1 Graph Theory

1.1 More on edge-counting

The number of edges incident on a vertex will be relevant for a number of reasons.

Definition 1. The *degree* of a vertex v in a graph G, denoted $d_G(v)$, is the number of edges in G which are incident upon v. The *minimum degree* in a graph G, denoted $\delta(G)$, is $\min_{v \in V(G)} d_G(v)$. The *maximum degree*, denoted $\Delta(G)$, is $\max_{v \in V(G)} d_G(v)$.

The subscripted G may be left out (and frequently is) if there is no ambiguity about which graph is being discussed. In older texts the degree may sometimes be called *valence*.

Definition 2. If every vertex in a graph G has the same degree, G is called *regular*. A graph in which each vertex has degree k is specifically called *k-regular*.

We can actually prove a cute theorem using graph degrees:

Proposition 1. *For any graph G, $2\|G\| = \sum_{v \in V(G)} d(v)$.*

Proof.

$$2\|G\| = \sum_{e \in E(G)} 2$$
$$2\|G\| = \sum_{e \in E(G)} \sum_{v \in V(G)} 1_{e \text{ incident on } v}$$
$$2\|G\| = \sum_{v \in V(G)} \sum_{e \in E(G)} 1_{e \text{ incident on } v}$$
$$2\|G\| = \sum_{v \in V(G)} |\{e \in E(G) : e \text{ incident on } v\}|$$
$$2\|G\| = \sum_{v \in V(G)} d(v)$$

□

Note that what we wrote above applies inconsistently to nonsimple graphs; it would work if we "double-counted" loops in totalling up the degree.

An interesting corollary:

Corollary 1. *If $|G|$ and k are odd, G cannot be k-regular.*

Proof. If G were k-regular, then $\sum_{v \in V(G)} d(v) = \sum_{v \in V(G)} k = |G|k$ would be odd. But $2\|G\|$ must be even. □

1.2 A Graph Library

We've dispensed with the varied labelings of graphs, so we can talk about them in terms of structure only — that is to say, henceforth, our references to "graphs" will be descriptive of entire isomorphism classes. There are a couple of particularly notable ones:

Definition 3. The *complete graph* K_n is a graph on n vertices in which every pair of distinct vertices is in the edge set.

In other words, the complete graph is the graph where every vertex is adjacent to every other vertex. It's fairly easy to see that $|K_n| = n$ and $\|K_n\| = \binom{n}{2}$; furthermore, K_n is $(n-1)$-regular.

Definition 4. The *empty graph* K_n^c is a graph on n vertices in which the edge-set is empty.

We'll see a justification for that superscripted "c" later. Clearly, $|K_n^c| = n$, $\|K_n^c\| = 0$, and K_n^c is 0-regular.

Definition 5. The *path* P_n is a graph on n vertices, denoted $v_1, v_2, \ldots, v_n$ here for clarity, with the edge-set $\{[v_1, v_2], [v_2, v_3], [v_3, v_4], \ldots, [v_{n-1}, v_n]\}$.

Note the special case $P_1 = K_1 = K_1^c$. It is also the case that $P_2 = K_2$. Furthermore, while identifying notable characteristics, it is worth mentioning that $|P_n| = n$ and $\|P_n\| = n - 1$. As pertains to degree, $\delta(P_n) = 1$ and $\Delta(P_n) = 2$.

Definition 6. For $n > 2$, the *cycle* C_n is a graph on n vertices, denoted $v_1, v_2, \ldots, v_n$ here for clarity, with the edge-set $\{[v_1, v_2], [v_2, v_3], [v_3, v_4], \ldots, [v_{n-1}, v_n], [v_n, v_1]\}$.

You are extremely unlikely to ever see a reference to C_1 or C_2, but if they were to be mentioned, C_1 would be the unique one-vertex graph, and C_2 the same as P_2 or K_2. Except for these (extremely unconventional) cases, it is generally true that $|C_n| = n$ and $\|C_n\| = n$, and that C_n is 2-regular.

