1. The procedure below takes an array of integers and determines if some elements occurs three (or more) times in the array. Which of the following big-$O$ estimates: $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$, $O(n^2 \log n)$, $O(n^3)$, $O(n^3 \log n)$, $O(n^4)$, and $O(2^n)$ best describes the worst-case running time of the algorithm.

```java
public boolean hasThreeEqual(int[] arr) {
    int n = arr.length;
    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            if (arr[i] == arr[j]) {
                for (int k = j + 1; k < n; k++) {
                    if (arr[j] == arr[k]) {
                        return true;
                    }
                }
            }
        }
    }
    return false;
}
```

2. Show $B(n) \leq n!$.

3. Let $V$ be a vector space of dimension 4 over a finite field with $q$ elements and let $L = \text{Sub}(V)$ be the lattice of subspaces. Find $n = |L|$ and the number $e_\prec = e_\prec(L)$ of covers in this lattices. If you express $e_\prec$ as powers of $n$ and take the limit as $q$ goes to infinity, it has the form $cn^r$ plus lower order terms. Find $c$ and $r$.

4. An $n \times n$ matrix is *doubly stochastic* if $0 \leq a_{ij} \leq 1$ and each row sum and each column sum is 1. Prove that if $A$ is doubly stochastic then it has a diagonal all of whose entries at nonzero.