MCAMC

2019

MCAMC: Team Round

Do not turn the page until you are told to do so.

This section of the competition is to be completed by **your team** within **1 hour**. This section consists of **20 questions**. No calculators, notes, compasses, smartphones, smartwatches, or any other aids are allowed. Only answers recorded on this page will receive credit. Answers must be exact (do not approximate π) and in simplest form, with all fractions expressed as improper fractions. Examples of unacceptable answers include: $\frac{4}{6}$, $1\frac{1}{3}$, $3+2$. Examples of acceptable answers include: $\frac{2}{3}$, $\frac{4}{3}$, 5. There is no need to include units for any answer, and the units are always assumed to be the units in the question. Either exact decimal answers or improper fractions will be accepted (i.e. 0.25 and $\frac{1}{4}$ are both acceptable).

Team Name: ____________________ Team ID: ____________

1.	________	6.	________	11.	________	16.	________
2.	________	7.	________	12.	________	17.	________
3.	________	8.	________	13.	________	18.	________
4.	________	9.	________	14.	________	19.	________
5.	________	10.	________	15.	________	20.	________

1 What is a Graph?

When you hear the word "graph" in math, you might be inclined to think about the plot of something like $y = 2x + 3$ on a coordinate plane. Today, we'll look at a completely different kind of graph that has nothing to do with the plots you may be familiar with – one that is studied in a field called *graph theory*. Graph theory is an important and fundamental subject in computer science, and finds applications in all kinds of problems, including scheduling, optimization, and more. We'll work our way toward one specific application today, namely, in search algorithms.

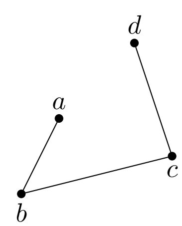
With that said, let's get started.

Definition. A *graph*, denoted G, is made up of some points (called *vertices*) and some lines between pairs of points (called *edges*). Any pair of distinct vertices either has 1 edge or no edge between them. It is possible to have no vertices and/or no edges in a graph.

Technically, this is actually the definition of a *simple graph*, but we will just call it a graph. We can visualize graphs as follows.

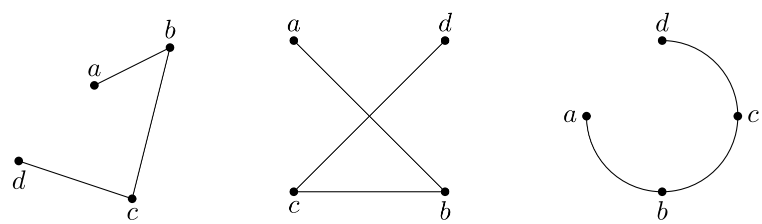
Example. Draw the graph G with vertices $\{a, b, c, d\}$ and edges $\{\{a, b\}, \{b, c\}, \{c, d\}\}$.

Solution. We draw all the vertices, and then connect the pairs specified in the list of edges.



□

Note that the placement of the vertices and the shape of the edges don't matter. We could've drawn the above graph as any of the following:



To reiterate, all of the above drawings represent *the same graph*. The only things that are important are the names of the vertices and whether or not each pair of vertices is connected by an edge.

Definition. The *degree* of a vertex is the number of edges that touch it (*incident* to it).

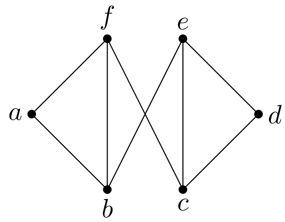
Example. Find the degree of every vertex in the graph with vertices $\{a, b, c, d\}$ and edges $\{\{a, b\}, \{b, c\}, \{c, d\}\}$.

Solution. Referring to the picture we drew before (or by looking at the list of edges), we see that a is incident to 1 edge, b is incident to 2 edges, c is incident to 2 edges, and d is incident to 1 edge. So we have the degree of a is 1, the degree of b is 2, the degree of c is 2, and the degree of d is 1. □

Problem 1. What is the maximum number of edges in a graph with 4 vertices?

Problem 2. How many distinct graphs are possible with 3 vertices a, b, and c, and some (possibly 0) number of edges?

Problem 3. In the following graph, how many vertices have degree 2 or 3?



Problem 4. Let the sum of the degrees of all the vertices in a graph be denoted s. Let m be the number of edges in a graph. What is the relationship between s and m? (Write an equation.)

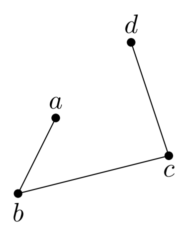
Problem 5. Suppose you took the graph from Problem 1 and counted the degree of each vertex. What is the sum of these degrees?

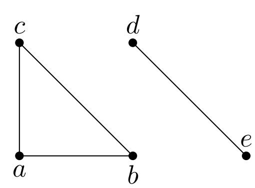
Problem 6. How many graphs with 5 vertices have the property that every vertex has odd degree? (Hint: Problem 4 might help you with this.)

