CST 412

Assignment 3

Programming problem. *For this problem, you can choose any programming language.*

Problem description:

- In this assignment, you will implement a program that manages and enforces an access control policy represented by an access control matrix (ACM) that we discussed in class. You will implement the ACM by row **using the CAPABILITIES approach.**

Provided input file:

- `input-acm-entries.txt`: This text file contains the entries of the following ACM:

	file1	**file2**	**file3**	**file4**	**file5**	**file6**	**file7**
userA	rw	orw	r		rw	r	r
userB		rw				orw	
userC	orw			orw			rw
userD	r		orw			rw	
userE	rw	r	rw		orw	r	orw

  Access rights explanation:
  - `r`: read.
  - `w`: write.
  - `o`: own.

  Each line in the file has the form `<sub_id>,<obj_id>,<a_right>`, where `<sub_id>` is the subject's id (`userA`, `userB`, …), `<obj_id>` is the object's id (`file1`, `file2`, …), and `<a_right>` is the access right (`r`, `w`, `o`) that the user has toward the object. Example: `userD,file1,r`

- `sample-update-acm-entries.txt`: This file contains the updates to the ACM. An update can be adding a new entry to the ACM or removing an entry from the ACM. Each line in this file has the form `<update_action>,<sub_id>,<obj_id>,<a_right>`, where `<update_action>` specifies the action of the update: `add` or `remove`, and `<sub_id>,<obj_id>,<a_right>` have the same meaning with the entries in the input ACM entries file described above.
  Example:
  - `add,userD,file2,r`
  - `remove,userC,file7,w`
  - `remove,userB,file7,w`

- `sample-requests.txt`: This file contains the access requests. Each line in this file is an access request. Each access request has the form `<sub_id>,<obj_id>,<a_right>`, which specifies that the subject with `<sub_id>` is trying to perform the access right `<a_right>` on the object with `<obj_id>`.
  Example:
  - `userE,file5,r`
  - `userA,file3,w`

Requirements:

- The program should provide data structures to represent an ACM using the Capabilities approach that we discussed in class.
- The program should provide a feature to print out the ACM. You can choose to print the ACM in any format
- The program should provide a feature to add new ACM entries and to remove existing ACM entries. The program needs to use this feature to add the input ACM entries from the input file `input-acm-entries.txt` and to update the ACM entries from an input file containing ACM entry updates such as `sample-update-acm-entries.txt`.
- The program should provide a feature to check if a request is **permitted** or **denied**. The program needs to use this feature to evaluate requests specified in an input file such as `sample-requests.txt`.

Grading Procedure:

Your program should display the menu with the following options:

1. Load input entries.
   - The program should ask for the file path.
2. Print ACM.
3. Update ACM entries from a file.
   - The program should ask for the file path.
4. Evaluate access requests from a file.
   - The program should ask for the file path.

Following is the grading procedure for this assignment:

- Choosing option 1 to load the input ACM entries from file `input-acm-entries.txt`.
- Choosing option 2 to print the ACM.
- Choosing option 4 to evaluate access requests from a test file. This file will follow the format of the file `sample-requests.txt`.
  Expected output: your program should print out the access decision for each request. Access decision can be "`Permit`/`Deny`", "`True`/`False`", or "`Yes`/`No`". For example, based on the input ACM, the access decision of the sample access requests mentioned above are:
  - `userE,file5,r:` **PERMIT**
  - `userA,file3,w:` **DENY**

- Choosing option 3 to process ACM entry updates from a test file. This file will follow the format of the file `sample-update-acm-entries.txt`. Your program needs to process the updates and print out a message indicating if the updates are successfully updated. Note that a removing update could be invalid if the entry does not exist in the current ACM. In this case, your program should print an error message. For example, based on the input ACM, the access decision of the sample access requests mentioned above are:
    - `add,userD,file2,r:` **`Successfully Updated.`**
    - `remove,userC,file7,w:` **`Successfully Updated.`**
    - `remove,userB,file7,w:` **`Invalid Update. Entry Not Found.`**
- Choosing option 2 to print the updated ACM.
- Choosing option 4 to evaluate access requests from another test file with the updated ACM.

Submission Instructions:

- **Please submit a zip file** containing your program and a document with instructions on how to run the program.

Grading Rubric:

Task	**Point**
Reading input ACM entries.	20
Updating ACM entries.	20
Evaluating access requests.	20
Generate correct outputs.	30
Program compiled and ran successfully and well-commented code	10
Total	100