**REPORT**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Load Factor** | **Hash Function** | **Collision Handling** | **Collision Count** | **Indexing Time (ms)** | **Avg. Search Time (ns)** | **Min. Search Time (ns)** | **Max. Search Time (ns)** |
| α=50% | SSF | LP | 54673744 | 3234 | 46933 | 200 | 288600 |
| DH | 2841872 | 1055 | 8445 | 100 | 145200 |
| PAF | LP | 25480473 | 1874 | 2479 | 100 | 91900 |
| DH | 21369418 | 1999 | 2813 | 100 | 66200 |
| α=80% | SSF | LP | 76513376 | 3244 | 47308 | 200 | 307400 |
| DH | 3964474 | 1053 | 9209 | 200 | 156800 |
| PAF | LP | 38191984 | 2052 | 2575 | 100 | 83300 |
| DH | 30627536 | 1994 | 3118 | 200 | 74500 |

Performance Monitroing Table

A load factor of 0.5 collides less than a load factor of 0.8. Hash coding with SSF has more collisions than hash coding with PAF. Linear search does a lot more overlap than double hash. We can say that using 0.5 load factor, SSF, and double hashing on our data is the most appropriate way to avoid collisions. The indexing time with a load factor of 0.5 and the indexing time with a load factor of 0.8 is similar. Using PAF with the linear probe or double hashing for indexing times is similar. Using SSF with double hash is faster than these, but using SSF with linear search is slower than these. In our data SSF, DH is the fastest way to index all data. A load factor of 0.5 searches slightly faster than a load factor of 0.8. Double hash is better for search than linear search. PAF is much faster in search compared to SSF.