Introduction to Graph Theory

A graph is two related sets: $V(G)$, the vertex set, and $E(G)$, the edge set. Each element of $E(G)$ is a set containing exactly two elements of $V(G)$.

We often visualize graphs by drawing the vertices as dots and the edges as lines between them. Here is an example of a graph with vertex set $\{A, B, C, D, E, F\}$ and edge set $\\{\{A, B\}, \{B, C\}, \{A, E\}, \{C, E\}, \{A, D\}, \{C, F\}, \{D, E\}, \{E, F\}\}$.

Note that we have defined an edge as a set of cardinality two: this means that a vertex cannot have an edge to itself, and there is no sense of direction in edges. Additionally, since edges are contained in a set, there can be at most one edge between any two vertices.

- If $u, v \in V(G)$, $u$ is adjacent to $v$ if $\{u, v\} \in E(G)$. (Note that by this definition a vertex is not adjacent to itself).
- The degree of a vertex is a count of the number of vertices it is adjacent to. Formally, for a vertex $v$, $\deg(v) = |\{u|\{v, u\} \in E(G)\}|$.
- The empty graph on $n$ vertices has an empty edge set.
- The complete graph on $n$ vertices $K_n$ has all possible edges between vertices, that is, $E(G) = \{(u, v)|u, v \in E(G), u \neq v\}$. Every pair of vertices is adjacent.
- A **subgraph** $G'$ of a graph $G$ is a graph such that $V(G') \subseteq V(G)$ and $E(G') \subseteq E(G)$. Since $G'$ is a graph, each edge in $E(G')$ must be between two vertices in $V(G')$.

**Task 1**

Let $G$ be a graph such that $V(G) = \{a, b, c, d, e, f\}$ and $E(G) = \{\{b, c\}, \{a, b\}, \{d, e\}, \{c, d\}, \{d, b\}\}$.

1. Draw $G$.

![Graph G](image)

2. What is the degree of each vertex in $G$?

<table>
<thead>
<tr>
<th>Vertex</th>
<th>Degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>1</td>
</tr>
<tr>
<td>$b$</td>
<td>3</td>
</tr>
<tr>
<td>$c$</td>
<td>2</td>
</tr>
<tr>
<td>$d$</td>
<td>3</td>
</tr>
<tr>
<td>$e$</td>
<td>1</td>
</tr>
<tr>
<td>$f$</td>
<td>0</td>
</tr>
</tbody>
</table>

3. Say we take a vertex $v$ uniformly at random from $V(G)$. Let $X$ be a random variable that represents the degree of $v$. What is the probability distribution of $X$?

<table>
<thead>
<tr>
<th>$X_i$</th>
<th>$0$</th>
<th>$1$</th>
<th>$2$</th>
<th>$3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Pr(X_i)$</td>
<td>$\frac{1}{6}$</td>
<td>$\frac{2}{6}$</td>
<td>$\frac{1}{6}$</td>
<td>$\frac{2}{6}$</td>
</tr>
</tbody>
</table>

4. What is $E[X]$?

$$E[X] = 0 \cdot \frac{1}{6} + 1 \cdot \frac{2}{6} + 2 \cdot \frac{1}{6} + 3 \cdot \frac{2}{6} = \frac{2}{3} + \frac{2}{3} + 1 = \frac{5}{3}$$

5. **Optional:** Can you develop a general expression for $E[X]$ in terms of $|V|$ and $|E|$?

$$E[X] = \frac{2|E|}{|V|}$$. The sum of all degrees is $2|E|$, and, by the linearity of expectation, we can do the calculation.
Definitions

- A **path** is a sequence of vertices, consecutive elements of which are endpoints of edges in the graph.

- A **simple path** is a path in which no vertex occurs more than once.

- A **cycle** is a path in which the first and last vertex are the same.

- A **simple cycle** is a cycle in which all but the last vertex form a simple path.

- A **Hamiltonian tour** is a simple cycle that includes all the vertices in the graph.

- Two vertices are **connected** if there exists a path between them (or if they are the same vertex).

- An **Eulerian tour** is a cycle that visits every edge exactly once.

- A **Bipartite graph** (optional) is a graph whose vertex set can be partitioned to two parts, and no edge connects a pair of vertices belonging to the same part. Any graph that is a subgraph of a complete bipartite graph is bipartite.

