EXERCISE 4:
Consider the following (2, 4)-B-Tree.

![B-Tree Diagram]

1. Insert the elements 80, 75, 65 and 70.

2. Delete 10, 15 and 30.

Trickquestion Consider a B-tree consisting of a root and $\sqrt{n}$ children each containing $\sqrt{n}$ elements. A binary search for an element would then cost $O(\log(\sqrt{n}))$ (2 binary searches in $\sqrt{n}$ elements). From other classes you know that binary search is bounded by $\Omega(\log n)$! How is that possible?