

**Department Of Computer Engineering**

**Operating Systems Term Project**

Implementation Of Banker’s Algorithm

Submitted to: Submitted by:

Özgür Özdemir Kazım Tulgaroğlu

118 200 58

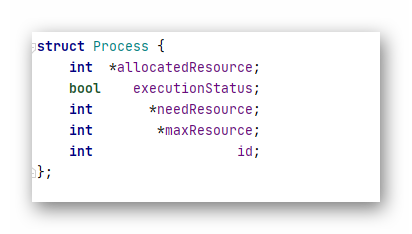
## What is Banker’s Algorithm?

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.

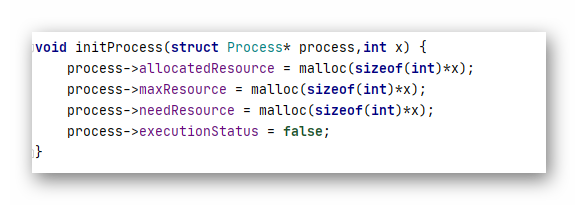
**Banker’s algorithm** is used majorly in the banking system to avoid deadlock. It helps you to identify whether a loan will be given or not.  Banker’s algorithm helps the operating system to successfully [share the resources](https://t4tutorials.com/monitors-examples-process-synchronization-role-procedures-shared-data-operating-systems-os/) among all the processes.

## Analysis Of Code

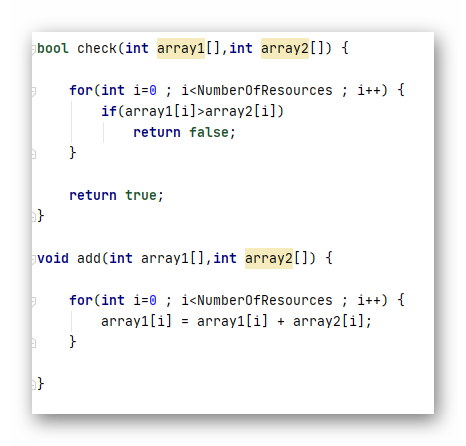
A struct which is called Process has been created to represent every process. The struct has 5 features. AllocatedResource specifies the number of resources for the that process. MaxResource specifies the max number of resources the process needs. NeedResource specifies the number of deficient resources. ExecutionStatus specifies that whether process is exucuted or not. Id is process’s id.



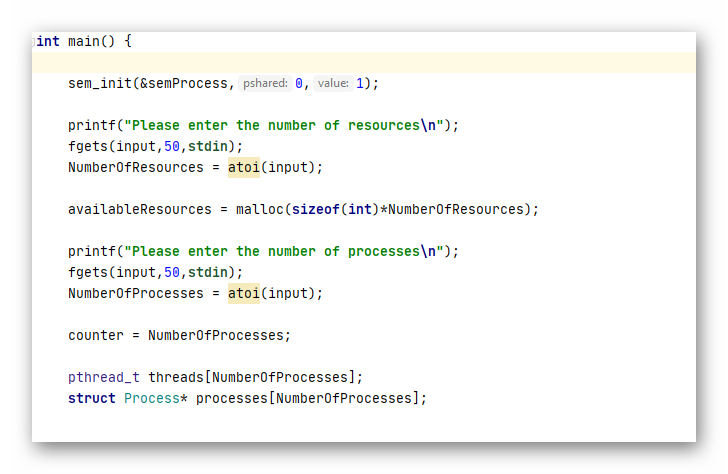
This function initializes the structs. It allocates the memory for arrays.

