Exercise 17.3. Let $G = (V, E)$ be an undirected graph, and let $a, b, c \in V$ be three distinct vertices. We define an (a, b, c)-path as a path from a to c that goes through b. Consider the problem of deciding if there exists an (a, b, c)-path. For this problem, either (a) design and analyze a polynomial time algorithm (the faster the better), or (b) prove that the problem is NP-Hard.

Solution. This problem has a polynomial-time solution.

Abstractly, if we can find two simple paths in G—one from a to b and one from b to c—that are vertex-disjoint except for at vertex b, then we can concatenate these two paths to get an (a, b, c)-path.

To do this, we construct an auxiliary digraph G'.

First, to force vertex-disjointedness, we split each vertex $v \in V$ into two vertices, v_{in} and v_{out} and assign a directed edge from v_{in} to v_{out}. Then for each undirected edge $\{u, v\}$ in G, we add two directed edges (u_{out}, v_{in}) and (v_{out}, u_{in}). Finally, we assign b_{out} to be the source vertex and direct both a_{out} and c_{out} to a single destination vertex t.

Now, finding two edge-disjoint paths from b_{out} to t in G' will guarantee vertex-disjoint paths from b to a and b to c in G. The former path can be reversed and concatenated to obtain an (a, b, c)-path.

abc-path(G, a, b, c):

/* *given $a, b, c \in G$ in an undirected graph, computes whether an (a, b, c)-path exists in G.* */

1. Construct the auxiliary digraph G':
   - A. For every vertex $v \in V$, create vertices v_{in} and v_{out} and add the edge (v_{in}, v_{out}).
   - B. For every undirected edge $\{u, v\} \in E$, add the edges (u_{out}, v_{in}) and (v_{out}, u_{in}).
   - C. Let $s \leftarrow b$.
   - D. Create a sink vertex t, and add edges from a_{out} and c_{out} to t.
2. If augmenting-paths(G', s, t) $= 2$ then
   - A. Return `true`.
3. Return `false`.

Runtime. We use augmenting-paths(G', s, t) to compute the maximal number of edge-disjoint (s, t)-paths. Note that we can slightly modify augmenting-paths() to return true immediately when we find 2 such paths so that the runtime is in $O(m)$. ■

Correctness. We now prove that an (a, b, c)-path exists in G if and only if augmenting-paths(G', s, t) $= 2$.

First, suppose there are two edge-disjoint paths from b_{out} to t in G'. Since the only incoming vertices to t are a_out and c_out, one path translates to a path that goes to vertex a (a_{in}, a_{out}) and other, a path that goes to vertex c (c_{in}, c_{out}) from b. Vertex-splitting guarantees that these two paths are vertex-disjoint in G; hence, there exist vertex-disjoint paths from b to a and b to c. The path from b to a can be reversed, since G is undirected, and prepended to the path from b to c to get an (a, b, c)-path.

Conversely, suppose G has an (a, b, c)-path. This path can be split in to two vertex-and-edge-disjoint paths; one from b to a and the other, from b to c. It follows that these two paths can be extended to two edge-disjointed paths from b_{out} to t in G'. Since the maximum path-packing in G' clearly has size 2, augmenting-paths(G', b_{out}, t) will return 2. ■

□