

# Υλοποίηση Συστημάτων Βάσεων Δεδομένων (Κ18)

Φροντιστήριο Πρώτης Εργασίας

Νίκος Περδικοπάνης

Νοέμβριος 2020

# Παράδοση και εξέταση εργασίας

Καλείστε να παραδώσετε κώδικα που εκτελείται σε linux του di  
Η εξέταση της εργασίας θα γίνει στους εν λόγω σταθμούς εργασίας  
**(linux 16.04 , gcc 5.4.0.)**

# Χρήσιμες πληροφορίες

Υποδομή Linux στο di

Απαιτούμενα προγράμματα

Απομακρυσμένη σύνδεση (SSH)

Απαιτούμενα για σύνδεση όταν

λειτουργικό σύστημα Windows

λειτουργικό σύστημα Linux

Μεταφορά αρχείων με χρήση filezilla

Μεταφορά αρχείων με χρήση command line

Από τοπικο υπολογιστή προς απομακρυσμένο (linux@di)

Απο linux@di προς τοπικό υπολογιστή

Κατάλογος παράδοσης εργασίας

Δημιουργία ubuntu Virtual machine (προαιρετικό)

# Σύνδεση με Linux@di

**Υποδομή Linux στο di**

Στο Di υπάρχουν 30 σταθμοι εργασίας με ονόματα linux01.di.uoa.gr έως linux30.di.uoa.gr

Απαιτούμενα προγράμματα

Απομακρυσμένη σύνδεση (SSH)

Απαιτούμενα για σύνδεση από

λειτουργικό σύστημα Windows

λειτουργικό σύστημα Linux

# Σύνδεση με Linux@di

Υποδομή Linux στο di

## **Απαιτούμενα προγράμματα**

Χρειάζεται να διαθέτετε ssh για την εκτέλεση εντολών και sftp για τη μεταφορά των αρχείων σας. Επιτρέπεται σύνδεση μόνο μέσω ssh protocol.

Απαιτούμενα για σύνδεση από

λειτουργικό σύστημα Windows

λειτουργικό σύστημα Linux

# Σύνδεση με Linux@di

Υποδομή Linux στο di

Απαιτούμενα προγράμματα

**Απαιτούμενα για σύνδεση**

**από λειτουργικό σύστημα Windows**

Χρειάζεται να χρησιμοποιήσετε ένα ssh client console app και ένα sftp client. Τα πιο δημοφιλή είναι

PuTTY (<https://www.putty.org/>) + Filezilla (<https://filezilla-project.org/>)

Bitvise (<https://www.bitvise.com/ssh-client-download>) ssh+sftp

**από λειτουργικό σύστημα Linux**

Απο linux αν θέλετε μπορείτε να χρησιμοποιήσετε τον Filezilla για sftp.

Για ssh δίνεται την εντολή

ssh [userid@linuXX.di.uoa.gr](mailto:userid@linuXX.di.uoa.gr)

e.g.

# Bitwise

pot 22

```
nikosp@linux01.di.uoa.gr:22 - Bitwise xterm
Last login: Fri Nov 13 11:02:33 2020 from 109.242.38.189
13/11/20:nikosp@linux01:~[1]:
```

nikosp@linux01.di.uoa.gr:22 - Bitwise SFTP

Window Local Remote Upload queue Download queue Log

Browse Upload queue Download queue Log

Local files Filter: C:\Users\perdi\Desktop

Name	Size	Type
test	0	File folder

Remote files Filter: /home/users/nikosp/temp/k

Name	Size	Type	Date Modified
------	------	------	---------------

Upload status: 1 item transferred

Local selection: 1 folder

nikosp@linux01.di.uoa.gr:22 - Bitwise SSH Client

Default profile

Save profile as

Bitwise SSH Server Control Panel

New terminal console

New SFTP window

New Remote Desktop

Login Options Terminal RDP SFTP Services C2S S2C SSH Notes About

Server

Host linux01.di.uoa.gr

Port 22 ☐ Enable obfuscation

Obfuscation keyword

Kerberos

SPN

☐ GSS/Kerberos key exchange

☐ Request delegation

☒ gssapi-keyex authentication

Authentication

Username nikosp

Initial method password

☒ Store encrypted password in profile

Password

☒ Enable password over kbd fallback

Elevation Default

Proxy settings Host key manager Client key manager Help

11:01:52.401 Attempting password authentication.

