Porject 2 group 8, yangyang xie

1.requirements.(system call)

Implement the message passing between user processes with some additional functionality.

Following system calls need to be implemented.

* + **TopicLoopup:** that will allow a process to discover what interest groups are there.
  + **TopicCreate:** that will allow a process to create an interest group.
  + **TopicPublisher:** that will allow a process to declare itself a publisher of a specific interest group.
  + **TopicSubscriber:** that will allow a process to declare itself a subscriber to an interest group.
  + **Publish:** that will allow a publisher to send a message to an interest group.
  + **Retrieve:** that will allow a subscriber to retrieve one message from an interest group.
* While a publisher is sending to a topic, no other user can publish to or retrieve from the topic. (it means, when you sending a topic/writing file, other operations should be block)However, multiple subscribers can be retrieving messages from the same topic at the same time(reading the same file from different processes are allowed).
* You will maintain a buffer that can contain 5 messages for each topic. No publisher can send to the topic when the buffer is full(receiving file with 5 maximum processes simultaneously, mutex=5). You decide whether a subscriber should be blocked when Rretrieve finds an empty buffer(set a flag to block subscriber). A message is removed from the buffer after it has been retrived by all the subscribers.
* A message is published once and retrieved by all subscribers, who had subscribed to the topic before the message was published, at least once and at most once(one subscriber must subscribe a message once, no more than once?).
* We ask you to specifically address these questions: Can deadlock occur with your IPCs? If not, why? If yes, do you detect deadlock and how? If so, how you can recover from it?. If you do not detect deadlock, you should also say why. If deadlock can occur, can you advise the programmer to avoid it?