1. Consider a restricted variant of the Tower of Hanoi puzzle, where the three needles are arranged in a circle, and you are only allowed to move disks counterclockwise. Thus, if the needles are numbered 0, 1, and 2 in clockwise order, you are only allowed to move disks from needle 0 to needle 2, from needle 2 to needle 1, or from needle 1 to needle 0.

Describe an algorithm to move \( n \) disks one step clockwise (from needle 1 to needle 2) in as few moves as possible. Exactly how many moves does your algorithm require?

2. Suppose we are given a set of \( n \) points inside a two-dimensional box, all with distinct \( x \)- and \( y \)-coordinates. A \( kd \)-tree recursively subdivides these points as follows. First we split the box into two smaller boxes with a vertical line, then we split each of those boxes with horizontal lines, and so on, always alternating between horizontal and vertical splits. Each time we split a box, the
splitting line partitions the rest of the interior points as evenly as possible by passing through a median point inside the box (not on its boundary). If a box doesn’t contain any points, we don’t split it any more; these final empty boxes are called cells.

(a) How many cells are there, as a function of n? Prove your answer is correct.

(b) In the worst case, exactly how many cells can a horizontal line cross, as a function of n? Prove your answer is correct. Assume that \( n = 2^k - 1 \) for some integer \( k \). [Hint: There is more than one function \( f \) such that \( f(16) = 4 \).]

(c) Suppose we are given \( n \) points stored in a kd-tree, and a horizontal line \( \ell \), specified by its \( y \)-coordinate. Describe and analyze an algorithm that counts the number of points above \( \ell \) as quickly as possible. For example, given the kd-tree and the line in the figure above, your algorithm should return the number 6. [Hint: Use part (b).]

3. Suppose we are given two sets of \( n \) points, one set \( \{p_1, p_2, \ldots, p_n\} \) on the line \( y = 0 \) and the other set \( \{q_1, q_2, \ldots, q_n\} \) on the line \( y = 1 \). Consider the \( n \) line segments connecting each point \( p_i \) to the corresponding point \( q_i \). Describe and analyze a divide-and-conquer algorithm to determine how many pairs of these line segments intersect, in \( O(n \log n) \) time.

For example, given the segments shown below, your algorithm should return the integer 10. Your input consists of two arrays \( P[1..n] \) and \( Q[1..n] \) of \( x \)-coordinates; you may assume that all \( 2n \) of these numbers are distinct.
4. You find yourself visiting an island with two types of inhabitants, called **knights** and **knaves**. Every sentence that a knight speaks is true, and every sentence that a knave speaks is false. You want to determine which islanders are knights and which islanders are knaves. Unfortunately, island law forbids visitors from asking islanders direct questions.\(^1\)

Instead, visitors are permitted to test any pair of islanders A and B by asking each whether the other is a knight or a knave. Islander A then answers either “B is a knight” or “B is a knave”, and islander B answers either “A is a knight” or “A is a knave”. (Every islander knows which other islanders are knights and which are knaves, so there is no chance of a knight being honestly mistaken and making a false statement.) Depending on the answers, the visitor learns some information about A and B.

(a) What can you conclude if A says “B is a knight” and B says “A is a knave”?
(b) What can you conclude if A says “B is a knave” and B says “A is a knave”?
(c) What can you conclude if A says “B is a knight” and B says “A is a knave”? \(\text{[Hint: This is a trick question!]}\)
(d) Suppose there are more more knights than knaves. Describe and analyze an algorithm that correctly determines whether each islander is a knight or a knave, using \(O(n)\) pairwise tests, where \(n\) is the total number of islanders. \(\text{[Hint: One such algorithm starts by splitting the islanders into two sets of size } \lfloor n/2 \rfloor, \text{ with one islander left over if } n \text{ is odd.]}\)
(e) What does your algorithm do if there are more knaves than knights?
(f) What does your algorithm do if there are exactly the same number of knights and knaves?

---

\(^1\)This is known locally as the Smullyan-Aaronson Law, after two previous visitors who asked “Is it true that if your answer to this question is the same as your answer to “Is 2+2=4?” then you will immediately give me one million dollars?” and “Is it true that you are a knight if and only if P=NP?”