Project 3: Nucleus Message Passing

(Done By Anagha Fatale SUID:438039600 and Honey Shah SUID:767850372 and)

## Synopsis

The purpose of this project is to implement the Nucleus Message Passing.

## Motivation

The purpose of the system nucleus is to implement these fundamental concepts: simulation of processes; communication among processes.

## Implemented Logic

BufferEntry:

BufferEntry class is created which has attributes:

BufferId, Sender, Receiver, Result and Answer.

BufferPool:

Have implemented BufferPool which initializes the objects of buffer entries along with their bufferIds.

ProcessPool:

To keep the tract of all existing processes, Process Pool is created.

ProcessQueue:

Each Process has the associated ProcessQueue with them. This ProcessQueue will keep the track all bufferIds associated with the communication in which the process is involved.

Below SystemCalls are implemented for Nucleus Message Passing:

SendMessage Implemented:

Initially for the communication, the buffer id is kept as '-1' since process is not aware about the buffer available.

Fetch the attributes like receiver name, message from the parameters passed to the system calls and finds first available buffer within the pool and stores the values in that buffer.

Then this buffer is passed for further communication and is also inserted into the queue of receiver so that receiver is aware of buffer used for further communication.

WaitAnswer Implemented: The process waits until answer from receiver arrives in its queue.

Here if the message is arrived into the queue then it checks whether the message is from receiver process or it is dummy message from system on behalf of receiver.

WaitMessage Implemented: The process waits until message arrives in its queue.

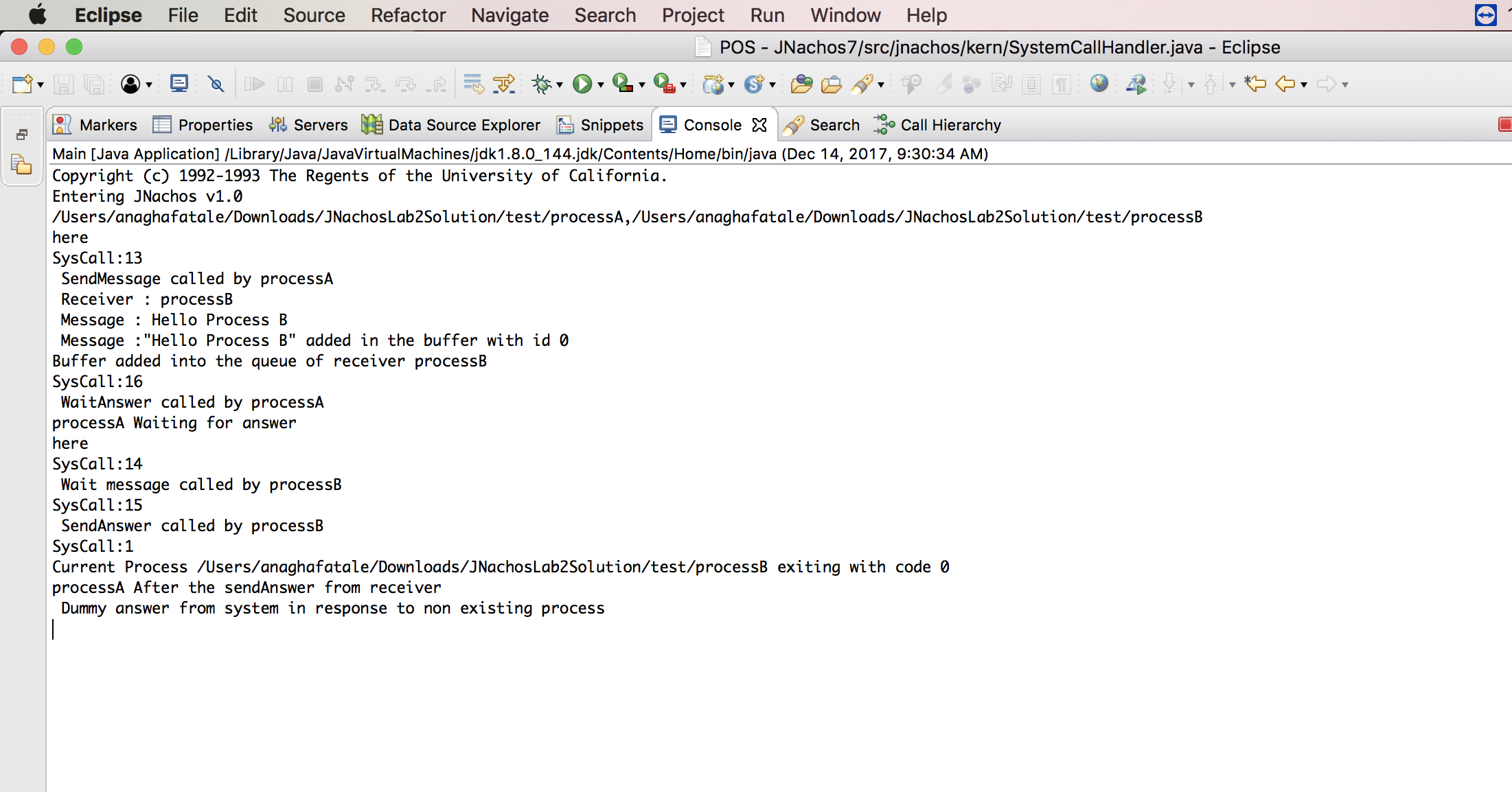
Here it also checks whether there is any message for it. Initially the BufferId is '-1'. It traverse its queue to find non negative bufferId which tells that some process has send the message.

The updated bufferId is retrived and send as return value to the systemcall.

SendAnswer Implemented: Here the process receives the buffer in its queue. Hence the process receives the message from sender. The process now updates its answer to sender by updating the same buffer and puts in the queue of the sender. This buffer is then removed from receiver’ queue.

# Test scenario for Implementation :

processA initiates communication and where processB exists before WaitAnswer is called again by processA.



## Code Sample:

Command line arguments for Process A and Process B:

-x /Users/honey/JNachosLab2Solution/test/processA,/Users/honey/JNachosLab2Solution/test/processB

## API Reference

Depending on the size of the project, if it is small and simple enough the reference docs can be added to the README. For medium size to larger projects it is important to at least provide a link to where the API reference docs live.

## Contributors

Original Developers of JNachos Operating System and Nucleus message passing implementation by Anagha Fatale and Honey Shah.