I. **Other combinatorial structures with phase transitions:**
   a. Example: Let \( N(n,p) \) be a subset of integers \( \{1,2,...,n\} \) where an integer is selected with probability \( p \).
      - For \( N(365, 0.1) \), the expected size of \( N \) is \( \mathbb{E}(|N|) = 36.5 \).
      - Any monotonic property of this will have a threshold.
         i. Properties like adjacent integers.
   b. Example: Sudoku [GRAPH]
      - One can develop simple rules to fill the graph in.
      - In addition, to create puzzles, start with a filled grid and randomly remove numbers.
      - These puzzles can vary in difficulty: the number of numbers to remove from the graph to create an “easy” puzzle has a sharp threshold.
   c. Example: Boolean formulas in Conjunctive Normal Form (CNF)
      Let \( f(x_1, x_2, ..., x_n) = (x_1 + x_2 + x_3)(x_1 + x_4 + x_5) ... \) [\( x \) means a literal]
      - \( n \) variables
      - \( k \) literals per clause
      - \( cn \) clauses [if all are true => the formula is true]
      Is there an assignment to variables for which the formula is true?
      Of the \( 2^n \) assignments of literals to variables, are they all true?
      As one increases the number of clauses, the probability of the these questions being satisfied decreases.

   **Upper Bound on Threshold \( r_k \):** [Where \( r_3 = 3 \) literals per clause]
   - For a given assignment (of literals to variables), a random clause is satisfied with probability \( [1-(1/2^k)] \)
   - There are \( cn \) independent clauses, so the probability they are all correct is \( [1-(1/2^k)]^{cn} \)
   - Since there are \( 2^n \) assignments:
     \[
     \mathbb{E}(# \text{ of satisfying arguments}) = 2^n \left[1-(1/2^k)\right]^{cn}
     \]
     - If \( c = 2^k \ln(2) \), then \( \mathbb{E}(# \text{ of satisfying arguments}) = 2^n \left[1-(1/2^k)\right]^{cn} = 2^n \left[1-(1/2^k)\right]^{(2^k \ln(2))} = 2^n e^{-n \ln(2)} = 2^n 2^{-n} = 1 \)

   Therefore: if \( c > 2^k \ln(2) \), \( \mathbb{E}(# \text{satisfying arguments}) \rightarrow 0 \) [if is not true]
   c. \( c < 2^k \ln(2) \), \( \mathbb{E}(# \text{satisfying arguments}) \rightarrow \infty \) [go to 2nd mmt. arg.]

   - **PROBLEM:** In this case, the \( (x_1, x_2, ..., x_n) \) are not statistically independent. Therefore, cannot use the 2nd moment argument.
Lower Bound on Threshold $r_k$: use unit clause heuristic!

**Unit Clause Heuristic:**
Proves $r_k \geq (2^k/k)$ and $(2^k/k) \leq$ threshold $\leq 2^k \ln(2)$

**definition:**

i. If clause has 1 literal, set the literal to be true
ii. Delete all clauses with this literal, because we know they're true too
iii. Delete the compliment of the literal wherever it occurs
iv. Otherwise, pick literals at random and set to equal true
v. As long as there are fewer than $2^k \ln(2)$, the formula is satisfied

For $k = 3$, the threshold is in the bounds: (2.667, 5.54)

II. Large Sparse Graphs:
   a. Example (like in the homework):
      
      | size of component: | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | ... | 100 |
      | # of components:   | 48 | 179 | 50 | 25 | 14 | 6 | 4 | 6 | 1 | 1 | ... | 1 |
      
      - One might think these are all the components (because it goes to 0...)
      but when the bottom column is added up, there are ~1000 nodes missing...why?
      - This is because there is a component of size 1436! Weird? **No.**
        - This is the large component
          i. $P = 0$ isolated vertices (small trees $\leq \ln(n)$)
          ii. $P = 1/n$ a cycle
          iii. $P = n^{2/3}$ one component
          iv. $P > 1/n$ there will be a giant component $[cn]$ that starts
              to swallow up the other components until only
              isolated vertices left. (hence the component of
              1436 in the example)
          v. $P = \ln(n)/4n$ a connected graph.
   b. Algorithm to find components:
      i. Start with a random vertices
      ii. Find all the vertices it is adjacent to
      iii. Go look at all these vertices, etc

**Discovered:** the new vertices
**Explored:** the vertices you've looked at the adj. vertices of
**Frontier:** discovered - explored

After running the algorithm for a while ($n^2$ of the vertices), you
will only be able to find $d/2$ new vertices. When you've found $n[(d-1)/d]$ vertices, will only be able to find one more vertex.

[GRAPH]
i. Let $G(n, d/n)$ be the frontier
   - When component finding algorithm runs, the size of the frontier is never 0 at $n[(d-1)/d]$.
   - This proves there are no components of a moderate size.
   (There is a gap)

Proof:
Let $Z_i =$ \#of vertices discovered in the first $i$ steps of the search.

**Claim:** $Z_i \sim 1 + \text{Bin}(n-1, 1-[1-(d/n)])$

- Why? Because if doing a search, the probability that a vertex is not adjacent to the vertex the search is starting with is $[1-(d/n)]$. Therefore, the probability it is adjacent is $1-[1-(d/n)]$.

...to be continued....