Introduction to Graph Theory	Instructor: Padraic Bartlett

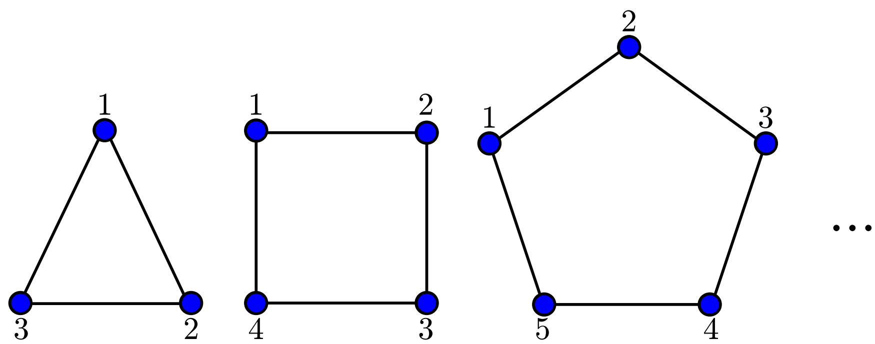
Lecture 2: A Menagerie of Graphs

Week 1 *Mathcamp 2011*

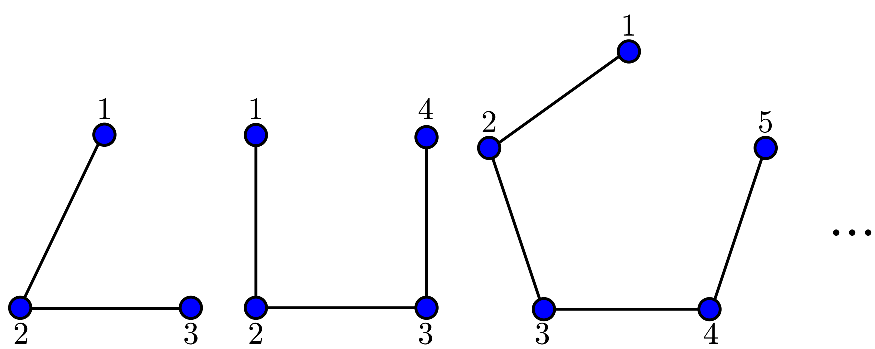
In the field of graph theory, there are several graphs which come up enough to where we've given them names. We list them here, along with some of their properties:

1 Several Key Graphs

- **The cycle graph C_n.** The cycle graph on n vertices, C_n, is the simple graph on the vertex set $\{v_1, v_2, \ldots v_n\}$ with edge set $E(C_n) = \{\{v_1, v_2\}, \{v_2, v_3\}, \ldots \{v_{n-1}, v_n\}, \{v_n, v_1\}\}$. The cycle graphs C_n can be drawn as n-gons, as depicted below:

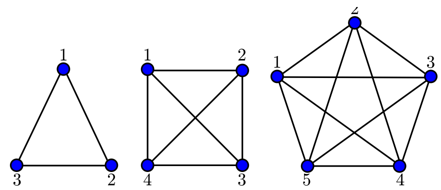


- **The path graph P_n.** The path graph on n vertices, P_n, is the simple graph on the vertex set $\{v_1, v_2, \ldots v_n\}$ with edge set $E(C_n) = \{\{v_1, v_2\}, \{v_2, v_3\}, \ldots \{v_{n-1}, v_n\}\}$. The path graphs P_n can be drawn as paths of length n, as depicted below:



  Every vertex in a P_n has degree 2, except for the two endpoints v_1, v_n, which have degree 1. P_n contains $n - 1$ edges.

