CS 307 – HW 4 – Memory Mapping & File Reading

Berna Yıldıran

1- fstream.cpp

In CPP file I use “fstream” to read the file and use “get” function inside a while loop to check every character of the file. Then I write an if statement inside the while loop which checks whether the character is equal to ‘a’. If it is equal than I increment the counter which counts the amount of ‘a’s in the file. I use time header inside my cpp file to calculate the execution time of the code. At the end, output of the code is there are 19082160 and this operation takes approximately 6 seconds. Fstream is easy to implement and safer but it takes more time for it to execute when compared to fopen and memory mapping. For large files using fstream is less efficient compared tother ones. Also, another reason for this method to be slower than others can be C++ is much slower than C while doing I/O operations and program checks every char with a while loop, it is better to use this option with smaller files.

2- fopen.c

In this C file, I use “fopen” to read the file and use “fgetc” function in a while loop to check each character in the file. If statement inside my loop checks that whether the character is equal to ‘a’. If it is equal than I increment the counter which counts the amount of ‘a’s in the file. I use time header inside my .c file to calculate the execution time of the code. At the end, output of the code is there are 19082160 and this operation takes approximately 4 seconds. This method is slightly faster than using fstream for reading a file, but it is not as fast as memory mapping. Also, it is 2 seconds faster than fstream but its also risky to implement when it compared to fstream. Since C is much faster than C++ in I/O operations and it is better to use this option with larger files.

3- mmap.c

In this C file, I use memory mapping. Memory mapping is generally considered the most efficient way to read files. It takes approximately 1 seconds for code to compile and return the output. Instead of reading blocks of data, you map the content of the file to a pointer and the operating system is responsible with filling in the data. It has the reputation to be amazingly fast because the data on disk can be mapped directly to memory without any undue copying. But this method is also less stable compared to other methods because with a small mistake while doing memory mapping and your program may crash.

It can be concluded that fstream is a better option for safely running code, but it takes more time to compile. On the other hand, fopen and memory mapping are better options for fast running codes. But when these tree option considered; memory mapping is the most time efficient method for the code.