

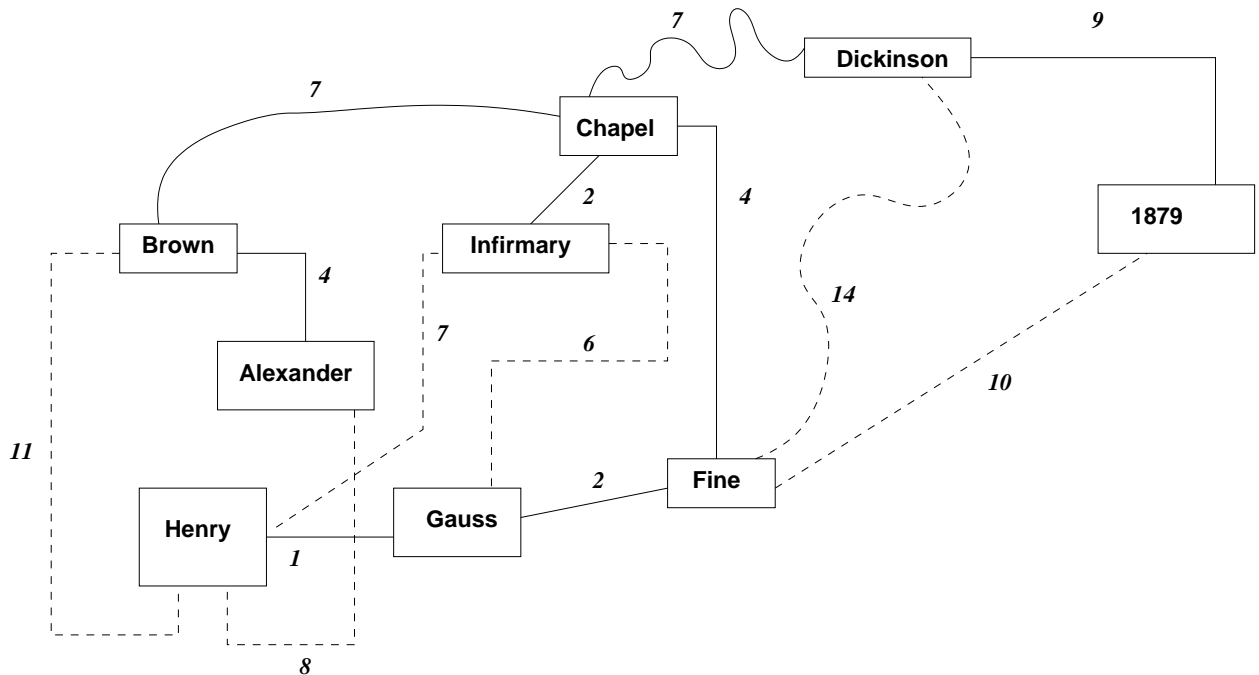
**Due:** Wed., 24 November 2004 at midnight

Create a directory to hold your answers to this homework set. Copy the files from `$master/hw/hw10` into this directory. Put non-program answers into a file `hw10`. Use the command `submit hw10` to submit your solutions to the problems below.

1. [From the 1992 Berkeley Programming Contest] Exercise 12.1 from *DSIJ* (however, look in directory `~cs61b/hw/hw10`, not `hw7`).

2. [From the 1996 Berkeley Programming Contest] Wolverine College, in Michigan's Upper Peninsula, is often covered with snow. Its grounds crew tends to get somnolent in the winter, however, and tries to do as little work as possible in clearing the sidewalks. Specifically, once they know what buildings will be used on any given day (the faculty isn't so energetic either), they clear off the shortest combination of sidewalks possible that leave some (possibly long) path connecting all the open buildings. Your task here is to write a program that determines this set of sidewalks. Use the template file `~cs61b/hw/hw10/Shovel.java`.

For example, given the diagrammatic map below, the solid lines represent a minimum-length set of clear sidewalks, the dashed lines represent the snow-covered walks, and the numbers are lengths of the walks.



Each set of input to your program will consist of a sequence of *sidewalk descriptions*, all in free form, followed by an isolated semicolon (i.e., surrounded by whitespace). A sidewalk description consists of two building names and a distance (a positive integer). Building names consist of up to

128 letters, digits, and underscores—no blanks or punctuation. You may assume that all buildings are connected by some path to all other buildings (so that, in particular, all building names are included in at least one sidewalk description). You may make no assumptions about the number of buildings or sidewalks. You may assume that no walk will be longer than  $10^9$  long. You may also assume that there will be a unique solution.

For each set of input, the output is to have the form illustrated in the sample input below. Output the sidewalks in the same order as their descriptions were input.

Input	Output
Brown Henry 11	Set 0:
Henry Alexander 8	Clear Gauss to Henry
Gauss Henry 1	Clear Brown to Chapel
Infirmary Henry 7	Clear Alexander to Brown
Brown Chapel 7	Clear Infirmary to Chapel
Alexander Brown 4	Clear Gauss to Fine
Infirmary	Clear Dickinson to 1879
Chapel	Clear Chapel to Dickinson
2	Clear Chapel to Fine
Gauss Infirmary 6 Gauss Fine 2	
Fine Dickinson 14	Set 1:
1879 Fine 10	Clear Alexander to Brown
Dickinson 1879 9	Clear Henry to Alexander
Chapel Dickinson 7	
Chapel Fine 4 ;	
Alexander Brown 4	
Brown Henry 11	
Henry Alexander 8	
;	