- Let $G = (V, E)$ and $G' = (V', E')$. $G$ and $G'$ are **isomorphic** (optional) if there is a bijection $f : V \rightarrow V'$ such that for every pair of vertices $\{x, y\} \in E$ if and only if $\{f(x), f(y)\} \in E'$. In other words, two graphs are isomorphic if we can find a consistent relabeling of the edges and vertices of the two graphs.
Task 2

Consider the following map of planets (labelled A–G) and the routes between them:

![Planets diagram]

a. A coffee addict wants to explore all the routes between a number of cafes. Can a cycle be found which traverses each route only once? Particularly, find a path that starts at A, goes along each route exactly once, and ends back at A.

In other words, find an Euler tour for the graph!

Many examples, one possibility is A → B → C → D → E → F → B → G → C → E → G → F → A. The coffee addict travels along each route (edge) just once but may visit a particular cafe (vertex) several times.

b. A coffee addict wants to visit a number of cafes. Can a cycle be found that visits each cafe exactly once? Particularly, find a path that starts at A, goes to each cafe exactly once, and ends back at A.

In other words, find a Hamiltonian tour for the graph!

One such solution is A → B → C → D → E → G → F → A. The coffee addict visits each cafe (vertex) just once but may omit several of the routes (edges) on the way.

Task 3

a. Given a certain labelling of the vertices, how many distinct graphs can be made with 4 vertices?

For each pair of vertices, there exists a possible edge. Thus, there are \( \binom{n}{2} \) possible edges. Each can be either in or not in the edge set of a graph, so there are \( 2^{\binom{n}{2}} \) possible graphs. When \( n = 4 \), this quantity is equal to \( 2^{\binom{4}{2}} = 2^6 = 64 \).

b. For any graph \( G \), explain why

\[
\sum_{v \in V(G)} \deg(v) = 2|E(G)|.
\]
The degree of a vertex is a count of the number of vertices it is adjacent to; hence, the number of edges that contain the vertex. When we add up all the degrees, we are counting each edge twice since each edge is a set of two vertices.

c. Tim Horton’s is having an event where two customers that purchase a pastry to split receive 10% off their order. 35 CS22 students are planning on taking advantage of this offer by having each student split pastries with 3 other students. Prove that it is impossible to do so.

Let each student be a vertex such that there exists an edge between two student if and only if they are splitting a pastry. Each of the 35 vertices has degree 3, so the sum of the degrees is 105. We just proved that this equals 2 times the number of edges, which means there are 52.5 edges. Since 52.5 isn’t an integer, this division is impossible.

d. Optional: Prove that in any graph there exist two vertices with the same degree. 
*Hint*: What are the possible degrees of a vertex in a graph with \( n \) nodes?

A vertex can have degree from 0 to \( n - 1 \). However, if a vertex has degree \( n - 1 \), there can’t be a vertex with degree 0 because that vertex is adjacent to every other. So, in this case, there are \( n - 1 \) possible values for the degree of a vertex, and \( n \) vertices: by the pigeonhole principle, two must have the same degree. On the other hand, if there is no vertex of degree \( n - 1 \), then vertices can have degree 0 through \( n - 2 \): still \( n - 1 \) options, so by the pigeonhole principle two have the same degree.

Checkpoint 1 — Call over a TA!
Trees!

- A cycle is a path that starts and ends with the same vertex.
- A graph is cyclic if it contains a simple cycle and acyclic otherwise.
- A graph is connected if there is a simple path between each pair of vertices (that is, all vertices are connected).
- A tree is a connected, acyclic graph.
- A forest is an acyclic graph. In other words, a forest is a set of trees.
- A leaf of a tree is a vertex with degree 1.
- A spanning tree of a graph \( G \) is a subgraph \( T \) of \( G \) such that \( V(T) = V(G) \) and \( T \) is a tree.

Task 4

In 1997, a chess-playing computer program called Deep Blue beat the reigning world champion of chess, Garry Kasparov, in a 6-round series. Although it was unclear whether computers can reason as well as humans, the victory did establish that computers are better at evaluating complex game trees, the basis of computer-based game strategies.

A game tree shows all possible playing strategies of both players in a game. Each vertex in the tree represents a configuration of the game as well as an indication of whose turn it is to play next. We illustrate game trees in a much simpler context of tic-tac-toe.