There's one last interesting graph to be mentioned:

Definition 7. The *complete bipartite graph* $K_{a,b}$ is a graph on $a+b$ vertices, denoted $u_1, u_2, \ldots, u_a, v_1, v_2, v_3, \ldots, v_b$ here for clarity, with an edge-set consisting of every pair $[u_i, v_j]$ for $1 \leq i \leq a$ and $1 \leq j \leq b$.

Note that $|K_{a,b}| = a + b$ and $\|K_{a,b}\| = ab$. Also, $K_{a,b} = K_{b,a}$. $\delta(K_{a,b}) = \min(a, b)$ and $\Delta(K_{a,b}) = \max(a, b)$, and when $a = b$, $K_{a,b}$ is a-regular. Despite the name, a complete bipartite graph is not a complete graph (except in the trivial case of $K_{1,1} = K_2$). One complete bipartite graph frequently referred to by a different name is $K_{1,n}$, which is called a *star graph*.

1.3 Graph from other graphs: Subgraphs, complements, and other oddities

One simple way to twist around a graph is just to invert its edge-set: replace every non-edge with an edge and vice versa.

Definition 8. The *complement* of a graph G, denoted G^c, is given by the ordered pair $(V(G), E(G)^c)$, where $E(G)^c$ is a set consisting of all ordered pairs $[u, v]$ with $u, v \in V(G)$ such that $[u, v] \notin E(G)$.

This notation explains why K_n^c denotes the empty graph. Other simple observiations can be made trivially: $(G^c)^c = G$, $|G^c| = |G|$, and $\|G^c\| = \binom{|G|}{2} - \|G\|$.

Graph complements aren't very interesting right now, although they'll feed some problems we look at later. What is far more interesting is finding substructures of graphs:

Definition 9. A graph H is a *subgraph* of a graph G, denoted $H \subseteq G$, if there is an injective map from $V(H)$ to $V(G)$ mapping adjacent vertices of H to adjacent vertices of G. H is a *proper subgraph* of G if H is not isomorphic to G.

So, for instance, one can show that $P_n \subset C_n \subset K_n$, or that $C_{2n} \subset K_{n,n} \subset K_{2n}$. Colloquially, we might say that, for instance, "$K_{n,n}$ contains C_{2n}". On the other hand, $P_3 \subset K_{1,n}$, but $P_4 \not\subset K_{1,n}$. This concept lends itself to several interesting questions (which we shall not investigate in depth now, but which actually pertain to ongoing, hot research topics in graph theory). For any given graph G, we might ask:

- What is the largest n such that $P_n \subset G$?
- What is the largest n such that $C_n \subset G$?
- What is the smallest $n > 2$ such that $C_n \subset G$?
- What is the largest n such that $K_n \subset G$?

A stronger concept that the subgraph is the *induced* subgraph:

Definition 10. A graph H is an *induced subgraph* of a graph G if there is an injective map from $V(H)$ to $V(G)$ mapping adjacent vertices of H to adjacent vertices of G and non-adjacent vertices to non-adjacent vertices.

The induced subgraph of G *on vertices* $v_1, v_2, \ldots, v_k$, denoted $G[v_1, v_2, \ldots, v_k]$ is a graph whose vertex-set is $\{v_1, v_2, \ldots, v_k\}$ and whose edge-set consists of all edges in $E(G)$ whose endpoints are both elements of $\{v_1, v_2, \ldots, v_k\}$.

To illustrate the difference, we may observe that while P_n is a subgraph of C_n, it is *not* an induced subgraph (however, P_{n-1} is; using conventional vertex-labeling, $P_{n-1} = C_n[v_1, v_2, \ldots, v_{n-1}]$). Similarly, while the empty graphs K_r^c are all subgraphs of P_n for $r \leq n$, the restrictions on induced subgraphs are stricter: K_r^c is a subgraph of P_n only if $r \leq \lceil \frac{n}{2} \rceil$ (to see that this works, note that $K_r^c = P_n[1, 3, 5, \ldots, 2r-1]$).

Some special attention will be paid to C_n subgraphs. There are a few notable graph properties which are associated with C_n subgraphs:

Definition 11. A graph with no cycles as subgraphs is called an *acyclic graph* or a *forest*.

