Project No. 2: Causal Ordering of Messages

DESCRIPTION

In this project, you will develop a communication system that maintains the causal ordering or messages. You will build the system on top of TCP and implement the Schiper-Eggli-Sandoz protocol. Specifically, you will implement the following functions.

- *void Causal_Init(char *filename)*. This function must be called at the beginning of a program that uses your communication system. After this function, each machine is associated with a *rank* that is used in the communication routines.

  This function loads the information about all the machines that are included in the communication system, initializes internal data structures, creates TCP sockets and establishes connections. The machine information is stored in file *filename*, whose format is as follows:

  ```
  <machine1's ip address port>
  <machine2's ip address port>
  ......
  <machineN's ip address port>
  ```

- *int Causal_Rank()* will return the rank of the process.
- *int Causal_Size()* will return the number of processes in the system.
- *void Causal_Write(void *data, ssize_t size, int dst_rank)*. This function sends *size* bytes *data* to the destination whose rank is *dst_rank*.
- *void Causal_Read(void *data, ssize_t size, int *src_rank)*. This function receives a message in causal ordering. The receiving buffer is of size *size* bytes. The function can receive messages whose size is less than *size*. If a larger message arrives, this function reports an error and exits. *src_rank* is a return value indicating from which process the message is received.

GRADING POLICY

1. Makefiles/README/Compiling errors, etc (20).
2. Causal_init, Causal_rank, Causal_size (20).
3. Causal_Read and Causal_Write without causal ordering (20).
4. Causal_Read and Causal_Write with causal ordering (40)

DEADLINES AND MATERIALS TO BE HANDED IN

There are two phases in this project. The deadline for phase one is **Feb. 19** on which date you must demonstrate your communication system without causal ordering (first 60 points). The final deadline is **March 5**. You must demonstrate the whole system and hand in a hard copy of the programs on March 5.

MISC.

You must design the demonstration. You must show at least two cases: a simple case where the causal ordering may be violated and a complex case that contains a lot of communications. IF YOU DO NOT SHOW IT IN THE DEMONSTRATION, THE GRADER WILL ASSUME OTHERWISE.