Please work on these problems and be prepared to share your solutions with classmates in class. Assignments will **not** be collected for grading. Please read Chapter 2. of CGAA.

1. Let $S$ be a set of disjoint line segments whose upper endpoints lie on the line $y = 1$ and whose lower endpoints lie on the line $y = 0$. These segments partition the horizontal strip $[-\infty : \infty] \times [0 : 1]$ into $n + 1$ regions. Give an $O(n \log n)$ time algorithm to build a binary search tree on the segments in $S$ such that the region containing a query point can be determined in $O(\log n)$ time. Also describe the query algorithm in detail.

2. The intersection detection problem for a set $S$ of $n$ line segments is to determine whether there exists a pair of segments in $S$ that intersect. Give a plane sweep algorithm that solves the intersection detection problem in $O(n \log n)$ time.

3. The algorithm that I sketched in class and that is described in more detail in CGAA Chapter 2, may have an event queue that is large. Can you show an example where the event queue can grow so that its space complexity is in $\Omega(n^2)$.

4. There is a trick that can be used to guarantee that the event queue never grows beyond $O(n)$ space, without increasing the time complexity. What is the trick and why does it guarantee the reduced storage requirement. (HINT: Read the chapter carefully.)