Advanced Operating System (2021 - 2022)

Assignment 1

It is needed to be familiar with the ***Deadlock*** problem and how to solve it. Therefore, it is required to apply the deadlock avoidance algorithms by applying the *Banker's algorithm*. You will write a Java code that implements the banker's algorithm: processes request and release resources and the banker will grant a request only if it leaves the system in a safe state. A request is denied if it leaves the system in an unsafe state.

The bank will employ the strategy outlined in the textbook & lectures whereby it will consider requests from n processes for m resources. The bank will keep track of the resources using the following data structures:

```
int [] available;          //the available amount of each resource

int [][] maximum;          //the maximum demand of each process

int [][] allocation;       //the amount currently allocated to each process
int [][] need;             //the remaining needs of each process
```

You should build a test program to test your code. The test program should take the initial number of the available resources at the bank, the maximum need, and the actually allocated resources for each process. Your program should calculate the need matrix as well as the new available vector.

<u>Output:</u>

The output screen must appear every action happened as well as the current state for the bank by showing the values for your data structures as well as the process requests and releasing for the resources. The bank decisions to either deny or approve the requests must be shown through the output screen.

After the testing the user can type:

- RQ <process#> <r1> <r2> <r3>: It means that process request resources. So add this request to the needed resources for the given process and check again if the system is in a safe state.

- RL <process#> <r1> <r2> <r3>: It means that process release resources. So check if release resources are less than or equal to allocated resources and if so, then subtract this release resources from allocated resources for a given process.

- Recover: if a request let to a **Deadlock**, apply recovery algorithm (choose a victim , force it to release resources and check again if the system is in a safe state, if still in an unsafe state then will choose a victim again force it to release and check again and keep repeating until it reaches a safe state).

  Mention based on what did you choose your victim (i.e. priority, how much longer to complete, resources used, resources needed …etc.).

- Quit: it closes the application

Submission

1- This assignment will be submitted into groups of **(3 to 4 students)** and the due date is on **Thurs. 7 April 2022 11:59 pm on Blackboard**. You should work in groups from the same lab.

2- You can deliver code in **(Java, c++, c#, or python)**. You should deliver a running code and match the assignment criteria.

3- You should submit your code on blackboard before the specified time. Follow this steps in your submissions or you will lose your grades **(No Excuses)**:

   a. You should deliver a folder with your program files. Do not deliver any subfolders just one folder with sub-files.

   b. Name the folder with your IDs separated by underscores **without any spaces**.

   c. Compress this folder in a **.zip** file with the same folder name and upload it.

   d. Only **one member** from the group should upload the assignment file.

Grading Criteria:

Banker Algorithm	2 marks
Recovery	1.5 marks
Request	0.5 marks
Release	0.5 marks
Quit	0.5 marks

Good Luck