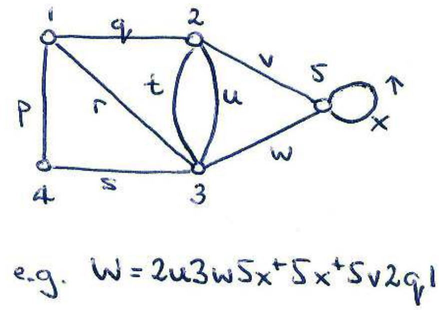
MOVING AROUND



- ∘ *walk* in G: alternating sequence of vertices and edges $W = v_0e_1v_1e_2v_2 \dots v_{\ell-1}e_\ell v_\ell$ where $\psi_G(e_i) = v_{i-1}v_i$ for each i. Direction in which a loop is used matters. (In simple graph can just write $W = v_0v_1v_2 \dots v_{\ell-1}v_\ell$.) *Length* is ℓ. *Initial vertex* v_0, *terminal* or * *final vertex* v_ℓ, *ends* v_0 and v_ℓ, *internal vertices* $v_1, v_2, \dots, v_{\ell-1}$.
- ∘ *reverse* of walk W: $W^{-1} = v_\ell e_\ell v_{\ell-1} \dots v_1e_1v_0$.
- ∘ *uv-walk* has initial vertex u, final vertex v.
- ∘ *closed* walk has initial vertex = final vertex.
- ∘ *trail* is walk with no repeated edges.
- ∘ *path* is walk with no repeated vertices. (So defines subgraph that is path graph.)
- ∘ *cycle* (sometimes circuit) is closed walk with no repeated vertices except that initial vertex = final vertex, at least one edge, and no repeated edges. (So defines subgraph that is cycle graph.) In simple graph write $(v_0v_1v_2 \dots v_{\ell-1})$.
- ∗ *reachability relation* R_G: uR_Gv or just uRv or *v is reachable from u (in G)* if there is a uv-walk in G. $R_G(u) = \{v \in V(G) \mid uR_Gv\}$, set of vertices reachable from u.

(M1) R_G is an equivalence relation. (R) trivial walk; (S) reverse walk; (T) concatenate two walks. Transitive closure of adjacency relation.

Recall: G is *connected* if for every partition of $V(G)$ into nonempty X and Y there is at least one edge from X to Y.

(M2) If G has a connected spanning subgraph, then G is connected. I.e., any supergraph of a connected graph with the same vertex set is connected.

Theorem M3: For a graph G, the following are equivalent.
(i) G is connected;
(ii) $\forall\, u, v \in V(G)$, uR_Gv (i.e., $\exists$ a uv-walk);
(iii) $\exists\, x \in V(G)$ such that $\forall\, v \in V(G)$, xR_Gv.

Proof: (i) ⇒ (ii): Suppose G is connected and let $u \in V(G)$. There is no edge xy with $x \in X = R_G(v)$ and $y \in Y = V(G) - X$, for we could extend a ux-walk with wy to get a uy-walk. Since $X \neq \emptyset$ ($u \in X$) and G is connected, we must have $Y = \emptyset$, i.e., there is a uv-walk for every $v \in V(G)$.

(ii) ⇒ (iii): Obvious.

(iii) ⇒ (i) Suppose that $\forall\, v \in V(G)$ there is an xv-walk. Assume (for a contradiction) that G is disconnected, with $V(G)$ partitioned into nonempty X and Y with no edge from X to Y. We may assume $x \in X$; choose $y \in Y$. By supposition there is an xy-walk $W = v_0e_1v_1 \dots v_\ell$ with $v_0 = x$, $v_\ell = y$. Let v_i be the last vertex of W that belongs to X. Then v_i exists because $x \in X$, $i < \ell$ because $y \notin X$, and $v_{i+1} \in Y$ by choice of v_i. But then e_{i+1} is an edge from $v_i \in X$ to $v_{i+1} \in Y$, a contradiction. Hence G is connected. ∎

Now we define components and make deductions about the component structure of a graph.

Recall: A *component* is a maximal connected subgraph. So every connected subgraph lies inside some component. Book defines components in an indirect way, hard to use for proofs.