Definition 12. If G is a non-acyclic graph, then the *girth* of G is the smallest n such that C_n is a subgraph of G.

Definition 13. If $|G| = n$ and G has a subgraph isomorphic to C_n, then G is known as a *Hamiltonian* graph.

Justification for the names "forest" and "Hamiltonian" will come in time – we will look more closely at both classes of graph soon.

1.4 Graph traversals

The term "traversal" broadly describes a sequence of adjacent vertices. The most general form of a traversal is a walk:

Definition 14. A *walk of length k* on a graph G is a sequence $(v_0, e_1, v_1, e_2, v_2, e_3, \ldots, v_{k-1}, e_k, v_k)$ where each $v_i \in V(G)$, each $e_i \in E(G)$, and $e_i = [v_{i-1}, v_i]$. Conventionally, v_0 is called the *initial vertex* or *start vertex* of the walk; v_k is the *terminal vertex* or *stop vertex*. This walk may be described as being "from v_0 to v_k", or "between v_0 and v_k".

In practice, a walk is frequently specified just by the vertex-sequence $(v_1, v_2, v_3, \ldots, v_{k+1})$, since the edges are uniquely determined and thus redundant information.

There are several specialized sorts of walk. We can forbid self-intersection, either on edges or on vertices:

Definition 15. A *trail* on a graph G is a walk $(v_0, e_1, v_1, e_2, v_2, e_3, \ldots, v_{k-1}, e_k, v_k)$ such that all the e_i are distinct.

Definition 16. A *path* on a graph G is a walk $(v_0, e_1, v_1, e_2, v_2, e_3, \ldots, v_{k-1}, e_k, v_k)$ such that all the v_i are distinct.

Or we can mandate a particular kind of self-intersection:

Definition 17. A *closed walk* on a graph G is a walk $(v_0, e_1, v_1, e_2, v_2, e_3, \ldots, v_{k-1}, e_k, v_k)$ such that $v_0 = v_k$. A *tour* is a closed walk which is a trail. A *cycle* is a closed walk such that, except for $v_0 = v_k$, all vertices of the walk are distinct.

Note that this is not the first time we've encountered the words "path" and "cycle". There is in fact a simple correspondence: A path of length k on a graph G corresponds easily to a P_{k+1} subgraph of G, and a cycle of length k correspondes to a C_k subgraph.

We'll start with a simple observation about paths and trails:

Proposition 2. *Every path is a trail. Likewise, every cycle is a tour.*

Proof. On a graph G, consider a path $P = (v_0, e_1, v_1, e_2, v_2, e_3, \ldots, v_{k-1}, e_k, v_k)$. Suppose P is not a trail. Then some $e_i = e_j$, with $i < j$. So the unordered pair $[v_{i-1}, v_i] = [v_{j-1}, v_j]$. Thus, either $v_{i-1} = v_{j-1}$ and $v_i = v_j$, or $v_i = v_{j-1}$ and $v_j = v_{i-1}$. Since P is a path, two vertices are identical only if their indices are identical, so the above possibilities become that either $i-1 = j-1$ and $i = j$, or $i = j-1$ and $j = i-1$. The second is arithmetically impossible, and the first is contradicted by our presumption that $i < j$.

On a cycle, matters become slightly more complicated, since $v_i = v_j$ implies that either $i = j$ *or* i and j are 0 and k in some order. The above casewise analysis thus becomes a bit messies, requiring additional casework for the possibilities that, for instance, $i = 0$ and $j = k$, or $i - 1 = 0$ and $j = k$. These cases, however, are easily dispensed with. □

Now we move on to a more interesting question: when does a walk exist with a given start and stop vertex, and is it different than when a trail or path exists? Reassuringly, all three methods of traversal yield the same notion of ability ot connect two points.

Proposition 3. *If there is a walk on a graph G with initial vertex u and final vertex v, then there is also a path (and thus a trail) with initial vertex u and final vertex v.*

Proof. One of the easiest ways to prove this is by induction (the induction is not strictily necessary, but removes some legwork). We shall induct on the length of a walk W. Our assertion, quantified with a variable k, is that if G contains a walk of length k between points u and v, then it contains a path from u to v.