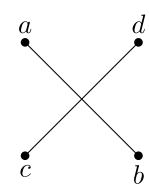
2 Connected Graphs

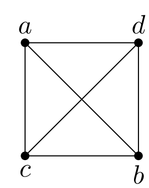
Definition. A graph is *connected* if for every pair of vertices, there exists a sequence of edges that one could walk along to get from one vertex to the other. If a graph is not connected, it is *disconnected*.

Example. Determine if the following graphs are connected.

1. 

2. 

3. 

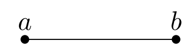
4. 

Solution.

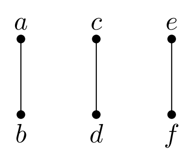
1. This is connected. You can check that no matter which two vertices you pick, there is a way to walk between them via the edges.
2. This is disconnected. There are many pairs between which no path along edges exists. For example, there is no path along edges between c and d. You can also see that the graph is in two parts, so it is disconnected.
3. This is disconnected. For example, there is no path along edges between a and c. Specifically, a can only go to b and b can only go to a. Walking along a sequence of edges means that you must walk the entire edge before choosing another edge to walk on. In other words, remember that you are only able to turn at vertices (not at the crossing in the middle). (Can you redraw this graph with a different placement of the vertices to make it more obvious that it is disconnected?)
4. This is connected. In fact, every pair of vertices is directly connected by an edge. □

Problem 7. Which of the following are connected? (Write "A", "B", "both", or "neither".)

A: 

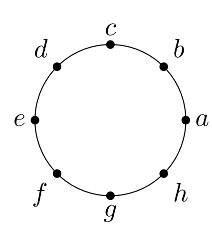
B: 

Problem 8. How many ways can you add 2 edges to the following graph in order to turn it into a connected graph? (Hint: reminder that curved edges are perfectly fine.)



Problem 9. How many connected graphs exist on 4 vertices a, b, c, and d?

Problem 10. In the following graph, what is the minimum number of vertices that must be removed in order to disconnect the graph? (When a vertex is removed, all edges connected to that vertex are removed as well.)

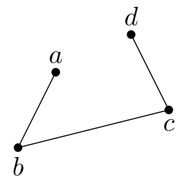


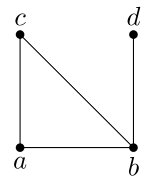
3 Trees

Trees are a special kind of graph with some amazing properties, and you will uncover some of these properties in the problems in this section. But before we define a tree, we must define a cycle.

Definition. A *cycle* is a path through some vertices of the graph (without backtracking) where the first vertex traversed is also the last.

Example. Determine if the following graphs have cycles.

1. 

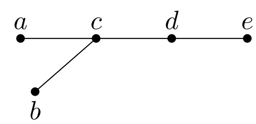
2. 

Solution.

1. This graph does not have a cycle. No matter what path you take, you can not come back to your starting vertex without backtracking.
2. This graph has a cycle. Namely, $a \to b \to c \to a$ is a cycle. $\square$

Definition. A *tree* is a graph that is connected and has no cycles.

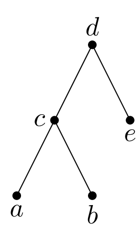
For example, the first graph in the previous example is a tree. The following is also a tree.



We often like to visualize trees by applying the following procedure:

1. Choose any vertex. Call this vertex the *root* of the tree.
2. Redraw the graph so that vertices are organized in *layers* below the root. Vertices in each layer should connect only with vertices in the layer directly above it or in the layer directly below it. (An example will make this clear.)

Drawing the above tree with root at d, for example, we get:

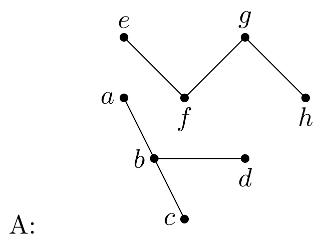


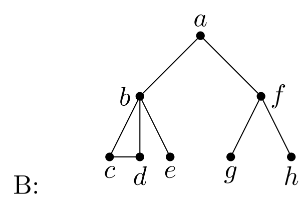
Note that this is the same graph, just drawn differently. The top layer is just the root d, then c and e form a layer under d, and a and b form a layer under c and e.

Definition. The *height* of a tree drawn with a root is the number of layers it has minus one.

For example, the above graph has 3 layers, so it has height 2.

Problem 11. Which of the following are trees? (Write "A", "B", "both", or "neither".)





Problem 12. If n is the number of vertices in a graph, then which of the following statements must be true? (There may be more than one. List all that are true.)

