

## Practice Problems 1

**Problem 1.** Big O notation. For each group, write these functions of  $n$  in big-O notation and sort them by growth rate. When two functions have the same big-O, list them in any order.

- |     |                    |               |                    |
|-----|--------------------|---------------|--------------------|
| (a) | $(3/2)^n$          | $3^{(n/2)}$   | $2^{(n/3)}$        |
| (b) | $\lg n$            | $\ln n$       | $\lg(n^2)$         |
| (c) | $n^{\lg 4}$        | $2^{\lg n}$   | $2^{(2 \lg n)}$    |
| (d) | $\max(50n^2, n^3)$ | $50n^2 + n^3$ | $\min(50n^2, n^3)$ |
| (e) | $n + 100$          | $10^{100}$    | $100n$             |

**Problem 2.** Find the big-O expression for function  $f(n) = 1 + c + c^2 + \dots + c^n$  for the cases below and briefly justify your answer.

- (a)  $0 < c < 1$ .
- (b)  $c = 1$ .
- (c)  $c > 1$ .

**Problem 3.** Find the worst case time complexity of the following algorithm, answer in big-O notation and give a brief justification.

```

1 def calculate(n):
2     count = 0
3     i = n
4     while i > 0:
5         for j in range(i):
6             count += 1
7         i //= 2
8     return count

```

**Problem 4.** Graph DFS.

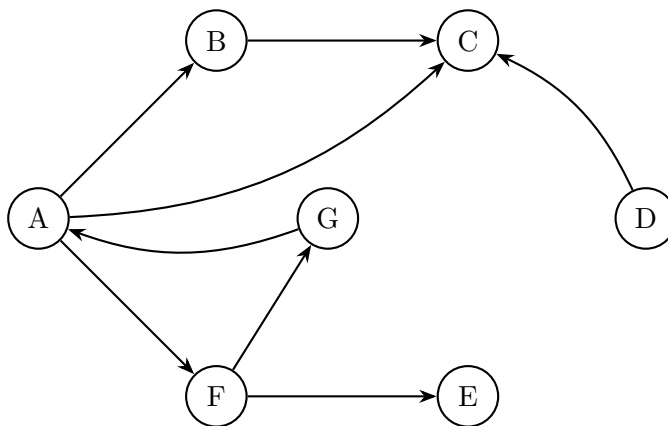


Figure 1: Problem 4 graph.

- (a) Write the adjacency matrix and adjacency list of the graph.
- (b) Traverse the graph in DFS starting from node A and write the order in which the nodes are visited. When there are multiple nodes to go next, pick the node in lowest alphabetic order.
- (c) Based on your search tree in (b), label the type of each edge (tree, forward, backward, cross).

**Problem 5.** Tripartite graph. A graph  $G = (V, E)$  is tripartite if its vertex set  $V$  can be partitioned into three mutually disjoint sets  $V_1$ ,  $V_2$ , and  $V_3$  such that no edge connects two vertices within the same set. Let  $n_1 = |V_1|$ ,  $n_2 = |V_2|$ , and  $n_3 = |V_3|$ . Assuming the graph is undirected, what is the maximum number of edges that  $G$  can have, expressed in  $n_1$ ,  $n_2$ , and  $n_3$ ? Briefly justify your answer.

**Problem 6.** Graph connectivity. Let  $G$  be an undirected graph on  $n$  nodes, where  $n$  is an even number. If every node of  $G$  has degree at least  $n/2$ , then  $G$  is connected. The degree of a node means how many edges are connected to it. Do you think it is true or false? Give a brief proof or a counterexample.