Banker’s Algorithm

Operating System Project

Banker’s algorithm in operating system?

- 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.

Why banker’s algorithm is named so?

- Banker’s algorithm is named so because it is used in banking system to check whether loan can be sanctioned to a person or not. Suppose there are n number of account holders in a bank and the total sum of their money is S. If a person applies for a loan then the bank first subtracts the loan amount from the total money that bank has and if the remaining amount is greater than S then only the loan is sanctioned. It is done because if all the account holders comes to withdraw their money then the bank can easily do it.
  In other words, the bank would never allocate its money in such a way that it can no longer satisfy the needs of all its customers. The bank would try to be in safe state always.

Cont.

Following **data structures** are used to implement the banker's algorithm:
let **'n'** Be the number of processes in the system and **'m'** be the number of resources types.

Available :

- It is a 1-d array of size **'m'** indicating the number of available resources of each type.
- Available[j] = k means there are **'k'** instances of resource type $\mathbf{r}_j$

Cont.

Max :

- It is a 2-d array of size ‘**n*m’** that defines the maximum demand of each process in a system.
- Max[i, j] = k means process $\mathbf{p}_i$ may request at most **‘k’** instances of resource type $\mathbf{r}_{j.}$

Cont.

Allocation :

- It is a 2-d array of size **‘n*m’** that defines the number of resources of each type currently allocated to each process.
- Allocation[i, j] = k means process $\mathbf{p_i}$ is currently allocated **‘k’** instances of resource type $\mathbf{r_j}$

Cont.

Need :

- It is a 2-d array of size **‘n*m’** that indicates the remaining resource need of each process.
- Need [i, j] = k means process $\mathbf{p_i}$ currently need **‘k’** instances of resource type $\mathbf{r_j}$
- Need [i, j] = max [i, j] – allocation [i, j]

Banker’s algorithms types

Banker’s algorithm consists of safety algorithm and resource request algorithm.

Safety algorithm
the algorithm for finding out whether or not a system is in a safe state can be.

Disadvantages of the banker algorithm

It requires the number of processes to be fixed; no additional processes can start while it is executing. It requires that the number of resources remain fixed; no resource may go down for any reason without the possibility of deadlock occurring.

What problems has banker's algorithm solved?

Deadlock avoidance is solved by banker's algorithm. Banker's algorithm pretends to have allocated to required resources to the processes, and if the system doesn't lead to deadlock, it actually allocates the resources.

Team members

- Nahed mohamed Mostafa
- Amira abdelmoean
- Mohamed samir
- Omar mohamed
- Mohamed abdelmoez

THANK YOU