του προγράμματος, υπάρχουν έλεγχοι εγκυρότητας της εκάστοτε επιλογής.

ΚΕΦΑΛΑΙΟ 5

ΜΕΛΛΟΝΤΙΚΕΣ ΒΕΛΤΙΩΣΕΙΣ

Το παρόν λογισμικό επιδέχεται κάποιες βελτιώσεις, οι οποίες όμως θα το απομάκρυναν από τον αρχικό εκπαιδευτικό σκοπό του.

Καταρχάς, το πρόγραμμα μπορεί να παραλληλοποιηθεί, ώστε να εκμεταλλεύεται πλήρως τους σημερινούς πολυπύρηνους επεξεργαστές. Υπάρχουν μέθοδοι που χρησιμοποιούν παράλληλη επεξεργασία, όπως η χρήση OpenMP, ένα σύνολο συναρτήσεων και οδηγιών προς τον μεταγλωττιστή. Κομμάτι προς παραλληλοποίηση θα μπορούσε να είναι η διαδικασία εισαγωγής ερωτήσεων στην βάση δεδομένων, ή ο υπολογισμός των στατιστικών στοιχείων -ανάλογα με το πλήθος των εγγραφών, ο κάθε επεξεργαστής να αναλάβει τον υπολογισμό ενός κομματιού (chunk) για την ταχύτερη λειτουργία του προγράμματος.

Ένας άλλος τομέας στον οποίο θα μπορούσε να γίνει αποδοτικότερη η εφαρμογή, είναι ο προγραμματισμός της σε μία διαφορετική γλώσσα, υψηλότερου επιπέδου, για μεγαλύτερη ευελιξία, διότι αν και η C επιτρέπει την όσο το δυνατόν βαθύτερη κατανόηση του εκάστοτε προβλήματος και μεγαλύτερη δύναμη στα χέρια ενός προγραμματιστή, εντούτοις απαιτούνται πολλές γραμμές κώδικα για φαινομενικά απλές λειτουργίες, οι οποίες σε μεταγενέστερες γλώσσες προγραμματισμού θα απαιτούσαν λίγες εντολές, με αποτέλεσμα την αμεσότερη πρόσβαση σε εκείνο. Επίσης, λόγω αυτού, το πρόγραμμα θα είναι ελαφρύτερο. Ακόμα, επιτυγχάνεται περισσότερη φορητότητα, μιας και ένα λογισμικό γραμμένο σε νεότερη γλώσσα προσφέρει μεγαλύτερη γκάμα πλατφορμών στις οποίες θα μπορεί να εκτελεστεί, ενώ θα είναι εφικτή η προσθήκη κάποιων επιπλέον χαρακτηριστικών, τα οποία με την χρήση της γλώσσας C θα επιβάρυναν κατά πολύ το πρόγραμμα, όπως η δυνατότητα εισαγωγής εικόνων ως ερώτηση ή απάντηση στην βάση δεδομένων (πράγμα που φυσικά πρέπει να υποστηρίζει και ο server στον οποίο θα αποθηκεύεται αυτή). Για παράδειγμα, το παρόν πρόγραμμα εκτελείται μόνο σε περιβάλλον Unix, ένα λειτουργικό σύστημα που δεν είναι τόσο διαδεδομένο όσο το Windows OS. **[11]**

ΠΑΡΑΡΤΗΜΑ

Παράθεση κώδικα C

Στο παράρτημα αυτό παρατίθεται ο κώδικας της γλώσσας προγραμματισμού C που γράφτηκε.

