CSCI 403/ECE 408

Program 3

Tentatively due April 28th

In this program you will implement a simulation of processes and resource assignment, requests and decisions of allocating requests based on deadlock detection. In case of "many resource request denials" you will be asked to implement process relinquish resources

You are to read file **data.txt** (note your code should be flexible enough to handle any file data.txt that fits the required format)

First line of file will be

N;a_1,a_2,…, a_w

This means there are **N** processes and **w** resources. There are a_1 many units of resource 1, a_2 many units of resource 2, …a_w many resources of resource w

Example

5; 2,5,7,1,4,9

There are 5 processes and 6 resources. Two units of R1, five units of R2, seven units of R3, one unit of R4, four units of R5 and nine units of R6

The next N lines of the file data.txt are the **initial allocation** for each of the N processes. You DO NOT NEED to check these allocations they WILL NOT cause a deadlock, also the allocations will not change unless

Example

1,0,0,1,2,0
0,1,2,0,0,2
0,2,1,0,0,2
1, 1,2,0,0,1
0,0,0,0,0,2

Once you have read the total number of recourses and then the allocation then you need to calculate **available**

Example

Available=0, 1,2,0, 2,2

The program will then read in N lines at a time: each of the lines (1 through N) represents a resource request. The i^{th} line is the resource request for process i

You are to run the deadlock detection to see if there exists a safe state. If a safe state exists **output yes and output the safe state (example: 3,4,1,2,5) , otherwise output no**

You do not modify allocation nor available

However whenever there are **three consecutive outputs of No to resource requests** we do the following: one at a time **starting at i=1 we relinquish the allocation of resources assigned to process i to available**. After each relinquish we run the deadlock detection for this third request. If the output is yes, we go back to reading file data.txt (N lines at a time) if the output is No we relinquish resources of process i+1 (stopping at N). Note: if we relinquish all N processes the program terminates

The program terminates once you have completely read in the entire file data.txt