- **The complete graph K_n.** The complete graph on n vertices, K_n, is the simple graph on the vertex set $\{v_1, v_2, \ldots v_n\}$ that has every possible edge: in other words, $E(K_n) = \{\{v_i, v_j\} : i \neq j\}$. We draw several of these graphs below:

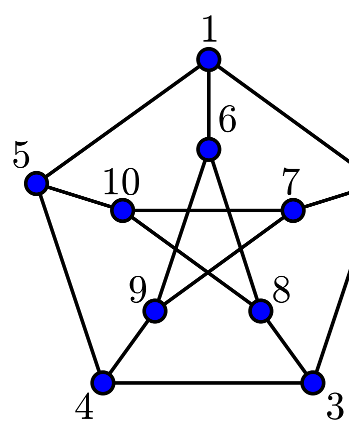


Every vertex in a K_n has degree $n-1$, as it has an edge connecting it to each of the other $n-1$ vertices; as well, a K_n has $n(n-1)/2$ edges in total in it, by the degree-sum formula. (Explicitly: every vertex has degree $n-1$ and there are n vertices, therefore the sum of the degrees of K_n's vertices is $n(n-1)$. We've shown that this quantity is twice the number of edges in the graph; dividing by 2 then tells us that the number of edges in K_n is $n(n-1)/2$, as claimed.)

- **The complete bipartite graph $K_{n,m}$.** The complete bipartite graph on $n+m$ vertices with part sizes n and m, $K_{n,m}$, is the following graph:
  - $V(K_{n,m}) = \{v_1, v_2, \ldots v_n, w_1, w_2, \ldots w_m\}$.
  - $E(K_{n,m})$ consists of all of the edges between the n-part and the m-part; in other words, $E(K_{n,m}) = \{(v_i, w_j) : 1 \leq i \leq n, 1 \leq j \leq m\}$.

  The vertices v_i all have degree m, as they have precisely m edges leaving them (one to every vertex w_j); similarly, the vertices w_j all have degree n. By either the degree-sum formula or just counting, we can see that there are nm edges in $K_{n,m}$.

- **The Petersen graph** P The Petersen graph P is a graph on ten vertices, drawn below:

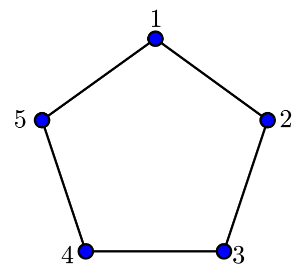


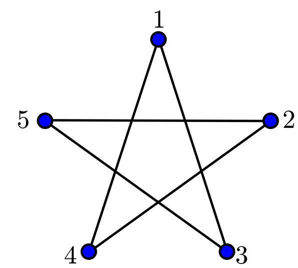
  The vertices in P all have degree three; by counting or the degree-sum formula, P has 15 edges.

2 The Concept of "Sameness"

In the graphs above, we've made a point of labeling all of the vertices in our graphs. We do this because this is part of the definition of what a graph *is* – a collection of labeled vertices and edges between them.

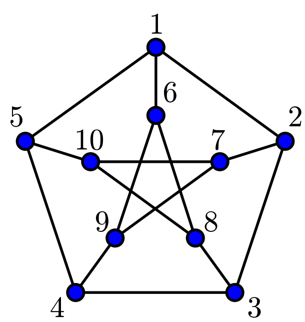
But is this really what we want for our definition? For example, consider the following two graphs:

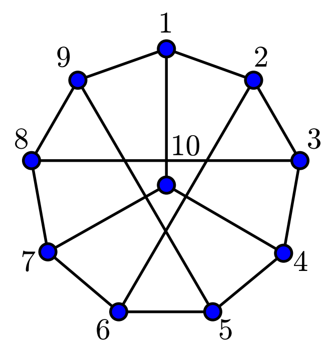




These graphs are, in one sense, different; the first graph has an edge connecting 1 to 2, where the second graph does not. However, in another sense, these graphs are representing the same situation: they're both depicting the graph sketched out by a pentagon!

For graphs like the ones in our menagerie, we don't care so much about the labeling of the vertices; rather, the interesting features of these graphs are the intersections of their edges and vertices. In other words, we want to say that both of the graphs below are "the" Petersen graph: even though they initially look rather different, there is a way of "relabeling" the vertices on the second graph so that (i, j) is an edge in the first graph iff it's an edge in the relabeled second graph.