Αρχείο admint.c

```
#define HOSTNAME "db4free.net"
#define USERNAME "tei34734"
#define PASSWORD "slide13"
#define DATABASE "tei34734db"
#define PORT "3306"

#include <stdio.h>
#include <stdlib.h>
#include <my_global.h>
#include <mysql.h>
#include <string.h>

float n = 0; // πλήθος φοιτητών

main(int argc, char *argv[])
{
    FILE *fp;
    char ch[1000], ch2[1080], filename[30], ent, sure;
    char sqlQ[] = "CREATE TABLE Questions(Id INT PRIMARY KEY
AUTO_INCREMENT, Erwtisi LONGTEXT, ApA LONGTEXT, ApB LONGTEXT,
ApC LONGTEXT, ApD LONGTEXT, Swsto TEXT, Apod LONGTEXT)";
    char sqlS[] = "CREATE TABLE Students(AM INT PRIMARY KEY, Examino
TINYINT, SwstesAp SMALLINT, Pote DATETIME)";
    char *tok;
```

```
int i = 1, j, ep, flag = 0;
MYSQL *con = mysql_init(NULL);      // Δημιουργία mysql object (C99)
int empty_stdin();                  // Prototypes
void normalEqu(float x[], float y[]); // συναρτήσεις

MYSQL_RES *result = mysql_store_result(con); // Δομές MySQL
κατάλληλες για
MYSQL_ROW row;                      // την εξαγωγή
αποτελεσμάτων

if (con == NULL)
{
    printf("\nΑποτυχία της mysql_init()\n\n");
    exit(1);
}

do
{
    if (flag)
        printf("\nΛάθος εισαγωγή - Ξαναπροσπαθήστε.\n");

    printf("\nΕπιλογές:\n\n");
    printf(" 1) Δημιουργία βάσης δεδομένων με ερωτήσεις τύπου Multiple
Choice\n");
    printf(" 2) Δημιουργία πίνακα Απαντήσεων Φοιτητών - στατιστικών
στοιχείων\n");
    printf("\t---ΠΡΟΣΟΧΗ διαγραφή προηγούμενου πίνακα αν
υπάρχει\n");
    printf(" 3) Εμφάνιση πίνακα Απαντήσεων Φοιτητών - στατιστικών
στοιχείων\n");
    printf(" 4) Εξαγωγή πίνακα Απαντήσεων Φοιτητών - στατιστικών
στοιχείων (Back up)\n");
    printf(" 5) Εμφάνιση στατιστικών στοιχείων\n");
    printf(" 6) Έξοδος\n");
```



```
        printf("\nΔώστε το νούμερο της επιλογής σας: ");
        flag = 1;
    }
    while (((scanf("%d%c", &ep, &ent) != 2 || ent != '\n') && empty_stdin()) || ep
< 1 || ep > 6);

    // Δυναμική σύνδεση μέσω #define
    if (mysql_real_connect(con, HOSTNAME, USERNAME, PASSWORD,
DATABASE, 3306, NULL, 0) == NULL) // Sundesi db k elegchos
    {
        printf("\nΣφάλμα κατά τη σύνδεση στην ΒΔ\n\n");
        mysql_close(con);
        exit(1);
    }

    switch(ep)
    {
        case 1:
            // Δημιουργία βάσης δεδομένων με ερωτήσεις τύπου Multiple Choice

            if (mysql_query(con, "DROP TABLE IF EXISTS Questions")) //
Αν υπάρχει ήδη πίνακας Questions, διαγραφή του
            {
                printf("\nΉδη υπάρχων πίνακας Questions - Τερματισμός
προγράμματος\n\n");
                mysql_close(con);
                exit(1);
            }

            if (mysql_query(con, sqlQ)) // Δημιουργία πίνακα Questions
            {
                printf("\nΣφάλμα κατά την δημιουργία πίνακα Questions -
Τερματισμός προγράμματος\n\n");
                mysql_close(con);
            }
        }
    }
}
```

```
        exit(1);
    }

    printf("\nΠαρακαλώ πληκτρολογήστε το όνομα του αρχείου, στο
οποίο βρίσκονται οι ερωτήσεις");
    printf("\n(Πρέπει να βρίσκεται στο ίδιο directory με το παρόν
εκτελέσιμο): ");
    scanf("%s", filename);
    fp = fopen(filename, "r");
    if (fp == NULL)
    {
        printf("\nΣφάλμα κατά το άνοιγμα του αρχείου, βεβαιωθείτε ότι
πληκτρολογήσατε σωστά το ");
        printf("όνομά του, καθώς και ότι βρίσκεται στο ίδιο directory με
το παρόν πρόγραμμα.\n\n");
        exit(1);
    }

    int records = 0, c;
    do
    {
        c = fgetc(fp);
        if(c == '\n')
            records++;
    }
    while (c != EOF);

    fseek(fp, 0, SEEK_SET); // Επιστροφή του file pointer στην αρχή
του αρχείου

    printf("\nΕισαγωγή ερωτήσεων στην Βάση Δεδομένων - Παρακαλώ
περιμένετε...\n");

    while (i <= records)
```

```
{
    printf("Εισαγωγή ερώτησης %d από %d...\r", i, records);

    if (fgets(ch, sizeof(ch), fp) != NULL)
    {
        // Χρήση της strtok() για τον διαχωρισμό της εκάστοτε
        ερώτησης από τις
        // απαντήσεις: Κατά την πρώτη κλήση της συνάρτησης,
        ορίζεται το αρχικό
        // σημείο ανάγνωσης του κειμένου προς σάρωση για
        εύρεση συγκεκριμένων
        // χαρακτήρων (στην παρούσα περίπτωση του
        χαρακτήρα του πλήκτρου Tab (\t))
        strtok(ch, "\t");

        snprintf(ch2, strlen(ch)+sizeof("INSERT INTO
Questions(Erwtisi) VALUES("),
        "INSERT INTO Questions(Erwtisi) VALUES('%s'", ch);
        // Pros8esi entolwn stin erwtisi

    }

    if (mysql_query(con, ch2)) // Προσθήκη της ερώτησης στην db
    {
        printf("\nΣφάλμα κατά την εισαγωγή ερώτησης %d -
πεδίο <<Ερώτηση>>\n\n", i);
        mysql_close(con);
        exit(1);
    }

    for (j = 0; j < 4; j++)
    {
        // Εντολές για την εισαγωγή στα πεδία των
```