11:01:52.472 Authentication completed.

11:01:52.862 Host key has been saved to the global database. Algorithm: ECDSA/nistp256, size: 256 bits, SHA-256 fingerprint: VR+qLLYBiBuebyhVNTlojJeeFmPbxzU6ATudGHAmjO0.

11:01:52.887 Host key has been saved to the global database. Algorithm: Ed25519, size: 255 bits, SHA-256 fingerprint: ezR2bhvYDk2yHaeBv98tw3nn8hPOy6MsiJDaPPBwqpk.

11:01:52.888 Host key synchronization completed with 2 keys saved to global settings. Number of keys received: 4, rejected: 1.

11:02:09.306 Terminal channel opened.

11:02:14.146 Terminal channel closed by client.

11:02:16.425 SFTP channel opened.

11:02:30.615 SFTP channel closed by user.

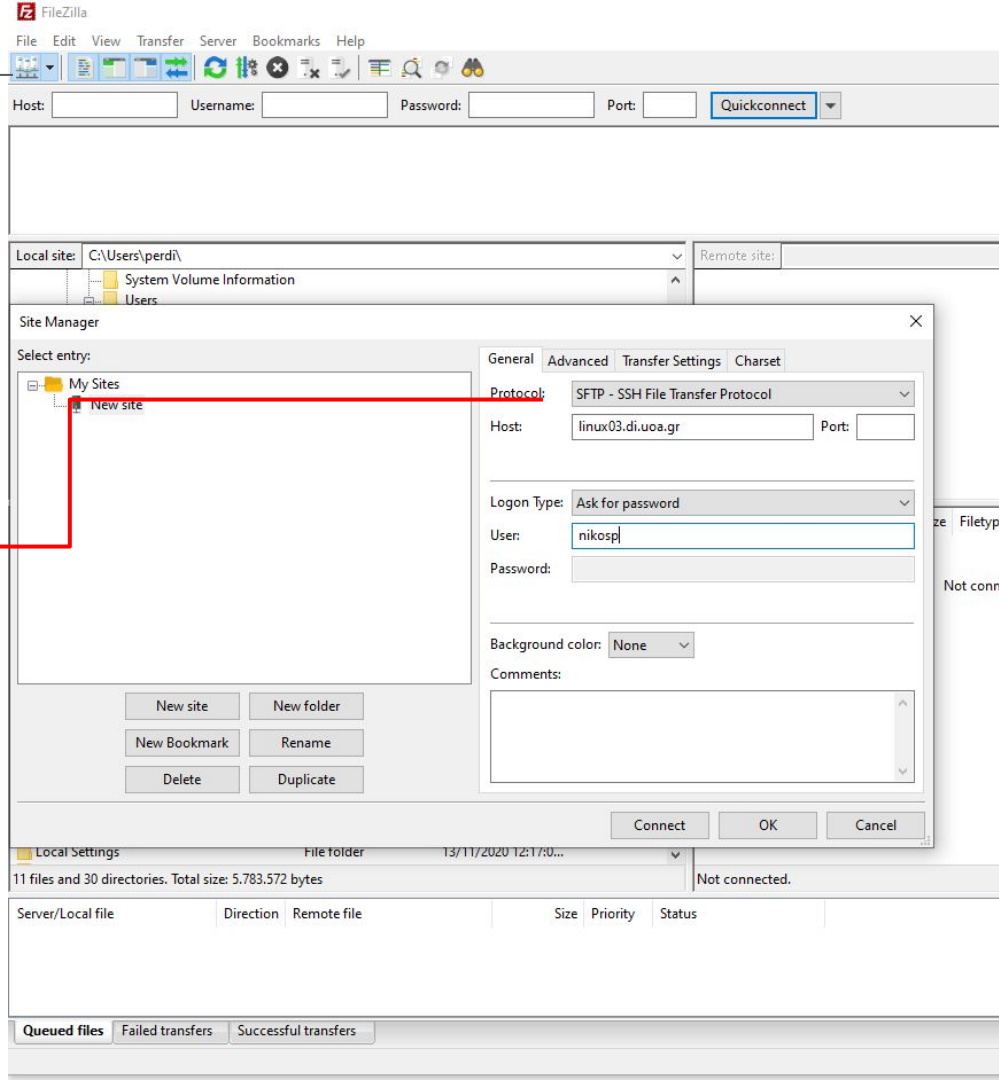
11:02:31.714 Terminal channel opened.

