Practice Problems

1. Prove/disapprove: $2n^3 + 35n - 46 \in O(n^3)$
2. Prove/disapprove: $2^{2n} \in O(2^n)$
3. Prove/disapprove: $3^n \in O(2^n)$
4. Prove/disapprove: For any natural number a, $\log n \in O(n^{1/a})$
5. Prove/disapprove: $100n^2 \in O(2^n)$
6. Determine the worst-case runtime as a function of n, give and prove a tight upper bound (big-O).

   (a)
   ```
   m = √n
   while (m > 0) {
       for (i = 0; i < 2^m; i++) {
           // something constant
       }
       m--
   }
   ```

   (b)
   ```
   for i = 0 to n
       for j = 0 to n/2
           for k = i + j to n
               // something constant
   ```

   (c)
   ```
   for (i=1; i<=n; i++){
       j = 1
       while (j<= n){
           <do something constant>
           j = 2*j;
        }
   }
   ```

(d)
```
i = 1;
while (i<=n){
    for(j=1; j<=2i; j++){
        <atomic operation>
    }
    i=3*i;
}
```

7. Prove by induction that

$$\sum_{i=1}^{n} i^2 = 1^2 + 2^2 + 3^2 + ... + n^2 = \frac{n(n+1)(2n+1)}{6}$$

8. Assume n is even. Compute the sum $\sum_{i=\frac{n}{2}}^{n} i$. You can use the sum $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

9. Prove by contradiction that there is no largest integer.

10. Prove by contradiction that there is no largest prime number.

11. Prove or disprove the following:

    (a) $3^n \in O(2^n)$

    (b) $100n^2 \in O(2^n)$

12. Derive the runtime of the following:

```
for i=1 to n
 for j=i to n
   sum = 0;
   for k=i to j
     sum = sum + A[k];
     if (sum == 0) return true
return false
```

13. Derive the runtime (express the number of steps in terms of big-O of n) of the following code snippet.

```
for (i=1; i<=n; i++) {
  for (j=1; j<=n; j++) {
    for (k=j; k>=i; k--)  {
       <do-some-atomic-ops>
    }
  }
}
```

14. You are given two sorted arrays A and B of integers. Write an algorithm to find all integers that are common to A and B. You can assume that there is no repetition of elements in each list.

```
Example: A = [3, 6, 9, 10, 11]
         B = [2, 6, 11, 15]
      Ans: C = [6, 11]
```

    Discuss the derivation of the runtime of your algorithm.

15. Given a sorted array of integers and a key of type integer, write an algorithm to output the first and last indices of the array where the key is located. If the key is not present in the array, the output should be `(-1,-1)`.

16. Given an array of integers, write an algorithm and its runtime to check whether there are two elements whose sum is equal to K.

17. Given an array of integers (no duplicates), write an algorithm to check whether there are four distinct elements x, y, z and w such that $x + y = z + w$.

18. Given an array of integers, write an algorithm to check whether there is a subarray whose sum is multiple of a given integer K.

    This problem utilizes the strategy of computing prefix-sums. A prefix-sum based problem will be done in class on Tuesday Sept 12. This is a variant of that problem.

19. Prove that the height of a heap tree is $\lfloor \log_2(n) \rfloor$ where n is the number of elements in the tree.

20. Draw a binary min heap tree and give the corresponding array where the elements are inserted to the heap in the following order:

$$50, 64, 87, 90, 23, 14, 2, 51, 27$$

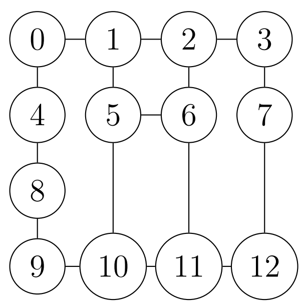
    Draw the tree after each insertion.

21. Consider the following heap sort algorithm (using min heap).

```
Build min heap H given the array with n elements
for i = H.length down to 1 {
    send H[1] to output
    swap H[1] and H[i]
    Decrement heap size
    heapifyDown(H, 1)
}
```

Apply this pseudocode to the example [5, 13, 2, 25, 7, 17, 20, 8, 4] and show all the trees in building the min heap and all the trees in the sorting process. Note that the array H that represents the heap tree is associated with a "size" variable which keeps track of the last index in the heap and the size decreases in the forloop.

22. Consider the following undirected graph $G = (V, E)$:



(a) Specify $|V|$ and $|E|$ for the graph $G = (V, E)$:

(b) Write out the adjacency list representation for the graph G. At the list in each node, the vertices connected to it should be in increasing order of vertex number.

23. Derive the runtime of the following loop structure as a function of the **input size**. Here, the **input size**, denoted by n, is the number of bits required to represent the input, i.e., $x \leq 2^n$.

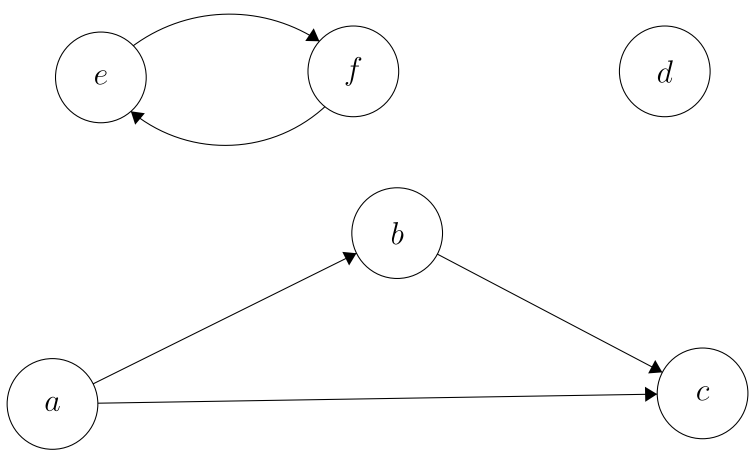
```
is_prime(x) {
  for i = 2 to x - 1 {
    if x % i == 0
      return false
  }
}
return true
```

24. You are given a sorted array A of integers, and a target integer T. Write an algorithm to verify whether there exists two integers x and y in L, such that $x + y = L$.

```
Example: A = [3, 6, 9, 10, 11]

T = 16
Ans: yes

T = 14
Ans: yes

T = 11
Ans: no

T = 22
Ans: no
```

Discuss the derivation of the runtime of your algorithm.

25. Write the given graph as an adjacency matrix and as an adjacency list.



26. Given an undirected graph represented as an adjacency list, write an algorithm that returns the equivalent adjacency matrix.

27. Given a directed graph represented as an adjacency list, write an algorithm that returns the equivalent adjacency matrix.

28. Given an undirected graph represented as an adjacency matrix, write an algorithm that returns the equivalent adjacency list.

29. Given a directed graph represented as an adjacency matrix, write an algorithm that returns the equivalent adjacency list.

30. Given a directed graph $G = (V, E)$ represented as an adjacency matrix, compute the reverse graph $G^R = (V, E')$ where $E' = \{(u, v)|(v, u) \in E\}$.

31. Given a directed graph $G = (V, E)$ represented as an adjacency list, compute the reverse graph $G^R = (V, E')$ where $E' = \{(u, v)|(v, u) \in E\}$.