Assignment 9

Operating System Lab (**CS342**)

Department of CSE, IIT Patna

Date:- 12-Mar-2019 **Time:-** 3 hours

Instructions:

1. All the assignments should be completed and uploaded by 05:00 pm, Marks will be deducted for late submission.
2. Markings will be based on the correctness and soundness of the outputs. Marks will be deducted in case of plagiarism.
3. Proper indentation and appropriate comments are mandatory.
4. You should zip all the required files and name the zip file as ***roll_no*.zip**, eg. **1501cs11.zip.**
5. Upload your assignment (**the zip file**) in the following link: https://www.dropbox.com/request/yzWz4v77qDGNLOzKQ7PZ

Q1. In this assignment you will test the efficacy of various virtual memory paging schemes. You will implement paging scheme: **Optimal**. Your program should read the sequence of page accesses from the file **"pages.txt"**, emulate the paging schemes, and then print a short report detailing the performance statistics.

Input Format

The file is comprised of a sequence of "tries". Each try consists of a line containing a single integer, *f*, representing the number of frames of memory in the operating system, and then, in subsequent lines, any number of integers, 0-99, representing page numbers. The sequence of integers is a "reference string"--a sequence of page requests. The reference string shall be terminated by a value of -1 (which does not represent a page number.)

Sample Input:

```
3
1 2 3 4 3 4 2 3 5 6 4 2 1 2 -1
4
1 2 3 4 2 7 5 1 1 6 4 7 2 1 2 5 -1
```

Output

For each "try" your program should print a table detailing the page replacements affected by the paging methods. Be sure that you properly handle the printing of these tables when the length of the reference string is large enough that the table is wider than the page. After printing the paging tables, your program should print a table detailing the number of page faults affected by the paging schemes. Below is the output that should result from the sample input.

Optimal:

1 2 3 4 3 4 2 3 5 6 4 2 1 2

---

1 1 1 4	4 4	1
0 2 2 2	2 2	2
0 0 3 3	5 6	6

Q2.

The banker's algorithm is a resource allocation and deadlock avoidance algorithm that tests for safety by simulating the allocation for predetermined maximum possible amounts of all resources, then makes an "s-state" check to test for possible activities, before deciding whether allocation should be allowed to continue.

Consider the following snapshot of a system:

Allocation Matrix

	A	B	C	D
P0	0	0	1	2
P1	1	0	0	0
P2	1	3	5	4
P3	0	6	3	2
P4	0	0	1	4

Max

	A	B	C	D
P0	0	0	1	2

P1	1	7	5	0
P2	2	3	5	6
P3	0	6	5	2
P4	0	6	5	6

Available

A	B	C	D
1	5	2	0

Implement Banker's algorithm(Safety Algorithm + Resource Allocation Algorithm) and answer the following questions using the banker's algorithm.

a. What is the content of the matrix Need?

b. Is the system in a safe state?

c. If a request from process P1 arrives for (0,4,2,0), can the request be granted immediately?