How can we do this? What notion can we introduce that will allow us to regard such graphs as being the "same," in a well-defined sense? Well, consider the following:

Definition. We say that two graphs G_1, G_2 are **isomorphic** if and only if there is a map $\sigma : V(G_1) \to V(G_2)$ such that

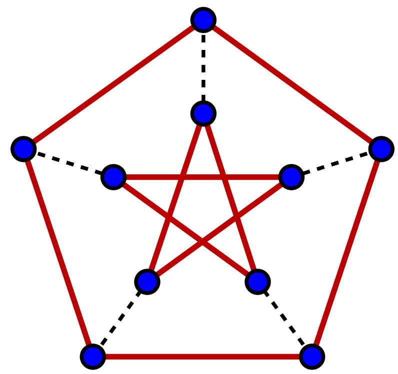
- σ matches each element of $V(G_1)$ to a unique element of $V(G_2)$, and vice-versa: in other words, σ is a way of relabeling G_1's vertices with G_2's labels, and vice-versa.
- $\{v_i, v_j\}$ is an edge in G_1 if and only if $\{\sigma(v_1), \sigma(v_2)\}$ is an edge in G_2.

We will often regard two isomorphic graphs as being the "same," and therefore refer to graphs like K_n or the Petersen graph without specifying or worrying about what the vertices are labeled.

A concept that's much more interesting (given the idea of isomorphism) is the concept of a **subgraph**, which we define below:

Definition. Given a graph G and another graph H, we say that H is a **subgraph** of G if and only if $V(H) \subset V(G)$ and $E(H) \subset E(G)$.

Example. The Petersen graph has the disjoint union of two pentagons $C_5 \sqcup C_5$ as a subgraph, which we shade in red below:



In general, when we ask if a graph H is a subgraph of a graph G, we won't mention a labeling of H's vertices; in this situation, we're actually asking whether there is *any* subgraph of G that is **isomorphic** to H.

For example, one question we could ask is the following: what kinds of graphs contain an triangle (i.e. a C_3) as a subgraph? Or, more generally, what kinds of graphs contain an odd cycle (i.e. a C_{2k+1}) as a subgraph?

We answer this in the next section:

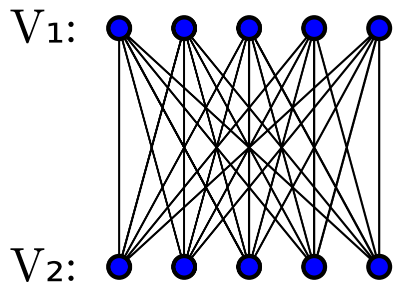
3 Classifying Bipartite Graphs

In our menagerie of graphs above, we defined the **complete bipartite graph** $K_{n,m}$. One natural generalization of this graph is to the concept of **bipartite graphs**, which we define below:

Definition. We call a graph G **bipartite** if and only if we can break the set $V(G)$ up into two parts $V_1(G)$ and $V_2(G)$, such that every edge $e \in E(G)$ has one endpoint in $V_1(G)$ and one endpoint in $V_2(G)$.

Alternately, we say that a graph is bipartite iff there is some way to color G's vertices red and blue – i.e. to take every vertex in G and assign it either the color blue or color red, but not both or neither – so that every edge has one blue endpoint and one red endpoint.

Example. The following graph is bipartite, with indicated partition (V_1, V_2):



However, there are graphs that are not bipartite; for example, C_3, the triangle, is not bipartite! This is not very hard to see: in any partition of C_3's vertices into two sets V_1 and V_2, one of the two sets V_1 or V_2 has to contain two vertices of our triangle. Therefore, there is an edge in C_3 with both endpoints in one of our partitions; so this partition does not make C_3 bipartite. Because this holds for every possible partition, we can conclude that no such partition exists – i.e. C_3 is not bipartite!

