**Solutions:**

**1.** The checksums can be found at <http://releases.ubuntu.com/18.04.1/MD5SUMS>, <http://releases.ubuntu.com/18.04.1/SHA1SUMS>, <http://releases.ubuntu.com/18.04.1/SHA256SUMS>.

**2.** Run sha1sum on all the files.

[ 0 13:11:31 christian .../Downloads/crimeandpunishment ] $ sha1sum \*

0a5f42ac180eb34ccc495bbe47871cb91756b754 10.txt

3d7e6942c59211bffb554ad7128bc000e5c81223 1.txt

3d7e6942c59211bffb554ad7128bc000e5c81223 2.txt

0a5f42ac180eb34ccc495bbe47871cb91756b754 3.txt

f11fff486608f195b46a32b31fd4d4bde153dbaa 4.txt

4d4dde976442a689a3515f62cb58a981d7bacc4c 5.txt

6c287645a7f151d89c3e577c60542be35b7e9af8 6.txt

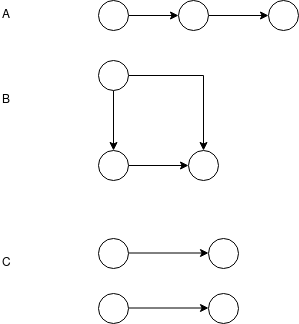
6c287645a7f151d89c3e577c60542be35b7e9af8 7.txt

0a5f42ac180eb34ccc495bbe47871cb91756b754 8.txt

f11fff486608f195b46a32b31fd4d4bde153dbaa 9.txt

1 = 2, 3 = 8 = 10, 4 = 9, 5 is unique, 6 = 7.

**3.** Many options, like for example



Note, the last one is not connected (it has two separate components.) That is OK.

4. The argument (made semi-formal) is:

Lemma: We can always find a node in a DAG with k nodes which has no ancestors. Assume we cannot. Then we can construct a path x[1] -> x[2] -> ... x[k+1] where x[i] is a node and x[i] and x[i-1] are connected by an edge. Since there are k+1 nodes in this sequence but only k nodes in the graph, two of those nodes must be identical. That would violate acyclicity, and therefore cannot be possible.

Let G be a DAG on n nodes. Let G[0] = G, and L[0] = {} an empty list. We construct G[i] and L[i] at step i of the process.

For i = 1 .. n-1:

Step i: pick a node from G[i-1] which has no ancestors. Possible by lemma. Let G[i] be the graph obtained by removing the node and all incident edges from G[i-1]. Create L[i] by appending the selected node to L[i-1]. If L[i-1] was topologically sorted, L[i] remains so, since no node in G[i] is an ancestor of any node in L[i].

This process exhausts G, and L[n-1] is a list of all nodes sorted topologically.

5. One possible order: F E D C B G A.

6. Open-ended. The git-command brings up the commit graph. They're asked to identify the root commit and merge commits.