Απαντήσεων:

```
        // Κάθε επόμενη φορά που καλείται η strtok(),
επιστρέφει την αρχή
        // του string που ορίζεται μεταξύ των επιθυμητών
χαρακτήρων (\t)
        tok = strtok(NULL, "\t");

        snprintf(ch2, sizeof(ch)+sizeof("UPDATE Questions
SET ApX = " WHERE Id = "),
        "UPDATE Questions SET Ap%c = '%s' WHERE Id =
%d", 'A'+j, tok, i); // Πρόσθεση εντολών στην απάντηση

        if (mysql_query(con, ch2)) // Προσθήκη της εκάστοτε
απάντησης στην db
        {
                printf("\nΣφάλμα κατά την εισαγωγή ερώτησης
%d - πεδίο <<Πιθανή απάντηση %d>>\n\n", i, j);
                mysql_close(con);
                exit(1);
        }
}

// Χρήση της strtok() για την εισαγωγή του string (πεδίο
Swsto),
        // το οποίο ορίζεται μεταξύ των επιθυμητών χαρακτήρων (\t)
        tok = strtok(NULL, "\t");

        snprintf(ch2, sizeof(tok)+sizeof("UPDATE Questions SET
Swsto = " WHERE Id = "),
        "UPDATE Questions SET Swsto = '%s' WHERE Id = %d", tok,
i); // Πρόσθεση εντολών στην σωστή απάντηση

        if (mysql_query(con, ch2)) // Προσθήκη του γράμματος της
σωστής απάντησης στην db
        {
```

```
        printf("\nΣφάλμα κατά την εισαγωγή ερώτησης %d -
πεδίο <<Σωστό>>\n\n", i);
        mysql_close(con);
        exit(1);
    }

    // Χρήση της strtok(), αυτή τη φορά για την εισαγωγή του
επιθυμητού
    // string (πεδίο Apod), το οποίο ορίζεται μέχρι το τέλος της
γραμμής (\n)
    tok = strtok(NULL, "\n");

    snprintf(ch2, sizeof(ch)+sizeof("UPDATE Questions SET Apod
= " WHERE Id = "),
    "UPDATE Questions SET Apod = '%s' WHERE Id = %d", tok,
i);    // {Πρόσθεση εντολών στην απόδειξη

    if (mysql_query(con, ch2)) // Προσθήκη της απόδειξης στην db
    {
        printf("\nΣφάλμα κατά την εισαγωγή ερώτησης %d -
πεδίο <<Απόδειξη>>\n\n", i);
        mysql_close(con);
        exit(1);
    }

    i++;

}

fclose(fp);

// |-----|
// ΤΕΛΟΣ ΕΙΣΑΓΩΓΗΣ ΕΡΩΤΑΠΑΝΤΗΣΕΩΝ - ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ
ΔΗΜΙΟΥΡΓΗΘΗΚΕ
```