(M4) The subgraph consisting of vertices and edges of a walk is connected, by M3(ii). In particular, any single edge with its ends is connected.

(M5) The union of two connected subgraphs is connected if and only if they have a common vertex. Suppose common vertex; by M3(ii) there is a walk from the common vertex to a vertex in either graph, but then by M3(iii) the union is connected. If no common vertex, vertex sets of subgraphs provide partition showing graph is disconnected.

(M6) If C is a component of G and v is a vertex of C, then C contains every edge of G incident with v. In particular, C is an induced subgraph. If some edge e incident with v is not in C, then C and e together form a larger connected subgraph, a contradiction.

(M7) Every connected subgraph of G is contained in a unique component of G. Contained in some component since connected; if in two then by (M5) their union is a larger connected subgraph. Therefore, the components of G partition the vertices of G and the edges of G. Thus, G is the disjoint union of its components.

(M8) The vertex sets of the components are the equivalence classes of R_G. If two vertices u, v are in the same component C, then uR_Cv (by M3(ii)) so uR_Gv. On the other hand, if uR_Gv, then the uv-walk W is connected and lies inside some component, so u, v belong to the same component.

So far we have been working with walks. Sometimes convenient to know we can get paths.

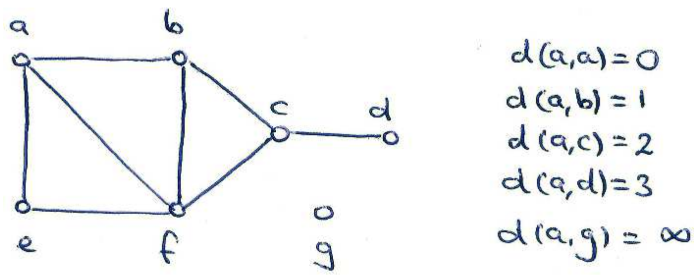
(M9) Any walk with repeated vertex x can be shortened by removing the segment between two occurrences of x. Applying this repeatedly, any uv-walk can be shortened to a uv-path. So if uR_Gv, i.e. a uv-walk exists, then a uv-path also exists. And a shortest uv-walk is a uv-path.

Can regard as algorithm: start with uv-walk, repeatedly remove redundant bits, get uv-path.

Corollary M10: Connectedness of G is also equivalent to
(iv) $\forall\ u, v \in V(G)$, $\exists$ a uv-path.

Often useful to look not just at existence of uv-walk or path, but how long it is.

The *distance* from u to v in G, $d_G(u, v)$, is the length of a shortest uv-path (or shortest uv-walk, by the proof of Lemma M9). Is ∞ if no uv-path. Convenient to be able to choose between walks or paths in proofs. Later see how to compute distance efficiently. Diameter, radius, etc. defined using distance.



Now relate cycles to bipartiteness. Book does later in different way in Section 4.2 (Theorem 4.7). Lemma first.

Lemma M11: A graph G has an odd (length) closed walk $\Leftrightarrow$ it has an odd cycle.

Proof: ($\Leftarrow$) An odd cycle *is* an odd closed walk.

($\Rightarrow$) Suppose G has an odd closed walk. Then there is a shortest odd length closed walk $W = v_0e_1v_1 \ldots v_\ell$. Assume (for a contradiction) that $v_i = v_j$ with $i < j$ and $(i, j) \neq (0, \ell)$. Then $W' = v_0e_1v_1 \ldots v_{i-1}e_i(v_i = v_j)e_{j+1}v_{j+1} \ldots v_\ell$ and $W'' = v_ie_{i+1}v_{i+1} \ldots v_j$ are closed walks, and since their lengths sum to the odd number ℓ, one of them is odd. But then we have a shorter odd closed walk, a contradiction. Hence W has no repeated vertices except $v_0 = v_\ell$, since W has odd length it has at least one edge and no repeated edges. Thus, W is an odd cycle. ∎

Again, can consider as algorithm to get odd cycle from odd closed walk, by applying repeatedly.

Not true for even walks and even cycles, e.g. K_2.

Allows us to connect bipartiteness to cycles.

Theorem M12: For a graph G the following are equivalent.
(i) G is bipartite.
(ii) G has no odd closed walks.
(iii) G has no odd cycles.

