Bilkent University

Department of Computer Engineering

CS 342 - Operating Systems
Spring 2024

Project 4

Bahadır Günenc - 22003672
Emir Tuğlu - 22003165

Introduction

In this report, the results of the experiments that are performed on the “fatmod” program that accesses and modifies FAT32 disk images will be discussed. Fatmod provides create, delete, read, and write functionality for files inside a FAT32 disk image. The program directly accesses the disk image without requiring to mount the file system. In the experiments, elapsed time while writing different amounts of data to a file and reading different sized files were examined. The elapsed time is the time between calling the program with the read/write command and completing printing/writing activity.

Experiment 1: Writing Different Amounts of Data

In this experiment, the effect of changing the write data size on the elapsed time is measured. The experiment was conducted on two different types of files. First, different sized data was written on empty files, and the elapsed times were recorded. In the second part, a file was populated with initial data whose size was larger than the maximum data size in the experiment (8000). In this way, we had a chance to observe the impact of allocating more clusters in the empty file. Recorded values are shown in the following table:

Data Size (bytes)	Time Elapsed in Empty File (ns)	Time Elapsed in Filled File (ns)
1000	76506490	40733122
2000	123429116	86675206
3000	252078524	151174524
4000	383650232	188755730
5000	433076269	147836060
6000	430417832	219273695
7000	529134265	221843477
8000	601455249	247137887

Table 1: Time elapsed for writing different sized data

The results demonstrate that the larger the data size is, the more time is required to perform the writing operation on the disk. In both parts of the experiment, the increase is mostly linear. Results also show that writing on an empty file is more time

consuming than writing on an already filled file. The reason behind this is that the fatmod program looks for an empty cluster to allocate whenever a new cluster is needed. In the empty file, every cluster is allocated in the runtime which increases the elapsed time. However, in the filled file, the clusters are already allocated, so the program only overwrites the existing data. This explains why the writing operation on the empty file has larger elapsed times than the wiring on the filled file.

Experiment 2: Reading Different Sized Files

In the second experiment, reading operations were performed on files with different sizes. The aim of this experiment is to monitor the effect of the size of the file on the time spent while the reading operation is performed.

File Size (bytes)	Time Elapsed (ns)
1000	520473
2000	459379
3000	1189541
4000	1465152
5000	1728866
6000	2375055
7000	2687141
8000	2855248

Table 2: Time elapsed for reading different sized files

Like the first experiment, the elapsed time increases linearly as the file size increases. This is an expected result since at the end of each cluster, the program finds out the next cluster number by performing some operations like accessing the FAT table, finding the next FAT table entry, accessing the next data cluster, etc. In the experiment, file sizes were chosen in the way that in each step forward, there is one more cluster needed since the cluster size is 1024 bytes. So, the results correlate with our expectations.

Conclusion

To conclude, we have developed a program called “fatmod” which performs read, write, create, and delete operations on FAT32 disk images. Using this program, we have performed two performance experiments and discussed their results and analysis. We have observed that the elapsed time increases linearly as the size of the data written to or read from the file increases.