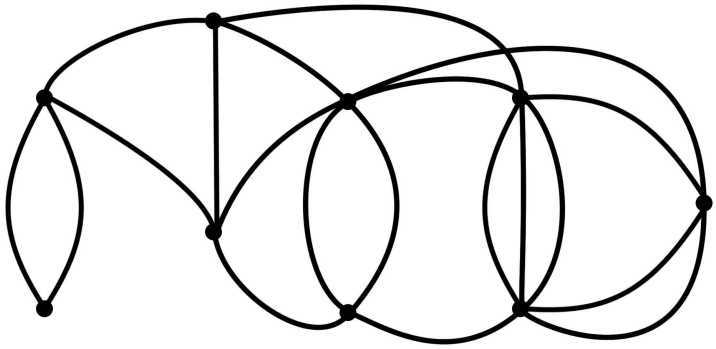
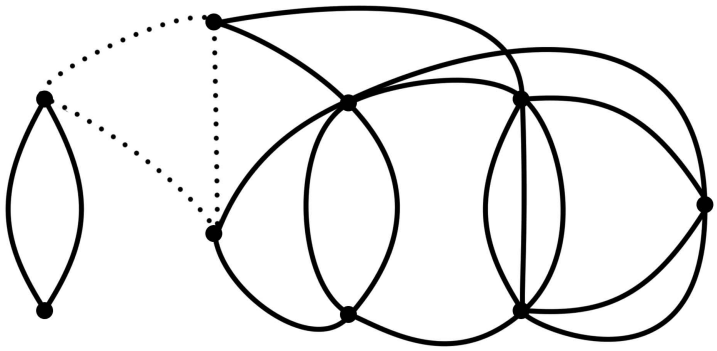
Handout: Induction

Suppose you had checked every possible layout of ropes up to 20 rope segments or less.

Let's convince ourselves that this configuration of 21 rope segments works must have a path that uses all the segment with one rope **without finding the actual path**.



1. What if you left off the bottom left loop? What would that mean for the rest? [Remember, we know we have already checked every example with less than 21 rope segments.]
2. How could the answer to (1) help?
3. Could you always find a loop to leave off, no matter what umbrella you start with?
4. Why is this possible?
5. What if that loop accidentally broke your original problem up; for instance, suppose you picked out the dotted loop here:



Could you still use this loop to help find a way to lay your ONE rope around the whole 21 segments?

6. Was there anything super special about the 21 rope segment case we just looked at?

   Could you use the same argument and the smaller problems to guarantee you can solve any configuration with 21 segments?

7. Could you generalize this thinking? Leapfrogging your way up . . . this is called **INDUCTION**.