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

```
// |-----|

printf("\n---Τέλος εισαγωγής - Βάση Δεδομένων δημιουργήθηκε---
\n\n");

break;

case 2:
// Δημιουργία πίνακα Students - πρέπει να επιβεβαιώσει ο χρήστης
γι'αυτή
// την επιλογή, γιατί θα διαγραφεί ο προηγούμενος πίνακας (αν
υπάρχει)

printf("\nΑν υπάρχει προηγούμενος πίνακας Απαντήσεων Φοιτητών,
θα διαγραφεί!!\n");

do
{
printf("\nΕίστε σίγουροι ότι θέλετε να προχωρήσετε στην
δημιουργία τέτοιου πίνακα; [y/n] ");

}
while (((scanf("%c%c", &sure, &ent)!=2 || ent!='\n') && empty_stdin())
|| sure!='n' && sure!='y' && sure!='N' && sure!='Y');

if (sure == 'y' || sure == 'Y')
{
if (mysql_query(con, "DROP TABLE IF EXISTS Students")) //
Αν υπάρχει ήδη πίνακας Students, διαγραφή του
{
printf("\nΉδη υπάρχων πίνακας - Τερματισμός
προγράμματος\n\n");

mysql_close(con);
exit(1);
}
```

```
    }

    if (mysql_query(con, sqlS))      // Δημιουργία πίνακα
Students
    {
        printf("\nΣφάλμα κατά την δημιουργία πίνακα Students -
Τερματισμός προγράμματος\n\n");
        mysql_close(con);
        exit(1);
    }

    printf("\nΔημιουργία πίνακα Απαντήσεων Φοιτητών
επιτυχής\n\n");
    }
    break;

case 3:
// Εμφάνιση στατιστικών στοιχείων - απαντήσεων από τους φοιτητές

printf("\n --- Πίνακας σωστών απαντήσεων φοιτητών ---\n");
printf("\nAM\tΕξάμηνο\tΣωστές\tΠότε έκανε το τεστ\n\t\t\t\tΗμερομηνία
Ώρα\n");

if (mysql_query(con, "SELECT * FROM Students"))
{
    printf("\nΣφάλμα κατά την εμφάνιση πίνακα Απαντήσεων
Φοιτητών\n");

    mysql_close(con);
    exit(1);
}

if ((result = mysql_store_result(con)) == NULL)
{
    printf("\nΣφάλμα κατά την εξαγωγή αποτελέσματος από την
```

```
Βάση Δεδομένων\n");
```

```
}
```

```
int num_fields = mysql_num_fields(result); // Αριθμός πεδίων του  
πίνακα Φοιτητών
```

```
while ((row = mysql_fetch_row(result))) // Εντολές για κάθε γραμμή  
του πίνακα Φοιτητών
```

```
{
```

```
for (i = 0; i < num_fields; i++)
```

```
{
```

```
printf("%s\t", row[i] ? row[i] : "NULL");
```

```
}
```

```
printf("\n");
```

```
}
```

```
printf("\n");
```

```
mysql_free_result(result);
```

```
break;
```

```
case 4:
```

```
// Back up της database
```

```
printf("\nΔημιουργία back up με όνομα αρχείου BackUpStats.sql στο  
ίδιο directory με το παρόν πρόγραμμα...");
```

```
if (system("mysqldump -h " HOSTNAME " -u " USERNAME " -P "  
PORT " " DATABASE " -p" PASSWORD " Students > BackUpStats.sql"))
```

```
printf("α\n");
```

```
printf("επιτυχής.\n\n");
```

```
break;
```

```
case 5:
```



```
// Εμφάνιση στατιστικών

if (mysql_query(con, "SELECT COUNT(*) FROM Students"))
{
    printf("\nΣφάλμα κατά τον υπολογισμό πλήθους φοιτητών που
πραγματοποίησαν το ερωτηματολόγιο.\n");