In general, we can say much more:

Proposition 1 *C_{2k+1} is not bipartite.*

Proof. We will prove this proposition with a proof by contradiction. In other words, we will assume that C_n is bipartite, and from there we'll deduce something we know to be false; from there, we can conclude that our assumption must not have been true in the first place (as it led us to something false,) and therefore that C_n is not bipartite.

To do this: as stated, we'll suppose for contradiction that C_{2k+1} is bipartite. Then, there must be some way of coloring the vertices $\{v_1, \dots v_{2k+1}\}$ of C_n red and blue, so that no edge is monochrome (i.e. has two red endpoints or two blue endpoints.)

How do we do this? Well: look at v_{k+1}. v_{k+1} has to be either red or blue: without any loss of generality[1], we can assume that it's red. Then, because no edge in C_1 is monochrome, we specifically know that none of v_{k+1}'s neighbors can be red: in other words, they both have to be blue! So both v_k and v_{k+2} are blue.

Similarly, we know that neither of v_k or v_{k+2}'s neighbors can be blue: so both v_{k-1} and v_{k+3} have to be red! Repeating this process, we can see that

- v_{k+1} being red forces
- v_k, v_{k+2} to be blue, which forces
- v_{k-1}, v_{k+3} to be red, which forces
- v_{k-2}, v_{k+4} to be blue, which forces
- ...
- which forces v_1, v_{2k+1} to both be the same color.

But there is an edge between v_1 and v_n in C_{2k+1}! This contradicts the definition of bipartite: therefore, we've reached a contradiction. Consequently, C_n cannot be bipartite.

This allows us to actually classify a large number of graphs as not being bipartite:

Proposition 2 *If a graph G has a subgraph isomorphic to C_{2k+1}, then G is not bipartite.*

[1] The phrase "without loss of generality" is something mathematicians are overly fond of. In general, it's used in situations where there is some sort of symmetry to the situation that allows you to assume that a certain situation holds: for example, in this use, we're assuming that v_1 is red because it has to be either red or blue, and if it was blue we could just switch the colors "red" and "blue" through the entire proof.

Proof. Suppose that G contains a subgraph H that's not bipartite. Then, for any coloring of H's vertices, there is some edge in H that's monochrome. Therefore, because any coloring of G's vertices into two parts will also color H's vertices, we know that any coloring of G's vertices with the colors red and blue will create a monochrome edge; therefore, G cannot be bipartite.

Is this it? Or are there other ways in which a graph can fail to be bipartite? Surprisingly, as it turns out, there isn't:

Proposition 3 *A graph G on n vertices is bipartite if and only if none of its subgraphs are isomorphic to an odd cycle.*

Proof. Our earlier proposition proved the "if" direction of this claim: i.e. if a graph is bipartite, it doesn't have any odd cycles as subgraphs. We focus now on the "only if" direction: i.e. given a graph that doesn't contain any odd cycles, we seek to show that it is bipartite.

First, note the following definitions:

Definition. A graph G is called **connected** iff for any two vertices $v, w \in V(G)$, there is a path connecting v and w.

Definition. Given a graph G, divide it into subgraphs $H_1, \ldots H_k$ such that each of the subgraphs H_i are connected, and for any two H_i, H_j's there aren't any edges with one endpoint in H_i and one endpoint in H_j. These parts H_i are called the **connected components** of G; a graph G is connected if it has only one connected component.

Definition. For a graph G and two vertices v, w we define the **distance** $d(v, w)$ between v and w as the number of edges of the smallest path connecting v and w. For a connected graph, this quantity is always defined, $d(v, v) = 0$, and $d(v, w) > 0$ for any $v \neq w$.

Take our graph G, and divide it into its connected components $H_1, \ldots H_k$. If we can find a red-blue coloring of each connected component H_i that shows it's bipartite, we can combine all of these colorings to get a coloring of all of G; because there are no edges between the connected components, this combined coloring would show that G itself is bipartite!