Log out Exit

# Filezilla

open the site manager

select SFTP





# Βιβλιοθήκη BF

Προσομοιώνει το επίπεδο διαχείρισης block. Αποτελεί το interface που θα χρησιμοποιήσετε για να υλοποιήσετε την άσκηση σας.

Κατ αρχήν πρέπει να μεταφέρετε τον κατάλογο που σας δίνεται στον χώρο των linux (όλα τα linux 01-30 μοιράζονται το ίδιο data storage με αποτέλεσμα τα δεδομένα σας να είναι ανεξάρτητα του σταθμού σύνδεσης)

Μεταφέρετε τα data

# Έλεγχος 32 or 64 bit λειτουργικό

Όπως θα δείτε στις βιβλιοθήκες που έχουμε στείλει υπάρχουν εκδόσεις 32 και 64bit

Στο ubuntu βρίσκεται την έκδοσή χρησιμοποιώντας το command

```
uname -a
```

```
linux01 4.15.0-123-generic #126~16.04.1-Ubuntu SMP Wed Oct 21 13:48:05 UTC 2020  
x86_64 x86_64 x86_64 GNU/Linux
```

Οπότε θα πρέπει να χρησιμοποιήσετε τις 64 bit βιβλιοθήκες

# Μεταφορά αρχείων με command line 1/2

Κατεβάζεται τα αρχεία `BF_lib.zip` και `record_examples.tar.gz`

Μπορείτε να τα αποσυμπιέσετε εκτελώντας

```
unzip BF_lib.zip -d BF_lib
```

**Μέσα στο *BF* δημιουργούνται τα *BF.h* και *main\_example\_BF.c* αλλά και τα**

***BF\_32.a.tar.gz*, *BF\_64.a.tar.gz* οποτε πάλι *unzip***

```
tar -xvf BF_lib/BF_32.a.tar.gz -C BF_lib/
```

```
tar -xvf BF_lib/BF_64.a.tar.gz -C BF_lib/
```

**Τέλος αποσυμπιέζω και τα παραδείγματα με τις εγγραφές**

```
tar -xvf record_examples.tar.gz
```

# Μεταφορά αρχείων με command line 2/2

**Για ένα μόνο αρχείο.**

scp file.txt userid@[linuxXX.di.uoa.gr:remote/directory/newFileName](#)

e.g. scp records10K.txt nikosp@linux01.di.uoa.gr:remoteTest/

**ή για ένα κατάλογο**

scp -r /local/directory [userid@linuxXX.di.uoa.gr:/remote/directory](#)

e.g. scp -r BF\_lib/ nikosp@linux01.di.uoa.gr:remoteTest/

# Μεταφορά αρχείων με χρήση command line

Μπορώ αν θέλω να μεταφέρω και αντίστροφα αλλάζοντας τη σειρά των ορισμάτων

Απο linux@di προς τοπικό υπολογιστή

```
scp userid@linuxXX.di.uoa.gr/remote/directory/newFileName file.txt
```

# Δοκιμή βιβλιοθήκης BF

Στον κατάλογο **BF\_lib** (εκεί που αποσυμπιέστε το αρχείο **BF\_lib.zip**) και έχετε μεταφέρει και αποσυμπιέσει στο linux υπάρχουν τα αρχεία:

BF\_32.a.tar.gz

BF\_64.a.tar.gz

BF\_32.a

BF\_64.a

BF.h

main\_example\_BF.c

# .a files

Τα αρχεία .a αποτελούν αρχεία βιβλιοθηκών. Συνδέονται στατικά κατά το στάδιο της μεταγλώττισης.

Δηλαδή αν γίνει μεταγλώττιση και αλλάξει κάτι στο .a πρέπει να ξαναγίνει μεταγλώττιση. (τα .a αρχεία δεν αλλάζουν στην εργασία). Τα .a αρχεία που παρέχονται περιέχουν τις βιβλιοθήκες BF για αρχιτεκτονικές 32 και 64 bit.