    mysql_close(con);
    exit(1);
}

result = mysql_store_result(con);
row = mysql_fetch_row(result);
n = atoi(row[0]); // Πλήθος φοιτητών
mysql_free_result(result);

float *exam, *swst;

exam = (float *)malloc(n * sizeof(float)); // Δημιουργία πινάκων με
μέγεθος το πλήθος των φοιτητών
swst = (float *)malloc(n * sizeof(float)); // και στοιχεία τις σωστές
απαντήσεις και τα εξάμηνά τους

if (mysql_query(con, "SELECT Examino, SwstesAp FROM
Students"))
{
    printf("\nΣφάλμα κατά την επιλογή Εξαμήνου, ΣωστώνΑπ από
το ερωτηματολόγιο.\n");
    mysql_close(con);
    exit(1);
}

result = mysql_store_result(con);

for (i = 0; row = mysql_fetch_row(result); i++)
```

```
        {
            exam[i] = atof(row[0]);
            swst[i] = atof(row[1]);
        }

        mysql_free_result(result);

        printf("\nΚανονικές εξισώσεις - x(εξάμηνο), y(σωστές
απαντήσεις)\n");
        printf("-----\n");
        normalEqu(exam,swst);

        printf("\nΚανονικές εξισώσεις - x(σωστές απαντήσεις),
y(εξάμηνο)\n");
        printf("-----\n");
        normalEqu(swst,exam);

        free (exam);
        free (swst);

        break;

        case 6:
            // Έξοδος
            printf("\n<<<< Τερματισμός >>>>\n\n");
            exit(0);
            break;
    }

    mysql_close(con);
}
```

```
// Σε περίπτωση λανθασμένης εισαγωγής απ'τον χρήστη, πρέπει να "αδειάσει"  
το  
// buffer του stdin για να δεχτεί νέα απάντηση.  
// Χωρίς τη χρήση της συνάρτησης, το πρόγραμμα στην επόμενη επανάληψη θα  
// δεχόταν ως όρισμα τη συνέχεια του string που δόθηκε στην προηγούμενη.
```

```
int empty_stdin()  
{  
    while (getchar()!='\n');  
    return 1;  
}
```

```
void normalEqu(float x[], float y[])  
{  
    float D, Da, Db, Dc; // Ορίζουσες  
    int i;  
    float a, b, c; // Συντελεστές κανονικών εξισώσεων  
    float sx = 0., sy = 0., sx2 = 0., sx3 = 0., sx4 = 0., sxy = 0., sx2y = 0., sy2 =  
0., xm, ym;  
    float s2x; // Διακύμανση  
    float ta; // Τυπική απόκλιση  
    float r; // Συντελεστής συσχέτισης  
    float r2; // Συντελεστής προσδιορισμού  
  
    for (i = 0; i < n; i++)  
    {  
        sx = sx + x[i];  
        sy = sy + y[i];  
        sx2 = sx2 + x[i]*x[i];  
        sx3 = sx3 + x[i]*x[i]*x[i];  
        sx4 = sx4 + x[i]*x[i]*x[i]*x[i];
```

```
sxy = sxy + x[i]*y[i];
sx2y = sx2y + x[i]*x[i]*y[i];
sy2 = sy2 + y[i]*y[i];
}

// Για ευθεία

// y = a + bx

// Μνημονικοί κανόνες για εύρεση κανονικών εξισώσεων
// sy = na + bsx    (1)
// sxy = asx + bsx2

// D = ..., Dx = ..., Dy = .. (2x2)

D = n*sx2 - sx*sx;
Da = sy*sx2 - sxy*sx;
Db = n*sxy - sx*sy;

// Το b δεν χρειάζεται να το βρω με Db/D, αλλά:
// (1) -> (διά n) : γμέσο = a + b*χμέσο
// αφαίρεση κατά μέλη: (γμέσο -> y')
