

# CS280 – BTrees

## April 4, 2016

*<http://azrael.digipen.edu/~mmead/www/Courses/CS280/BTrees-1.html>*

### Homework

- All the nodes start at infinity
  - Constantly updating paths as you find cheaper ones
  - Have to clear previous path before adding a new one
- Use priority queue
  - Vector
  - Sort
- Edge class
- Node class

### B-Trees

- External storage
  - Not everything fits in memory
- Storing pointers to blocks of disk space instead of memory address
- More children per node, smaller the height
- Btrfs file system (B-Tree file system)
- Database programming
  - MySQL
- If you can even keep one node in memory that will give a performance increase since it is one less disk access

- Minimum degree is minimum # of children you can have
- Can adjust how full you want to keep your nodes on the tree
- Typically one node == one disk read
  - Depending on node size you may be able to store more than one in memory
- Key-range searches
  - Lets you avoid going up and down the tree in range searches
- Doing actual node manipulation overhead is almost un-noticable because it is CPU complexity, which is much less than disk access time
- Lazy delete just marks as deleted
- Keep data stored in separate structure and just have pointers to different sort order structures
  - One copy of data, just pointers to sort differently
- $\log_t(N)$
- Take database classes!
  - For every game programming job, 20 database jobs
  - Databases used for MMORPG's
  - Relational calculus
  - SQL – Structured Query Language