Δεν απαιτείται κατά το run time (όπως συμβαίνει με τις δυναμικά συνδεδεμένες βιβλιοθήκες .so)

# Text editor στο linux (nano)

`nano main_example_BF.c`

Βασικές εντολές

`ctrl+X` `exit` (και αντίστοιχα `Y` αν έχει αλλάξει κάτι για να σώσουμε τις αλλαγές)

`ctrl+X ->Y` `save`



## main\_example\_BF.c

```
#define FILENAME "nikos"
#define MAX_FILES 10
#define MAX_BLOCKS 500
int main(int argc, char** argv) {
    int bfs [MAX_FILES];
    int i, j;
    char filename [5];
    void* block;
    int blkCnt;
    BF_Init(); //πραγματοποιείται η αρχικοποίηση του επιπέδου BF.
    strcpy(filename, FILENAME);
    for (i = 0; i < MAX_FILES; i++) { // Δημιουργία MAX_FILES αρχείων
        printf("The FileName is %s\n", filename);
        if (BF_CreateFile(filename) < 0) { // Δημιουργεί Αρχείο το οποίο αποτελείται από block
            BF_PrintError("Error creating file");//stderr περιγραφή του πιο πρόσφατου σφάλματος
            exit(EXIT_FAILURE);
        }
        if ((bfs[i] = BF_OpenFile(filename)) < 0) {//ανοίγει ένα υπάρχον αρχείο από block filename
            BF_PrintError("Error opening file");
            break;
        }
    }
```

## main\_example\_BF.c

<pre>for (j = 0; j &lt; <b>MAX_BLOCKS</b>; j++) {     printf("Block %d\n", j);     if (<b>BF_AllocateBlock(bfs[i])</b> &lt; 0) {<b>//δεσμεύεται νεο block στο τέλος του bfs[i] και</b> <b>αρχικοποιείται με μηδενικά και δεσμεύεται πάντα στο τέλος του αρχείου</b>         BF_PrintError("Error allocating block");         break;     }     blkCnt = <b>BF_GetBlockCounter(bfs[i])</b>; <b>//βρίσκει τον αριθμό των διαθέσιμων block του bfs[i]</b>     printf("File %d has %d blocks\n", bfs[i], blkCnt);     if (<b>BF_ReadBlock(bfs[i], j, &amp;block)</b> &lt; 0) <b>{//βρίσκει το block με αριθμό j του bfs[i]</b>         BF_PrintError("Error getting block");         break;     }     strncpy(block, (char*)&amp;j, sizeof(int));     if (<b>BF_WriteBlock(bfs[i], j)</b> &lt; 0)<b>{//γράφει στο δίσκο το δεσμευμένο block με j του αρχείου</b> <b>bfs[i]</b>         BF_PrintError("Error writing block back");         break;     } } filename[0]++; <b>//change fileName</b> if (<b>BF_CloseFile(bfs[i])</b> &lt; 0) <b>{//κλείνει το ανοιχτό αρχείο bfs[i]</b>     BF_PrintError("Error closing file");     break; }</pre>	<pre>} } return 0; }</pre>
--	--

# Μεταγλώττιση και εκτέλεση της test main

## Μεταγλώττιση

*cd BF\_lib*

*gcc -o testMain main\_example\_BF.c BF\_64.a -no-pie*

**Και εκτελώ το παραγόμενο εκτελέσιμο καλώντας**

`./testMain`

# Αρχεία Δεδομένων

```
typedef struct{  
    int id,  
    char name[15],  
    char surname[25],  
    char address[50];  
} Record;
```

Δίνονται στο `record_examples.tar.gz` εγγραφές ελεγχου

πχ

```
{0,"name_0","surname_0","address_0"}
```

```
{1,"name_1","surname_1","address_1"}
```