The first three layers of the game tree of tic-tac-toe:

The root of the tree is the initial configuration of the game, which is an empty grid in the case of tic-tac-toe. The children of a configuration \( c \) are all the configurations that can be reached from \( c \) by a single move of the correct player. A configuration
is a **leaf** vertex in the tree if the game corresponding to that configuration is over (won, lost, tied).

Here, we see a subtree near the end of the game:

![Subtree](image)

1. How many leaf nodes are in the subtree drawn above?
   
   6

2. What is the length of the longest path from a root to a leaf in a tree represented by the full tic-tac-toe game?
   
   9, since the longest tic-tac-toe game lasts 9 moves.

3. **Optional:** What is the length of the longest path between any two vertices in a tree represented by the full tic-tac-toe game?
   
   18. The path would be from a leaf to another leaf after passing through the root node, which is $9 + 9 = 18$.

**Task 5**

1. Prove that, in a tree, there is exactly one simple path between any two vertices.

   The graph is connected, so there must be at least one path between any two vertices. Assume there were two paths between two vertices: there is a point where they stop being the same path, and a point where they rejoin at the end. If we put together the parts between these points, we get a cycle.
2. One way of defining a tree is as a **minimally connected, maximally acyclic graph**. That is because if we remove an edge, our graph will no longer be connected and if we add an edge, the graph will no longer be acyclic. Prove these two statements.

There is only one path between two vertices, so if we remove an edge \( \{a, b\} \), there is now no way to get from \( a \) to \( b \). All vertices are connected, so if we add the edge \( \{a, b\} \), there are now two ways to get from \( a \) to \( b \) and hence a cycle.

3. **Optional:** Let \( T \) be a tree with at least two vertices. Prove that \( T \) has at least two leaves. **Hint:** Start by assuming that the longest simple path in \( T \) does not start and end with leaves. What contradiction can we reach?

Assume the longest simple path in \( T \) has a non-leaf endpoint. As all non-leaf vertices have degree greater than 1 and our path is simple (so the endpoint only appears once, there is at least one edge incident to the ending vertex that is unused. Extend the path by also traversing this edge. The path will still be simple because, if we already used the new vertex, we would now have two ways of getting from the endpoint to the vertex. So, we have a longer simple path, which contradicts our assumption that our original path was longest. So, both end vertices are leaves.
Induction on Graphs

Many graph theory questions involve induction. When doing induction on graphs, we often use a technique referred to in CS0220 as “build-down” induction. It is just regular induction, with a little extra care taken to make sure it applies to objects more complicated than integers. To review, here are the steps of an inductive proof:

1. Define the predicate $P(n)$.
2. Show that the base case is true.
3. Assume the inductive hypothesis is true. If you are using standard induction, then you will assume $P(k)$ is true for some integer $k$. If you are using strong induction, then you will assume $P(i)$ is true for all $i \leq k$.
4. Show that $P(k + 1)$ is true given the inductive hypothesis.

Here is where build-down induction differs from our usual perspective. A lot of the time, we start by invoking our inductive hypothesis, and manipulating the $k$ case to get the $k + 1$ case. In build-down induction, we do the following instead:

i. Consider an arbitrary $k + 1$ case.
ii. Alter the $k + 1$ case to obtain a $k$ case.
iii. Invoke the induction hypothesis, asserting that your property holds for the $k$ case.
iv. Recover the original $k + 1$ case by undoing your alteration.
v. Prove that the property still holds despite building back up to the original $k + 1$ case.

This technique is especially useful when it is not clear how to address all $k + 1$ cases in general when starting from a $k$ case. For example, if we started with a graph with $k$ edges, where do we add an edge to obtain a general graph with $k + 1$ edges? There are a lot of ways to add this extra edge that we have to consider. To avoid this ambiguity, we can start with an arbitrary $k + 1$ case. Then, we build down to a $k$ case, at which point we can invoke our induction hypothesis. Then, we build back up to the original $k + 1$ case.

Try it out on a few sample problems!
Task 6

1. Prove that a tree with \( n \) vertices has \( n - 1 \) edges.

Let \( P(n) \) be the predicate that a tree with \( n \) vertices has \( n - 1 \) edges.