This statement is trivially true for $k = 0$ and $k = 1$, since a walk of length 0 is trivially a path with only one vertex, and a walk of length 1 consists of a single edge, which has distinct endpoints (note that we could, if we wish, add language here to deal with nonsimple graphs — this proposition is also true when G contains loops).

Now, for the inductive step, let us consider a walk $W = (v_0, v_1, \ldots, v_k)$ of length k. If W is itself a path, our assertion is clearly true. If, on the other hand, W is not a path, then there is some $i < j$ such that $v_i = v_j$. We may now construct a new walk $W' = (v_0, v_1, \ldots, v_i, v_{j+1}, v_{j+2}, \ldots, v_k)$. Clearly W' is shorter than W, since we have removed one or more vertices $v_{i+1}, \ldots, v_j$ from the sequence. In addition, W' is easily seen to still be a walk. Applying the inductive hypothesis to W', we shall find that there must be a path from v_0 to v_k. □

This proof can easily be thought of with a visual intuition: given a walk W, we can find all the self-intersections, and perform "surgery" to chop off the loop-shaped parts.

Since all three of our concepts of "connectability" are identical, we can be comfortable in producing a unified definition of the concept:

Definition 18. Vertices u and v of a graph G are *connected* if there is a path (or walk, or trail) from u to v.

Note that, despite similarity in common usage, "connected" and "adjacent" are *not* the same idea.

We shall find that connectedness is a property not simply of pairs, but of sets.

Theorem 1. *Connectedness on a graph G is an equivalence relation on the vertices.*

Proof. For any vertex u, (u) is a length-zero walk from u to u, so u is conenected to u, demonstrating reflexivity.

For any vertices u and v, if there is a walk from u to v, a walk from v to u results by reversing the walk, demonstrating symmetry.

If there is a walk from u to v, and from v to w, then splicing the walks end-to-end gives a walk from u to w (note that doing this with paths would be more complicated, since a splice of two paths would not necessarily be a path). This demonstrates transitivity. □

Because connectedness is an equivalence relation, we can collect the vertices of a graph into equivalence classes in which all points are mutually connected, called *components*.

Most of the time when we perform algorithms on graphs, or try to find other results, we'll be interested in the case where the graph is itself one big component.

Definition 19. If a graph G consists of a single component, or alternatively if every pair of vertices u, v in G is connected, then G is *connected*.

We'll end off this week with a cute little proof: either a graph or its complement is connected.

Proposition 4. *For any graph G, either G or G^c is connected.*

Proof. We perform induction on the number of vertices in G. Our base cases are quite trivial: if $|G| \leq 1$, then both G and G^c are easily seen to be connected.

Now, for the inductive step, let us consider a graph G, and choose an arbitrary vertex v of G; now let $H = G - v$; that is, the subgraph consisting of all vertices except v, and all edges except those incident on v. By the inductive hypothesis, either H is connected, or H^c is connected. We can address each of these cases separately.

Case I: H is connected. Hence, H lies entirely within a single component of G. Either v has degree zero in G, or v has nonzero degree in G. In the latter situation, v is adjacent to some vertex in H, so v lies in the same component of G as H, so G consists of a single component and is thus connected. On the other hand, if v has degree zero, then v is non-adjacent to every other vertex of G, so, in G^c, v is *adjacent* to every other vertex, and thus all vertices of G^c are connected: v is adjacent to every vertex u and thus connected to it; any other vertices u and w are connected via the path (u, v, w).

Case II: H^c is connected. We perform an extremely similar casewise analysis to the above G^c. Either v has degree zero in G^c, or v has nonzero degree in G^c. In the latter situation, v is adjacent to some vertex in H^c, so v lies in the same component of G^c as H^c, so G^c consists of a single component and is thus connected. On the other hand, if v has degree zero in G^c, then v is non-adjacent to every other vertex of G^c, so, in G, v is *adjacent* to every other vertex, and thus all vertices of G are connected: v is adjacent to every vertex u and thus connected to it; any other vertices u and w are connected via the path (u, v, w). □