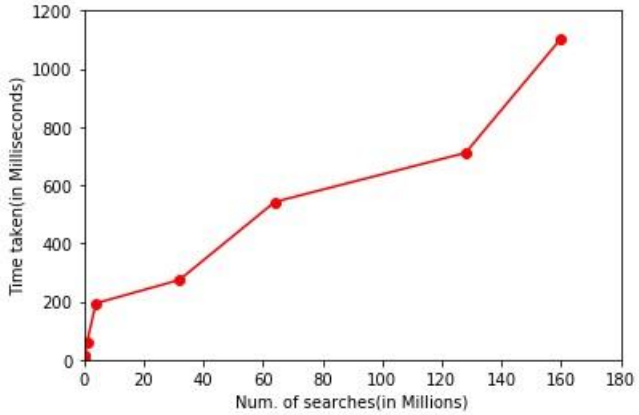
Harmonia: A High Throughput B+tree for GPUs

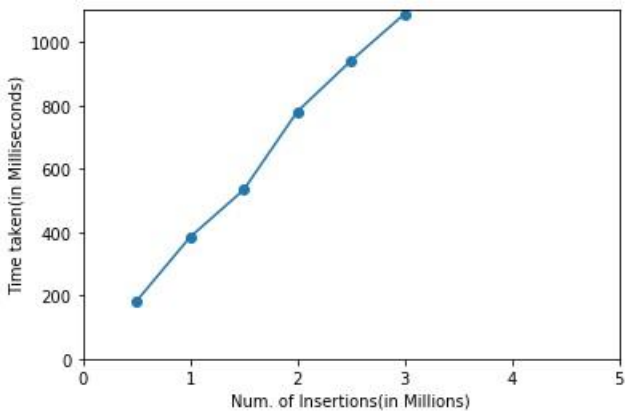
Jash Khatri

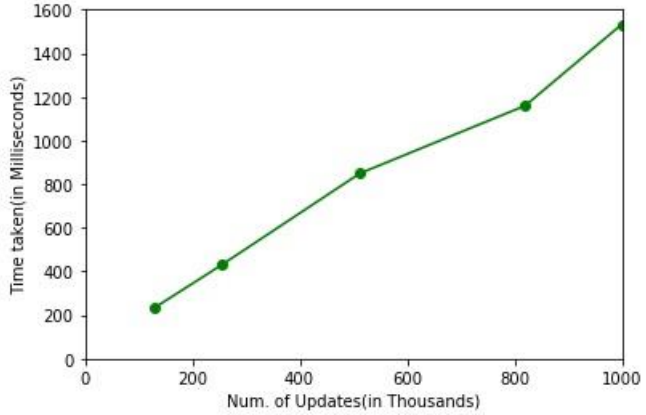
CS19S018

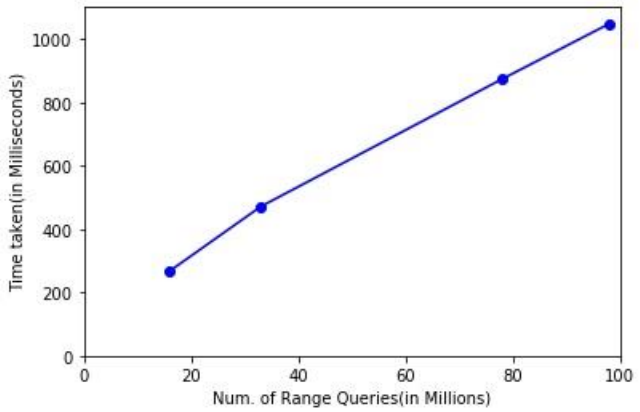
Problem Statement and Results

- **Problem Statement**: To perform Database operations like Search, Range Query, Update, Insert using a novel B+ tree structure called Harmonia to achieve high throughput for the given operations on GPU.
- **Results:**
- Throughput(Search) = 160 Million queries / sec (approximate)
- Throughput(Range Queries) = 100 Million queries / sec (approximate)
- Throughput(Insertion) = 3 Million queries /sec (approximate)
- Throughput(Updates) = 1 Million queries / sec (approximate)









Challenges Faced

- Implementing Search with multiple thread to perform comparison on a node at each level. There was issue with proper coordination among threads. Resolved using barriers at proper places.
- Issue in implementing radix sort for large arrays. Resolved using Batch-wise partial Radix sort.
- Issue in implementing Narrow Thread Grouping. Resolved by storing the number of currently stored keys in a given Node inside the structure of Harmonia. This will avoid useless comparisons inside the Node of Harmonia.
- Support for Record Pointers within the Harmonia structure. Resolved by adding tuples itself inside the structure of Harmonia.