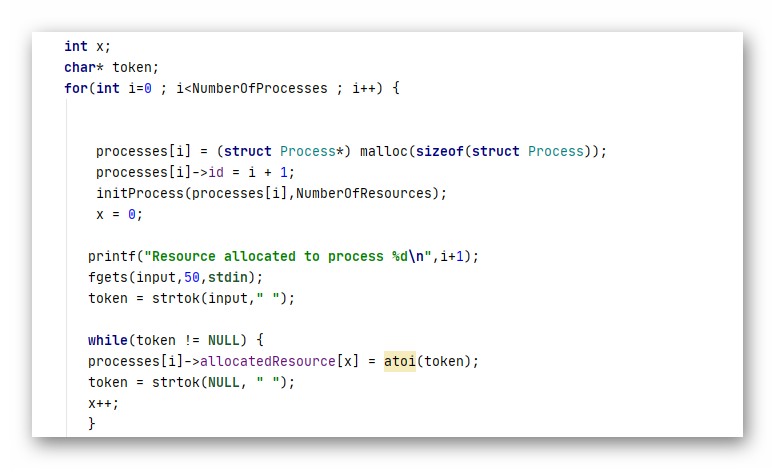


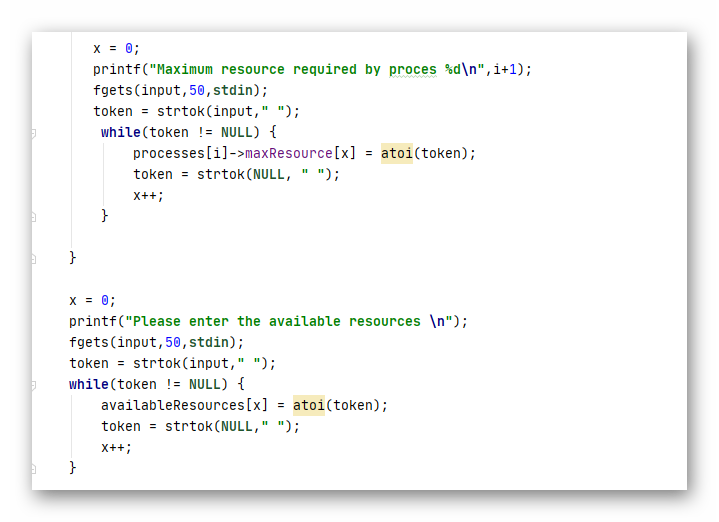
There are two helper functions. First one is check. It returns a boolean expression. It compares allocated resources to max resources that process needs. If there are enough resource for process to be executed then it returns true. Second one is add. After a process is executed its allocated resources are released. This method adds allocated resources into available resources.



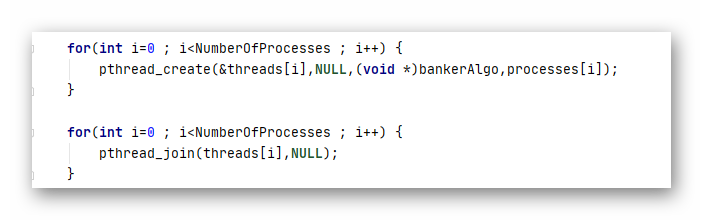
At the beginning of pragram some informations are asked to enter from user. These are number of resources and processes. After that numerical values of resources of procesess are asked to enter from user.



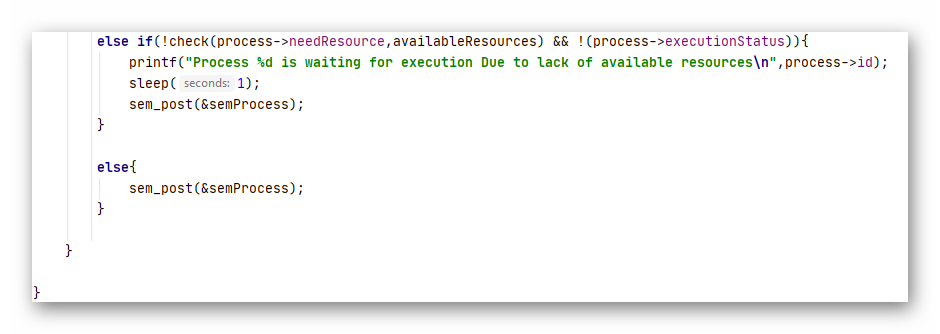




At the end of program threads are created and started.



Every thread performs the same function called bankerAlgo. At the beginning of function numbers of resources that processes need are calculated simultaneously. After this part of function critical section begins. Only one thread can access this part. This part is in endless loop. There is a variable called counter in here. At the beginning of program the variable is set to number of process. This variable used to keep track of processes that are executed. Whenever a process is executed counter is reduced one by one. When it reaches zero program terminates itself. 



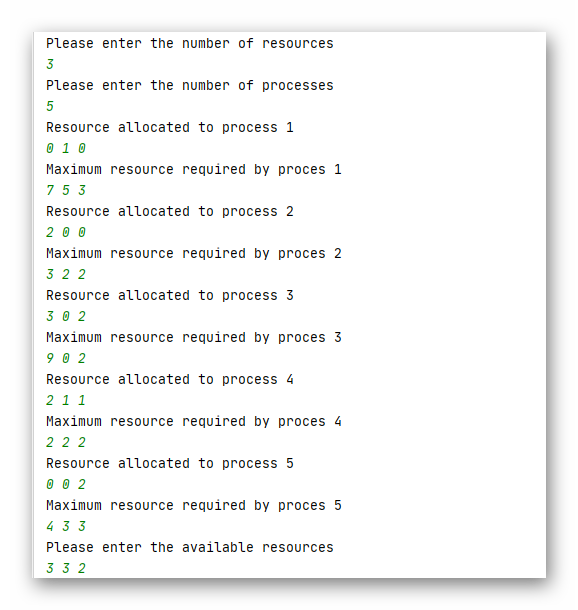
## Sample Use Of Program

|  |  |  |  |
| --- | --- | --- | --- |
|  | Allocated | Max | Available |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Processes | A | B | C | A | B | C | A | B | C |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Process 1 | 0 | 1 | 0 | 7 | 5 | 3 | 3 | 3 | 2 |
| Process 2 | 2 | 0 | 0 | 3 | 2 | 2 |  |  |  |
| Process 3 | 3 | 0 | 2 | 9 | 0 | 2 |  |  |  |
| Process 4 | 2 | 1 | 1 | 2 | 2 | 2 |  |  |  |
| Process 5 | 0 | 0 | 2 | 4 | 3 | 3 |  |  |  |

INPUTS



Output