Therefore, it suffices to just show that any connected graph H on n vertices without any odd cycles in it is bipartite. To do this, take any vertex $y \in V(H)$, and construct the following sets:

- $N_0 = \{w : d(v, y) = 0\}$
- $N_1 = \{w : d(v, y) = 1\}$
- $N_2 = \{w : d(v, y) = 2\}$
- $\ldots$
- $N_n = \{w : d(v, y) = n\}$

First, notice that every vertex v shows up in at least one of these sets, as H is connected and has n vertices (and thus, any path in H has length $\leq n$.) Furthermore, no vertex shows up in more than one of these sets, because distance is well-defined. Finally, notice that for any $x \in N_k$ and any path P given by $y = v_0 e_{01} v_1 e_{12} \ldots e_{k-1,k} v_k = x$, each of the vertices v_j lies in N_j. This is because each of these has a path of length j from y to v_j (just take our path and cut it off at v_j), and has no shorter path (because if there was a shorter path, we could use it to get from y to x in less than k steps, and therefore $d(y, x)$ would not be k.)

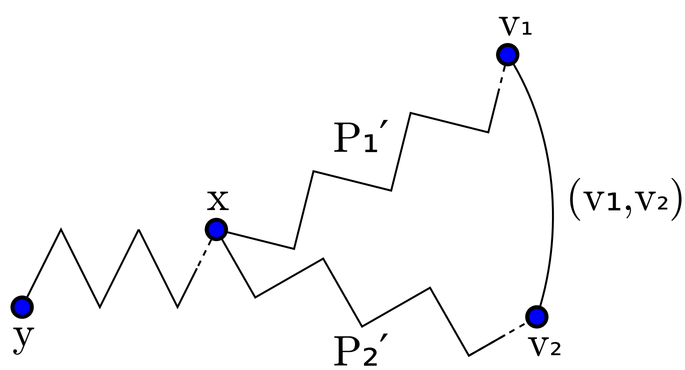
Now, color all of the vertices in the even N-sets red, and all of the vertices in the odd N-sets blue. We claim that there are no monochromatic edges.

To see this, take any edge $\{v_1, v_2\}$ in our graph H. Let $d(y, v_1) = k$ and $d(y, v_2) = l$, P_1 be a path of length k connecting v_1 with y, and P_2 be a path of length l connecting v_2 with y. These paths may intersect repeatedly, so take x to be the furthest-away vertex from y that's in both of these paths. Let P_1' be the path that we get by starting P_1 at x and proceeding to v_1, and P_2' be the path that we get by starting P_2 at x and proceeding to v_2.

There are two possiblities. Either x is one of v_1 or v_2, in which case (because there's an edge from v_1 to v_2) the distance from y to v_1 is either one greater or one less than the distance from y to v_2. In either case, v_1 and v_2 have different colors (because our colors alternated between red and blue as our distance increased,) so this edge is not monochrome.

Otherwise, x is neither v_1 or v_2. In this case, look at the cycle formed by doing the following:

- Start at x, and proceed along P_1'.
- Once we get to v_1, travel along the edge $\{v_1, v_2\}$.
- Now, go backwards along P_2' back to x.



This is a cycle, because P_1' and P_2' don't share any vertices in common apart from x. What is its length? Well, the length of P_1' is just $d(y, v_1) - d(y, x)$, the length of P_2' is just $d(y, v_2) - d(y, x)$, and the length of a single edge is just 1; so, the total length of this path is

$$d(y, v_1) - d(y, x) + d(y, v_2) - d(y, x) + 1 = d(y, v_1) + d(y, v_2) - (2d(y, x) + 1).$$

We know that this cannot be odd, because our graph has no odd cycles; so the number above is even! Because $(2d(y, x) + 1)$ is odd, this means that $d(y, v_1) + d(y, v_2)$ must also be odd; in other words, exactly one of $d(y, v_1), d(y, v_2)$ can be odd ,and exactly one can be even. But this means specifically that exactly one must be blue and one must be red (under our coloring scheme,) so our edge must not be monochromatic.

Therefore, our graph has no monochromatic edges; so it's bipartite!