Base case: A tree with 1 vertex has 0 edges.

Inductive Hypothesis: Assume all trees with \( k \) vertices have \( k - 1 \) edges.

Inductive step: Consider a tree with \( k + 1 \) vertices. There must be some leaf, which has one edge. Remove the leaf and its edge. The resultant graph will still be acyclic, since removing an edge cannot create a cycle, and is connected because the edge we removed only connected the removed vertex to the rest of the graph. So, we now have a tree with \( k \) vertices, which by the IH has \( k - 1 \) edges. Put the removed vertex and edge back in, and we now have \( k \) edges. As our \( k + 1 \) tree was arbitrary, all trees with \( k + 1 \) vertices have \( k \) edges.

2. Optional: Prove that \( K_n \) has \( \frac{n(n-1)}{2} \) edges. Note that we could do this without build-down induction, but you should use it here for practice.

Let \( P(n) \) be the predicate that \( K_n \) has \( \frac{n(n-1)}{2} \) edges.

Base case: \( K_1 \) has 0 edges, which is \( \frac{1 \times 0}{2} \).

IH: Assume \( K_k \) has \( \frac{k(k-1)}{2} \) edges.

IS: Consider \( K_{k+1} \). We can get \( K_k \) by removing a vertex and all its incident edges, of which there are \( k \). By IH, \( K_k \) has \( \frac{k(k-1)}{2} \) edges. If we add back in the vertex and edges, we get \( \frac{k(k-1)}{2} + k = \frac{k(k-1)+2k}{2} = \frac{k(k+1)}{2} \) edges as needed.
Graph Coloring

Maps are often drawn such that no two adjacent regions are the same color. This makes it easy when looking at the map to see the distinct regions. For instance, here is a map of the United States where no adjacent states are the same color:

![United States Map](image1)

You might notice that this map was drawn with only 4 distinct colors. There are 50 U.S. states, so it might be a bit surprising that it is possible to use only four colors. This observation leads to a question that cartographers and mathematicians have asked for centuries: given a map, how many colors will we need to draw it?

Like so much else in life, this problem is really graph theory in disguise! Any map can be easily converted to a graph.

For instance, we can represent this map of a region of Europe with the displayed graph. The graph was created by making each country into a vertex, and drawing an edge between two vertices if the countries share a border (and therefore need distinct colors in the map):

![Europe Map](image2)

**Task 7**

1. Consider the map of states and territories of Australia, which can be represented as the following graph:

![Australia Map](image3)
Determine the minimum number of colors we need to properly color this graph, and show one such possible coloring.

The minimum number of colors is 3, such as in the following map:

2. *Optional*: Now, consider the map of countries in South America, which can be represented as the following graph:
Determine the minimum number of colors we need to properly color this graph, and show one such possible coloring.

The minimum number of colors is 4, such as in the following map:
Optional: Isomorphism

Task 8

Two graphs are isomorphic if there exists a matching or relabelling of the vertices of the graphs.

The two graphs above are isomorphic because we can match a with 0, b with 3, c with 1, d with 4, and e with 2. Given this matching of the vertices, it’s easy to check that the vertices are connected the same way in the graphs. Yet, it’s hard to determine if two graphs are actually isomorphic! Let’s try this ourselves:

1. Out of the three graphs below, determine which two are isomorphic.
   Hint: Which of these are bipartite graphs?

\[ G_1 \quad G_2 \quad G_3 \]

\[ G_1 \text{ and } G_3 \text{ are isomorphic. An easy way to tell is that } G_2 \text{ is the only graph with cycles of length 3. Similarly, } G_1 \text{ and } G_3 \text{ are both bipartite and thus can be colored using two colors, while } G_2 \text{ cannot. This example can also be found on Wikipedia’s [Graph Isomorphism] page.} \]
2. Let’s try an even harder example! Determine which two out of these four graphs are isomorphic.

\[ G_1 \text{ and } G_4 \text{ are isomorphic, and they’re known as Petersen’s graph! } G_2 \text{ and } G_3 \text{ are not isomorphic with } G_1/G_4 \text{ because they have cycles of length } 4. \text{ Moreover, } G_3 \text{ is the only graph with cycles of length } 3, \text{ so it is not isomorphic with any other graphs.} \]

Checkoff — Call over a TA!