Advance Operating System

Assignment 3

By Qazi Danish Ayub

Msds20075

**Questions: For this assignment, make a small file (say 20bytes), memory-map it and then enter data into that file until the data reaches the file size limit, you will encounter an error then. Observe where this error occurs and what type of error is this. Reopen your file and then check either your data is still there or not?**

**Solution**

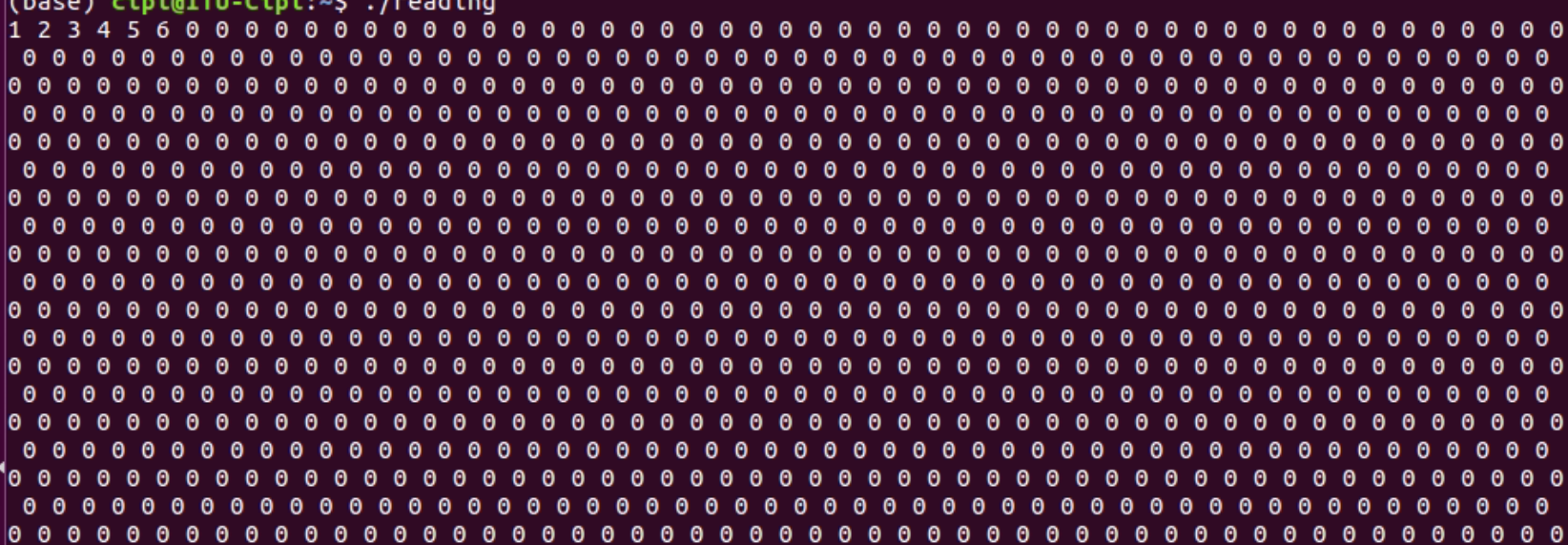
**Myfile.c**  
The code creates a file.txt of size 20 bytes. The file is created using the "w" mode which means that if the file already exists, its contents will be overwritten. The file pointer is then moved to the 20th byte using fseek and a null character is written using fputc. Finally, the file is closed using fclose.

**Mapping.c**

The code maps the file "file.txt" into memory, allowing it to be read and written to as if it were an array of integers. The for loop then writes the numbers 1 through 1000 to the file, one number per line. Finally, the munmap function unmaps the file from memory, and the close function closes the file descriptor.

**Reading.c**  
The code first opens the file "file.txt" for reading only. It then uses the mmap() function to map the file to memory. The map is 1000 bytes long, and the PROT\_READ flag means that the map is read-only. The MAP\_SHARED flag means that changes to the map will be visible to other processes that map the same file. The file descriptor and offset arguments are passed to mmap(). The code then loops through the map, printing out each integer. Finally, it unmaps the file from memory and closes the file.

file.txt (20bytes)



**Error.**

The error occurs when the data reaches the file size limit. This is a buffer overflow error. The data is still there when the file is reopened.

but looking in real time it shows that the while increasing the data rate the next fields were not getting what it was meant to be written in the data.