Proof: (ii) $\Leftrightarrow$ (iii) by Lemma M11.

(i) $\Rightarrow$ (ii): Suppose $G[X, Y]$ is bipartite. Every walk must alternate between vertices in X and vertices in Y, and so every closed walk must have even length.

(ii) $\Rightarrow$ (i): Suppose G has no odd closed walks. Let the components of G be $G_1, G_2, \ldots, G_c$. Choose $r_i \in V(G_i)$, and let

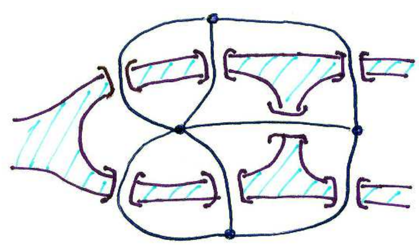
$X_i = \{x \in V(G_i) \mid d(r_i, x) \text{ is even}\}$ and $Y_i = V(G_i) - X_i = \{y \in V(G_i) \mid d(r_i, y) \text{ is odd}\}$.

For each $v \in V(G_i)$, let P_v be a $r_i v$-path of length $d(r_i, v)$. Suppose G_i has an edge e between u and v. Then $P_u e P_v^{-1}$ is a closed walk, so its length $d(r_i, u) + 1 + d(r_i, v)$ must be even, meaning that one of $d(r_i, u)$ and $d(r_i, v)$ is even and the other is odd. Thus, one of u and v belongs to X_i and the other to Y_i. Thus, (X_i, Y_i) is a bipartition of G_i. Hence, $(X_1 \cup X_2 \cup \ldots X_c, Y_1 \cup Y_2 \cup \ldots Y_c)$ is a bipartition of G. ∎

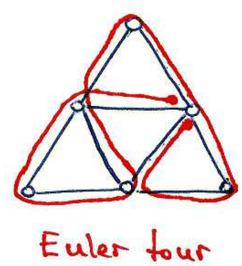
Two useful concepts associated with cycles:
∘ When G has at least one cycle:
girth = length of shortest cycle,
circumference = length of longest cycle.

Euler trails and tours

Euler, 1736: Königsberg Bridges Problem.
Start of graph theory.



∗ *euler trail* = trail using all **vertices and** edges of G (standard definition just says all edges, but then get into problems with isolated vertices),
∘ *euler tour* = closed euler trail,
∘ G is *eulerian* if it has an euler tour.



Theorem M13: A graph G is eulerian if and only if it is *even* (every vertex has even degree) and connected.

Proof: (Essentially the one in the first edition of West.)

($\Rightarrow$) Suppose there is an euler tour T. At each vertex T leaves and enters the same number of times, so there must be an even number of ends of edges at each vertex. G is connected by (M4).

($\Leftarrow$) Suppose G is even and connected. Let T be a longest closed trail in G, and let $R = E(G) - E(T)$. Assume (for a contradiction) that $R \neq \emptyset$.

First we claim that there is an edge of R incident with a vertex of T. If $V(G) = V(T)$ then any edge of R is suitable; otherwise, since G is connected, there is an edge joining $V(T)$ to $V(G) - V(T)$, which must belong to R.

So let $e \in R$ with end $x \in V(T)$. Construct a trail $S = xe \dots y$ in G using only edges of R that is maximal—it cannot be extended from y. If $y \neq x$, then since $d_G(y)$ is even, T uses an even number of ends of edges at y, and S uses an odd number of ends of edges at y, there is at least one unused edge-end at y that may be used to extend S, a contradiction. Therefore $S = xe \dots x$. Since $x \in V(T)$ we may splice S into T at x to get a longer closed trail, which is a contradiction.

Hence, $R = E(G) - E(T) = \emptyset$. Therefore, there are no edges between $V(T)$ and $V(G) - V(T)$, but G is connected, so $V(G) - V(T) = \emptyset$. Thus, T is an euler tour. ∎

Again, can transform proof into algorithm to construct an euler tour.

Book gives alternate proof based on different construction method, Fleury's algorithm, relies on idea of cutedge which we will look at soon.

Corollary: A graph has an euler trail if and only if it is connected and has at most two odd degree vertices.

Proof idea is to add edge between two odd degree vertices.