// y - y' = b*(x-x')
// b = (y-sy/n) / (y-sx/n);

a = Da / D;
b = Db / D;

printf("\nΓια χάραξη ευθείας: y = %.4f + (%.4f) x", a, b);

// Ο ορισμός για την διακύμανση είναι  $\Sigma[(x-\chi\mu\epsilon\sigma\omicron)^2] / n$ , όμως
// έτσι θα πρέπει ο πίνακας των x να προσπελαστεί 2 φορές, μία
// για την εύρεση του x μέσου, κι άλλη μία φορά για την εύρεση της
// διαφοράς (x - χμέσο), οπότε δεν είναι τόσο αποδοτικός τρόπος.
```

```
// Αν όμως αναπτυχθεί το τετράγωνο μέσα στο άθροισμα:  
//  $\Sigma[(x^2 - 2*x*χμέσο + χμέσο^2)] / n =$  (περνάω το  $\Sigma$  στα επιμέρους)  
//  $= (\Sigma[x^2]) - 2*χμέσο*\Sigma[x] + n*χμέσο^2) / n =$  (διαιρώ διά n επιμέρους)  
//  $= \Sigma[x^2]/n - 2*χμέσο^2 + χμέσο^2 =$   
//  $= \Sigma[x^2]/n - χμέσο^2$   
// Έτσι, ο πίνακας των x θα προσπελαστεί μόνο μία φορά.
```

```
s2x = sx2/n - xm*xm;
```

```
ta = sqrt(s2x); // Εύρεση τυπικής απόκλισης
```

```
// Για παραβολή
```

```
//  $y = a + bx + cx^2$ 
```

```
// Μνημονικοί κανόνες για εύρεση κανονικών εξισώσεων
```

```
//  $sy = na + bsx + csx^2$  (βάζω  $\Sigma$  και στα δύο μέλη)
```

```
//  $sxy = asx + bsx^2 + csx^3$  (πολλαπλασιάζω επί x και βάζω  $\Sigma$  και στα δύο μέλη)
```

```
//  $sx^2y = asx^2 + bsx^3 + csx^4$  (πολλαπλασιάζω επί  $x^2$  και βάζω  $\Sigma$  και στα δύο μέλη)
```

```
D = n*sx2*sx4 + sx*sx3*sx2 + sx2*sx*sx3 - sx2*sx2*sx2 - n*sx3*sx3 -  
sx4*sx*sx;
```

```
Da = sy*sx2*sx4 + sx*sx3*sx2y + sx2*sxy*sx3 - sx2y*sx2*sx2 - sx3*sx3*sy  
- sx4*sxy*sx;
```

```
Db = n*sxy*sx4 + sy*sx3*sx2 + sx2*sx*sx2y - sx2*sxy*sx2 - sx2y*sx3*n -  
sx4*sx*sy;
```

```
Dc = n*sx2*sx2y + sx*sxy*sx2 + sy*sx*sx3 - sx2*sx2*sy - sx3*sxy*n -  
sx2y*sx*sx;
```

```
a = Da / D;
```

```
b = Db / D;
```

```
c = Dc / D;
```

```
// Για εύρεση ρ^2 (μεταβάλλεται από -1 ως 1):
// πολλαπλασιασμός συντελεστών κατεύθυνσης των κανονικών εξισώσεων
// άρα: b = Db / D = (n*sxy - sx*sy) / (n*sx2 - sx*sx);
// b * b' = (n*sxy - sx*sy)^2 (δεν αλλάζει ο αριθμητής) / ((n*sx2 - sx*sx) * (n*sy2 - sy*sy))
r = (n*sxy - sx*sy) / sqrt((n*sx2 - sx*sx) * (n*sy2 - sy*sy)); // Συντελεστής
συσχετίσεως
//r2 = (n*sxy - sx*sy) * (n*sxy - sx*sy) / ((n*sx2 - sx*sx) * (n*sy2 - sy*sy));

r2=r*r;          // Συντελεστής προσδιορισμού

printf("\nΓια χάραξη παραβολής: y = %.4f + (%.4f) x + (%.4f) x^2\n", a, b, c);
printf("\nΜέσος Όρος x (x̄): %.4f, Μέσος Όρος y (ȳ): %.4f", xm = sx / n, ym
= sy / n);
printf("\nΔιακύμανση (σ^2): %.4f, Τυπική Απόκλιση (σ): %.4f", s2x, ta);
printf("\nΣυντελεστής Συσχετίσεως (r): %.4f, Συντελεστής προσδιορισμού
(r^2): %.4f", r, r2);
printf("\nΠλήθος: %.0f\n", n);

sx = 0. , sy = 0. , sx2 = 0. , sx3 = 0. , sx4 = 0. , sxy = 0. , sx2y = 0., sy2 = 0.,
xm = 0, ym = 0;

}
```