1. If a graph has $n - 1$ edges and has no cycles, then the graph is a tree.
2. If a graph is connected and has $n - 1$ edges, then the graph is a tree.
3. If a graph is connected and the removal of any one edge renders a graph disconnected, then the graph is a tree.
4. If a graph has no cycles, but adding an edge anywhere creates a cycle, then the graph is a tree.

Problem 13. What is the minimum height of a tree with 256 vertices if each vertex can have no more than 2 children?

Problem 14. What is the largest degree a vertex in a tree can have? Express your answer in terms of n, the number of vertices in a tree.

Problem 15. Suppose you want to color the vertices of a tree such that any two vertices connected by an edge have different colors. What is the minimum number of colors you need to guarantee that you will be able to color any tree?

4 Binary Search and Binary Search Trees

Finally, we arrive at binary search, an efficient way to determine if a number is in a list. To motivate this section, suppose a computer has a list of numbers [1, 3, 4, 7, 9, 9], and you want to ask the computer if the number 8 is in the list.

One way the computer can find out is to check every number in the list and see if any of them are 8. But a much faster way (if the list is in non-decreasing order, as it is here) is to run *binary search*, which only checks a few of the numbers.

Procedure (Binary Search). Given a list of numbers and the number you are looking for n,

1. If the list contains just a single entry, compare it with n. If they are the same, then n is in the list. If they are not the same, then n is not in the list.
2. Otherwise, compare the middle element of the list with n. (If there are an even number of elements, compare with the element just left of center.) If they are the same, then n is in the list. If n is bigger, run binary search on the right half of the list. If n is smaller, run binary search on the left half of the list.

Example. Run binary search on [1, 3, 4, 7, 9, 9] to determine if 8 is in the list.

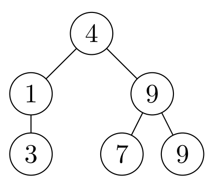
Solution. We have our list [1, 3, 4, 7, 9, 9] and $n = 8$. Following the procedure, we first skip to step 2 because the list has more than one entry. We compare $n = 8$ with the element just left of center, 4. Because 8 is bigger than 4, we take the half of the list to the right of 4, namely [7, 9, 9], and run binary search on [7, 9, 9].

Now, we have our list [7, 9, 9] and $n = 8$. Again we skip to step 2, and we compare $n = 8$ with the middle element, the first 9. Because 8 is less than 9, we take the half of the list to the left of the first 9, [7], and run binary search on [7].

Finally, we have our list [7] and $n = 8$. The list contains a single entry, so we do step 1 now. Comparing 7 with $n = 8$, they are not equal, so 8 is not in the list. □

Although for this small list, the speed increase is not that apparent, you can imagine how much faster this is when the list is big, because every comparison cuts the size of the list by a factor of two.

Binary search is closely related to trees. In fact, we can draw the following binary search tree corresponding to the list [1, 3, 4, 7, 9, 9].



Notice that when we looked for the number 8, we compared it to 4, then 9, then 7. This corresponds to one branch down the tree from the root to the bottom. If we were looking for the number 3 for example, we would have compared it to 4, then 1, then 3, corresponding to another branch. You will investigate the relationship between binary search trees and binary search in greater detail some of the problems below.

Problem 16. Suppose you run binary search on the list [1, 2, 2, 2, 2, 3, 4, 5, 6, 7, 8]. If the number you are looking for is 2, which 2 will be found by the computer? (Answer "first", "second", "third", or "fourth".)

Problem 17. Suppose you run binary search on the list [1, 2, 2, 2, 2, 3, 4, 5, 6, 7, 8]. If the number you are looking for is 9, how many comparisons will be made before the computer determines that 9 is not in the list?

Problem 18. You and your friend are playing a game where your friend randomly chooses a whole number between 1 and 127, inclusive, and you are trying to guess it. Your friend tells you whether your guess is correct, too high, or too low. Using the best strategy, what is the maximum number of times you will have to guess, regardless of what number your friend chose?

Problem 19. You and your friend are playing a game where your friend randomly chooses a whole number between 1 and N, inclusive, and you are trying to guess it. Your friend tells you whether your guess is correct, too high, or too low. Fill in the blanks of the following statement: If x is the ____________ (word or phrase) of the binary search tree corresponding to the best strategy, then the maximum number of times you will have to guess is __________________ (expression in terms of x). (Hint: there are multiple possible answers, but one is much simpler than the others.)

Problem 20. A magician welcomes you to a room with 80 labeled paintings. He tells you that behind one painting is a pile of treasure, and if you can guess which one it is, you will be able to keep the treasure. Furthermore, during each turn, you can guess two numbers, and if neither of the guesses is correct, the magician will tell you whether the label of the painting is less than the two numbers, in between the two numbers, or greater than the two numbers. What is the maximum number of turns you will need to guarantee that you will find the treasure, using the best strategy? (Hint: Think about binary search, but ternary (meaning 3), not binary.)