```
main:{
    creates the two threads(for read and write).
    while{
        if(flag == "input finished"){
            return
        }
    }
}

write_thread:{
    while{
        if(pointer_write < size_buffer){}
            if(there is input) => calls pipe_write;
            else => sends no input flag at read in order to
terminate;
            else => send termination flag;
        }
        else{
            if(pointer_read isn't at the start of the buffer){
                pointer_write = 0;
            }
        }
    }

}

pipe_write:{
    writes the data in the buffer;
    while{
        if(pointer_write == pointer_read -1){
            yield;
        }
        else{
            break;
        }
    }
    pointer_write++;
}

read_thread:{
    while{
        if(the pipe hasn't just start && pointer_read < size_buffer){
            check = (calls pipe_read);
        }
        else{
            if(pointer_write isn't at the start of the buffer){
                pointer_read = 0;
            }
        }
        if(check = 0 aka there is no data to read){
            send termination flag;
        }
    }

}

pipe_read:{
```

```
        checks if he has been send no input flag and if he has read all the
data
            from the buffer and then terminates read_thread;

        reads the data from the buffer;

        while{
            if(pointer_read == pointer_write - 1){
                yield;
            }
            else{
                break;
            }
        }
        pointer_read++;

        prints the data;
}
```