Jason Ellis

Harry Hull

## CPSC 3385 File Structures and Multimedia

**Project 3**

In project 3 a B+Tree was implemented to test what the affect the block size has on its performance. All of the code was jointly written by both group members. Jason Ellis took the final output data and made graphs, while Harry Hull wrote the final project analysis.

The cluster size on the tested machine is 4 KB.

|  |  |  |  |
| --- | --- | --- | --- |
| Block Size | Fanout | Actual Height | Calculated Best Case Height (All nodes are full) |
| 64 | 4 | 11 | 9 |
| 512 | 38 | 4 | 4 |
| 1024 | 78 | 3 | 3 |
| 3072 | 235 | 3 | 3 |
| 4096 | 314 | 3 | 3 |
| 6144 | 472 | 3 | 2 |
| 8192 | 629 | 2 | 2 |
| 16384 | 1259 | 2 | 2 |

As the block size increases the total construction time increases (linearly), and total search time increases (logarithmically). However, the number of nodes read while searching decreases (logarithmically), the number of nodes written decreases (logarithmically), the height decreases (logarithmically), and the number of nodes read while inserting decreases (logarithmically).

All of the block sizes reached their optimal heights except for block sizes 64 and 6144, because of the split-down method of splitting nodes and the insertion order of the data.

The optimal block size for the B+Tree is 512 Bytes. The 512 Bytes block size's number of nodes read/written while searching/inserting and height are very close to the rest of the block sizes (except for 64 which it is significantly smaller) while the rest of the block sizes also have much longer construction and search times.

The design tradeoff for this project was between tree search/creation time and hard disk accesses. A block size of 512 seemed to be optimal for a cluster size of 4 KB. Future projects could test the effects of varying cluster sizes on the B+Tree's performance.