Operating Systems 1 - Laboratory Task 1 90 minutes
POSIX program execution environment

Vote Database Manager

Write a simple program called **vdbm** which is designed to manage list of votes in a file called votes database (*.vdb) from now on. A database element consists of vote textual option description and a number representing number of votes submitted for this option.

The database file consists of constant length text lines, one for each vote option. Each line has 64 characters. First 32 bytes store the option description as an ASCII text. For options shorter than 32 chars space characters have to be appended to fill the remaining bytes. The rest (excluding newline char) contains number of votes in textual format.

The program depending on command line arguments used performs different operations described below.

Invocation I: `./vdbm create file.vdb "Option 1" "Option 2" ...`

Creates a new database file with name given as a second argument (file.vdb) and populates it with initial vote options given as arguments from the third up. In case file.vdb exists program displays error and does nothing else. It should be possible to give no vote options at all - in that case program creates an empty database.

Invocation II: `./vdbm show file.vdb`

Displays contents of file.vdb in following format (precisely!):

```
Database filename: file.vdb
Option 1                        31
Option 2                        0
Option 3                        12
Database is locked/unlocked
```

Invocation III: `./vdbm vote file.vdb "Option X"`

Increments by 1 number of votes for option with description `"Option X"`. In case such option doesn't exist and the database is unlocked program creates it at the end of file. Otherwise it does nothing.

Invocation IV: `./vdbm vote -r file.vdb`

Decrements by 1 number of votes for option with description `"Option X"`. In case such option doesn't exist program does nothing. Number of votes can't drop below 0.

Invocation V: `./vdbm create -l file.vdb "Option 1" "Option 2" ...`

Creates new locked database i.e. one that cannot have new options appended by calling `./vdbm vote file.vdb "Non-existent option"`. The command above works exactly the same way as Invocation I and later appends a special line filled with 'X' characters at the end of the database file. This line should later be recognized by other operations and change their behaviour. If this special line is present `vdbm show` displays `Database is locked` text and `vdbm vote` forbids appending new vote options.

Please, note that vote options cannot have more than 32 characters. Errors should be displayed in case of longer arguments given.

STAGES (TOTAL OF 14 POINTS)

Stage	Points	Requirements
1	4	Creating new database (invocation I) works as described Verify by inspecting file size and viewing file in editor.
2	3	Showing database contents (invocation II) works as described Format should be exactly the same as presented in the task description.
3	3	Incrementing/decrementing votes (invocations III and IV) work as described. Non-existent options are appended. Modifications should be visible when using `vdbm show`
4	4	Creating locked databases (invocation V) works as described. `vdbm show` should display Database is locked message. `vdbm vote` should forbid appending new options.

UPLOAD

Please upload your solution to: **`/home2/samba/sobotkap/unix/`**

You have to upload each stage immediately after finishing it. You upload to a network share via ssh.mini.pw.edu.pl server:

```
scp user.etapX.tar.bz2 user@ssh.mini.pw.edu.pl:/home2/samba/sobotkap/unix/
```

Please name your stages files according to the schema: `LOGIN.etapN.tar.bz2(.gz)`

THE STATEMENT

By decree 27/2020 of University Rector you must add the following statement to the uploads:

```
------------------------------------------------------------------------
I declare that this piece of work which is the basis for recognition of
achieving learning outcomes in the OPS2 course was completed on my own.
[First and last name] [Student record book number (Student ID number)]
------------------------------------------------------------------------
```

Please add it as comment at the beginning of each source file you upload.
Replace square brackets with your data.