Αρχείο usert.c

```
#define HOSTNAME "db4free.net"
#define USERNAME "tei34734"
#define PASSWORD "slide13"
#define DATABASE "tei34734db"
#define PORT 3307
```

```
// Το παρόν πρόγραμμα δημιουργεί ένα εκτελέσιμο, το οποίο απευθύνεται στον
```

τελικό χρήστη (φοιτητή).

```
#include <stdio.h>
#include <stdlib.h>
#include <my_global.h>
#include <mysql.h>
#include <string.h>
#include <time.h>
#include <ctype.h>

main()
{
    int empty_stdin(); // Prototype συνάρτησης
    char ch[500], apant, ent;
    int i, rows, r, AM, exam, swstes = 0;
    MYSQL *con = mysql_init(NULL);

    // dunamiki sundesi mesw #define
    if (mysql_real_connect(con, HOSTNAME, USERNAME, PASSWORD,
DATABASE, 3306, NULL, 0) == NULL) // Sundesi db k elegchos
    {
        printf("\nΣφάλμα κατά τη σύνδεση στην ΒΔ - Τερματισμός
προγράμματος.\n\n");
        mysql_close(con);
        exit(1);
    }

    do
    {
        printf("\nΠαρακαλώ δώσε Αριθμό Μητρώου σπουδαστή:\t");
    }
    while (((scanf("%d%c", &AM, &ent) != 2 || ent != '\n') && empty_stdin()) || AM
< 1 || AM > 100000);
```

```
do
{
    printf("\nΠαρακαλώ δώσε Εξάμηνο σπουδαστή:\t");
}
while (((scanf("%d%c", &exam, &ent) != 2 || ent != '\n') && empty_stdin()) ||
exam < 1 || exam > 50);

// Query για την εμφάνιση πλήθους εγγραφών της database που
δημιουργήθηκε
strcpy(ch, "SELECT COUNT(*) FROM Questions");
if (mysql_query(con, ch))
{
    printf("\nΣφάλμα κατά την προσπέλαση ερωτήσεων - Ελέγξτε αν
υπάρχουν εγγραφές\n\n");
    mysql_close(con);
    exit(1);
}

// Αποθήκευση του αποτελέσματος του query σε structure τύπου
MYSQL_RES
MYSQL_RES *result = mysql_store_result(con);

// Απομόνωση των αποτελεσμάτων
MYSQL_ROW row = mysql_fetch_row(result);
mysql_free_result(result);
rows = atoi(row[0]);

for (i = 1; i <= rows; i++)
{

    snprintf(ch, sizeof("SELECT * FROM Questions WHERE Id = "),
"SELECT * FROM Questions WHERE Id = %d", i);

    if (mysql_query(con, ch))
```



```
{
    printf("\nΣφάλμα κατά την επιλογή ερώτησης\n\n");
    mysql_close(con);
    exit(1);
}

// Αποθήκευση του αποτελέσματος του query σε structure τύπου
MYSQL_RES
MYSQL_RES *result = mysql_store_result(con);

// Απομόνωση των αποτελεσμάτων
MYSQL_ROW row = mysql_fetch_row(result);
mysql_free_result(result);

printf("\n\nΕρώτηση %s: %s\n\nA: %s\nB: %s\nC: %s\nD:
%s\n\n",row[0], row[1], row[2], row[3], row[4], row[5]);
printf("\n\nΑπάντηση:\t");

for (;;)
{
    if (((scanf("%c%c", &apant, &ent) != 2 || ent != '\n') &&
empty_stdin()) || apant<'A' || apant>'D')
        printf("Πιθανές απαντήσεις: A, B, C, D -
Ξαναπροσπάθησε: ");
    else
        break;
}

if (apant == row[6][0])
{
    printf("\nΣωστή απάντηση!\n");
    swstes++;
}
else
```

```
        {
            printf("\nΛάθος απάντηση...\n");
            printf("Σωστή απάντηση: %c", row[6][0]);
            printf("\nΑπόδειξη: %s\n\n", row[7]);
        }
    }

    printf("\n -- Σωστές απαντήσεις: %d / %d --\n\n", swstes, rows);

    snprintf(ch, 14+sizeof("INSERT INTO
Students(AM,Examino,SwstesAp,Pote) VALUES(",",","),NOW()+INTERVAL 1
    HOUR"),
        "INSERT INTO Students(AM,Examino,SwstesAp,Pote)
VALUES('%d','%d','%d',NOW()+INTERVAL 1 HOUR)", AM, exam, swstes);    //
    Prosbesi entolwn sto query

    if (mysql_query(con, ch)) // Εισαγωγή AM, εξαμήνου και πλήθους σωστών
    απαντήσεων στον πίνακα Students (Νέο record)
    {
        printf("\nΣφάλμα κατά την εισαγωγή record στον πίνακα Students - ο
    κάθε ");

        printf("φοιτητής έχει δικαίωμα να συμπληρώνει μία φορά το
    ερωτηματολόγιο.\n");

        // Σε περίπτωση που ο ίδιος φοιτητής επιθυμεί να ξανακάνει το test,
    δεν θα μπορέσει
        mysql_close(con);
        exit(1);
    }

    mysql_close(con);
}
```