Δίνονται αρχεία με 1κ 5κ 10κ και 15κ εγγραφές

# Σωρός Heap HP

```
int HP_CreateFile(char *fileName, /* όνομα αρχείου */, char attrType, /* τύπος πεδίου-κλειδιού: 'c', 'i' */,  
char* attrName, /* όνομα πεδίου-κλειδιού */, int attrLength /* μήκος πεδίου-κλειδιού */ )
```

```
HT_info* HP_OpenFile( char *fileName /* όνομα αρχείου */ )
```

```
int HP_CloseFile( HP_info* header_info )
```

```
int HP_InsertEntry(HT_info header_info, /* επικεφαλίδα του αρχείου*/, Record record /* δομή που  
προσδιορίζει την εγγραφή */ )
```

```
int HP_DeleteEntry(HT_info header_info, /* επικεφαλίδα του αρχείου*/, void *value /* τιμή του πεδίου-  
κλειδιού προς διαγραφή */)
```

```
int HP_GetAllEntries( HT_info header_info, /* επικεφαλίδα του αρχείου */ , void *value /* τιμή του  
πεδίου-κλειδιού προς αναζήτηση */)
```

# Hash Table HT

```
int HT_CreateIndex(, char *fileName, /* όνομα αρχείου */, char attrType, /* τύπος πεδίου-κλειδιού: 'c', 'i' */ , char* attrName, /* όνομα πεδίου-κλειδιού */,int attrLength, /* μήκος πεδίου-κλειδιού */, int buckets /* αριθμός κάδων κατακερματισμού*/)
```

```
HT_info* HT_OpenIndex( char *fileName /* όνομα αρχείου */ )
```

```
int HT_CloseIndex( HT_info* header_info )
```

```
int HT_InsertEntry(HT_info header_info, /* επικεφαλίδα του αρχείου*/ , Record record /* δομή που προσδιορίζει την εγγραφή */ )
```

```
int HT_DeleteEntry(HT_info header_info, /* επικεφαλίδα του αρχείου*/ , void *value /* τιμή του πεδίου-κλειδιού προς διαγραφή */)
```

```
int HT_GetAllEntries( HT_info header_info, /* επικεφαλίδα του αρχείου */, void *value /* τιμή του πεδίου-κλειδιού προς αναζήτηση */)
```

```
int HashStatistics( char* filename /* όνομα του αρχείου που ενδιαφέρει */ )
```

# FAQ

## Ποιο χαρακτηριστικό χρησιμοποιώ για το hash:

Χρησιμοποιώ το id

## Ποια hash function χρησιμοποιώ:

Υπάρχουν πολύ ωραίες Hash . Μπορείτε να χρησιμοποιήσετε όποια θέλετε.

(google is your friend) eg

## Τι είναι τα .a αρχεία

Είναι μεταγλωτισμένα αρχεία βιβλιοθήκης που πρέπει να γίνουν link κατά τη μεταγλώττιση του προγράμματος

## Τι έλεγχοι θα γίνουν κατά την εξέταση:

Θα χρησιμοποιηθούν τα αρχεία που δίνονται. **Ενδεικτικά και όχι μόνο.** θα ελεγχθεί η ορθή εισαγωγή, και διαγραφή εγγραφών, (έλεγχοι αν υπάρχει ήδη η εγγραφή στο αρχείο , αν μετά τη διαγραφή πράγματι έχει διαγραφεί το αρχείο κτλ.) Προσοχή θα ελεγχθούν και test τι γίνεται αν οι εγγραφές δεν χωρούν σε block

## Γιατί μόνο σε linux φέτος.

Λογω της ιδιαιτερότητας της κατάστασης

# Δημιουργία ubuntu Virtual machine (προεραϊτικο)

Μεταφορτώστε και εγκαταστήστε στο windows pc σας το VirtualBox από

<https://www.virtualbox.org/>

(χρειάζεται την windows host έκδοση **Downloads-->VirtualBox XXXX platform packages Windows hosts**)

(Αν χρησιμοποιείτε 32bit win τότε μεταφορτώστε την έκδοση 5.2 και όχι κάποια νεότερη)

<https://download.virtualbox.org/virtualbox/6.1.16/VirtualBox-6.1.16-140961-Win.exe>

Μεταφορτώστε την εικόνα του ubuntu 16.04 από:

<https://www.linuxvmimages.com/images/ubuntu-1604/#ubuntu-160407-lts>

Username: ubuntu

Password : ubuntu

**unzip to .ova file → run virtualBox-->File-->insert Device**