Government Engineering College, Thrissur

CS331 – System Software Lab

Documentation -

Exp 4 – Banker's Algorithm

Date of Submission
14 September 2020

Submitted By
Kowsik Nandagopan D
Roll No 31
TCR18CS031
GECT CSE S5

Experiment 4

Implement the banker’s algorithm for deadlock avoidance

Compilation of Code

Prerequisite

- The code is provided in the **program.c** along with this documentation. You can open the terminal in Linux (Ubuntu 18.04 tested). Then run the command

  gcc program.c

  ./a.out

- We can execute the code in console and see the output as soon as we press enter key. There are **five input files**

  1. *nums.txt:* This text file is to provide the number of processes, number of resources and process id for which we would make request to test the safe sequence exists or not if that process asks for a that amount of resources. **If Process ID number is -1 then request won’t be evaluated.**

     <Process Number> <Tab> <Request Number> <Tab> <Process ID Number>

  2. *req.txt:* Text file to store the requested amount of resources.

     <Number> <Tab> … (Based on number of resources)

  3. *max.txt:* Maximum resources that can be allocated. Numbers are tab separated and the each line is new process. **Last line should not be empty.**

     <Number> <Tab> … (Based on number of resources)

  4. *alloc.txt:* Currently allocated resources. Numbers are tab separated and the each line is new process. **Last line should not be empty.**

     <Number> <Tab> … (Based on number of resources)

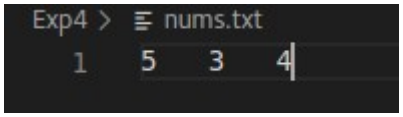
  5. *avail.txt:* Available resources that can be allocated. Numbers are tab separated. This file contain only one line. (No blank lines should be there.

     <Number> <Tab> … (Based on number of resources)

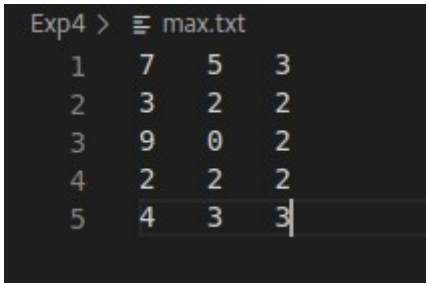
- Output of the code will be printed on the **console**
- **Note: Please see the my_machine_output.txt file for the output I got on my machine.**

Output / Screenshots

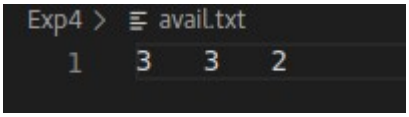
nums.txt



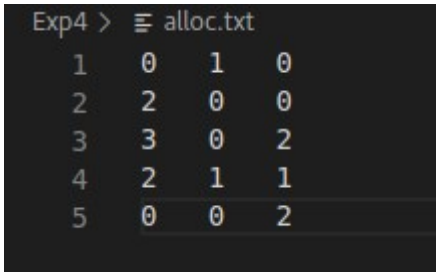
max.txt



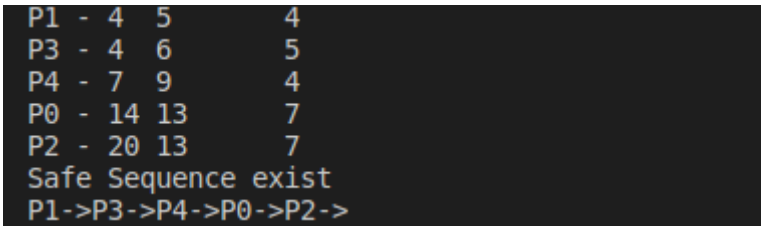
avail.txt



alloc.txt



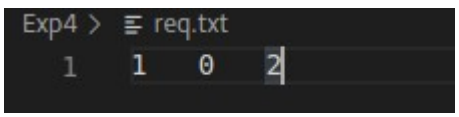
Output when no special request is made



Output based on requested

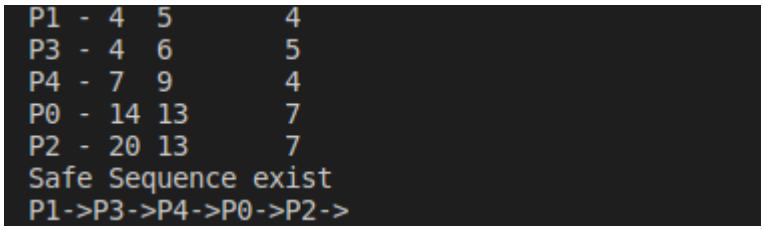
Safe sequence

req.txt



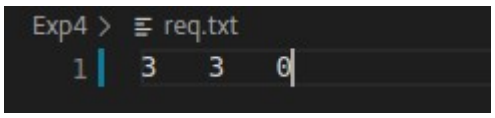
Output

3



Unsafe Sequence

req.txt



Output