// Σε περίπτωση λανθασμένης εισαγωγής απ'τον χρήστη, πρέπει να "αδειάσει"

το

```
// buffer του stdin για να δεχτεί νέα απάντηση.  
// Χωρίς τη χρήση της συνάρτησης, το πρόγραμμα στην επόμενη επανάληψη θα  
// δεχόταν ως όρισμα τη συνέχεια του string που δόθηκε στην προηγούμενη.
```

```
int empty_stdin()  
{  
    while (getchar()!='\n');  
    return 1;  
}
```

Οδηγίες εγκατάστασης

Όπως ήδη αναφέρθηκε, η εφαρμογή λειτουργεί σε συστήματα Unix, τα οποία έχουν προεγκατεστημένη τη γλώσσα προγραμματισμού C.

Για την απροβλημάτιστη λειτουργία της πρέπει να εγκατασταθεί στο σύστημα και η MySQL για την διαχείριση της βάσης δεδομένων.

Για να γίνει αυτό, πρέπει να εκτελεστεί η ακόλουθη εντολή σε ένα τερματικό (terminal prompt):

```
$ sudo apt-get install mysql-server
```

Σημείωση: Από τα Ubuntu 12.04 και μετά, η MySQL 5.5 εγκαθίσταται αυτόματα. Αν και είναι πλήρως συμβατή με την MySQL 5.1, θα χρειαστεί να εγκατασταθεί και αυτή, οπότε θα χρειαστεί και η εντολή:

```
$ sudo apt-get install mysql-server-5.1
```

Κατά την διαδικασία της εγκατάστασης ο χρήστης θα κληθεί να εισάγει ένα κωδικό (password) ως MySQL root χρήστης.

Με την ολοκλήρωση της εγκατάστασης, ο MySQL server θα πρέπει να αρχίσει την λειτουργία του αυτόματα. Η επόμενη εντολή μπορεί να εκτελεστεί για να φανεί αν ο server λειτουργεί:

```
$ sudo netstat -tap | grep mysql
```

Όταν εκτελεστεί η εντολή αυτή, πρέπει να εμφανιστεί η επόμενη γραμμή ή κάτι

παρόμοιο:

```
tcp    0    0 localhost:mysql    *.*          LISTEN      2556/mysql
```

Αν ο MySQL server δεν λειτουργεί σωστά, χρειάζεται η ακόλουθη εντολή για να ξεκινήσει χειροκίνητα:

```
$ sudo service mysql restart
```

Επίσης, για την ομαλότερη λειτουργία του λογισμικού, πρέπει να εγκατασταθούν και οι MySQL σχετικές με την C βιβλιοθήκες. Αυτό γίνεται μέσω της επόμενης εντολής:

```
$ sudo apt-get install libmysqlclient-dev
```

Οδηγίες εκτέλεσης

Για την εκτέλεση του προγράμματος, θα πρέπει να γίνει πρώτα compiling των αρχείων .c που υπάρχουν, δηλαδή η μετατροπή του C κώδικα σε μορφή εκτελέσιμου αρχείου.

Έτσι, στο τερματικό, αφού ο χρήστης βρίσκεται (μέσω της εντολής cd) στο ίδιο directory με το εκάστοτε αρχείο.c, εκτελείται την ακόλουθη εντολή:

```
$ gcc αρχείο.c `mysql_config --cflags --libs`
```

(Όπου "αρχείο.c" το όνομα του αρχείου)

Με την εντολή αυτή, δημιουργήθηκε ένα εκτελέσιμο, το οποίο για να "τρέξει", χρειάζεται την εντολή:

```
$ ./a.out
```

Αν χρειάζεται να υπάρχουν περισσότερα εκτελέσιμα ταυτόχρονα, θα δημιουργηθούν με ξεχωριστό όνομα το καθένα, δηλαδή με την εντολή:

```
$ gcc αρχείο1.c -o αρχείο1 `mysql_config --cflags --libs`
```

και

```
$ gcc αρχείο2.c -o αρχείο2 `mysql_config --cflags --libs`
```

Έτσι, τώρα υπάρχουν τα αρχείο1 και αρχείο2, τα οποία με τη σειρά τους μπορούν να εκτελεστούν με τις εντολές:

Ολοκληρωμένο διαδικτυακό σύστημα αξιολόγησης σπουδαστών

\$./αρχείο1 όρισμα (Αν χρειάζεται όρισμα για να εκτελεστεί)

\$./αρχείο2

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] <http://nemertes.lis.upatras.gr/jspui/bitstream/10889/5495/1/Diplwmatiki.pdf>
- [2] <http://vivliothmyy.ee.auth.gr/846/>
- [3] <http://www.mysql.com/information/benchmarks.html>
- [4] <http://dev.mysql.com/>
- [5] <http://www.cprogramming.com/>
- [6] <http://www.cplusplus.com/>
- [7] Αλεξανδρόπουλος Αν., Κατωπόδης Επ., Παλιάτσος Αθ., Πρεζεράκος Ν.
(1994) *Στατιστική*, Αθήνα: Σύγχρονη Εκδοτική.
- [8] <http://ebooks.edu.gr/modules/ebook/show.php/DSGL-C100/493/3203,13012/>
- [9] Murray R. Spiegel, John Schiller, R. Alu Srinivasan (2000) *Schaum's Probability and Statistics*, Second Edition, McGraw-Hill International Editions
- [10] George W. Snedecor, William G. Cochran (1967) *Statistical Methods*, Sixth Edition, The Iowa State University Press
- [11] <http://cgi.di.uoa.gr/~geotz/bib/papers/LeoTzo02-bsc-Ompi.pdf>

