BTree

Goal: Build a tree that uses _________________ /node!  
...optimize the algorithm for your platform!

A BTree of order m is an m-way tree where:
1. All keys within a node are ordered.

BTree Insert, using m=5

...when a BTree node reaches m keys:

BTree Insert, m=3:

Great interactive visualization of BTrees:  
https://www.cs.usfca.edu/~galles/visualization/BTree.html

BTree Properties
For a BTree of order m:
1. All keys within a node are ordered.
2. All leaves contain no more than m-1 keys.
3. All internal nodes have exactly one more child than keys.
4. Root nodes can be a leaf or have [2, m] children.
5. All non-root, internal nodes have \([\text{ceil}(m/2), m]\) children.
6. All leaves are on the same level.

Example BTree

What properties do we know about this BTree?

BTree Search
```cpp
bool Btree<K, V>::_exists(BTreeNode & node, const K & key) {
    unsigned i;
    for (i=0; i < node.keys_ct_ && key < node.keys_[i]; i++) {}  
    if ( i < node.keys_ct_ && key == node.keys_[i] ) {  
        return true;
    }  
    if ( node.isLeaf() ) {  
        return false;
    } else { 
        BTreeNode nextChild = node._fetchChild(i);
        return _exists(nextChild, key);
    }
}
```

**BTree Analysis**
The height of the BTree determines the maximum number of ________ possible in search data.

...and the height of our structure:

**Therefore**, the number of seeks is no more than: ________.

...suppose we want to prove this!

---

**Proof:**

1a. The minimum number of nodes for a BTree of order m at each level is as follows:

    root:
    level 1:
    level 2:
    level 3:
    ... level h:

1b. The minimum total number of nodes is the sum of all levels:

---

2. The minimum number of keys:

---

3. Finally, we show an upper-bound on height:

---

**CS 225 – Things To Be Doing:**

1. Programming Exam B starts on Tuesday
2. MP4 is due tonight by 11:59pm; MP5 released Tuesday
3. lab_btree released on Wednesday
4. Daily POTDs are ongoing!