Design of Assignment 1

**//Introduction/Purpose**

The purpose of this project is to show and apply principles of Inter-Process Communication(IPC). This project has allowed us to understand how to use shared memory, message queues, signals, and working with both sender and receiver processes.

**//Project Overview**

This project uses IPC in order to synchronously transfer files between two process, a sender and receiver. The sender will send files to the receiver process and the receiver will receive files from the sender. The receiver will then send a message to the sender.

**//Design Constraints**

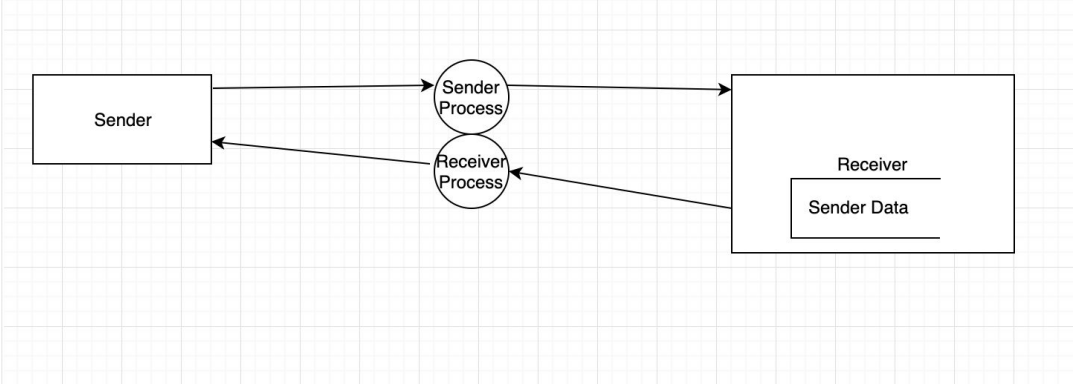
A constraint that we had was having to use and implement our code into the given skeleton code. Another constraint we had was implementing the code into the said diagrams, making sure all key components were correct.

**//Responsibilities**

| Name | Responsibilities |
|---|---|
| Samantha Yee | Design/Written Documentation |
| Kate Torres | Implementation |
| Gregory Vasquez | Design/ Written Documentation |
| Daniel Walsh | Implementation |

**//Data Flow Diagram**

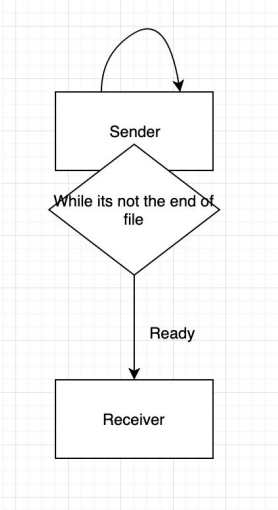
Entity- Sender, Receiver
Process- sender process, receiver process
data structure-sender data
data flow-sender sends using sender process into receiver, sender data is saved in receiver, receiver uses receiver process to let sender know message was received by sender.



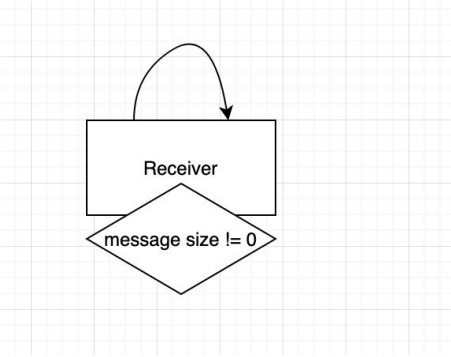
**//Structure chart**

The structure for the Sender to Receiver to Sender:

In the main loop for the Module Sender, the sender condition will send a message to the receiver signaling that the data is ready and will keep sending data until the end of the file is reached.




The Module Receiver is accessed by sender. The diamond shape is a condition that shows when the end of the file is reached, sends a message to the receiver, and signals to the receiver that sender will send no more. The curved arrow represents the loop where receiver will repeat until the whole file has been read through.



**//HIPO Diagram**