Assignment

Problems identified by \( x.y(z) \) denote the problem “\( y \)”, in chapter “\( x \)” of the textbook, with part “\( z \)”. If “\( z \)” is
not noted, then the entire problem is required.

Assignment 4a, Due Feb 21

• 4.22 (2 points) Read this problem, and write down your ideas and questions about how to turn it into a
  graph path problem, such that our shortest path algorithms can be modified to solve it.
• (5 points) Implement the binary heap from Figure 4.16 of the text book. Test it for correctness. Bring a
  printed copy of your code for this submission.

Assignment 4b, Due Feb 23

• 4.1(a) (2 points) Run Dijkstra, tracking the problem data in a table.
• 4.12 (2 points) Your algorithm should be \( O(|V|^2) \) or better.
• (5 points) Create a runtime measuring program for the binary heap. Include the ability to measure
  performance of individual methods as a function of the number of items in the heap. Use powers of 2 for
  the sizes. Bring a printed copy of your code for this submission.

Assignment 4c, Due Feb 26

• 4.1(b) (2 points) Run Dijkstra, show shortest-path tree.
• 4.14 (2 points) By efficient, we mean no worse than Dijkstra’s algorithm.
• (5 points) Measure the runtime of the \( \text{makeheap()} \), \( \text{deletemin()} \), \( \text{insert()} \) and \( \text{decreasekey()} \) methods.
  For powers of 2 from \( 2^4 \) to at least \( 2^{28} \). Record the results in a spreadsheet. Bring a printed copy of
  your table for this submission.

Assignment 4d, Due Feb 28

• 4.2(a) (2 points) Run Bellman-Ford, tracking the problem data in a table as we did in class. Each
  iteration is a new array, based on the previous array. Start from node S.
• 4.2(b) (2 points) Draw the shortest-path tree, using your table data.
• (5 points) Add theoretical functions to your spreadsheet of results, include at least \( \log n \), \( n \), \( n \log n \) and
  \( n^2 \). Produce a table of these values normalized the \( \text{deletemin()} \) column at size \( 2^{20} \). Chart this table.
  Bring a printed copy of your table and your chart for this submission.

Assignment 4e, Due Mar 2

• 4.8 (2 points) Prove = proof, disprove = counter-example
• 4.5 (2 points)

Assignment 4f, Due Mar 5

• 4.11 (2 points) How can you find cycles using path algorithms in this chapter?
• 4.15 (2 points)
• 4.19 (2 points) Look for a modified version of Dijkstra that meets the criteria.

Assignment 4z, Due Never (optional)

• Other problems from the chapter

Submission

• For the written work, at the beginning of class, on the due dates, submit paper copies of your solutions.
• For the experimental determination, at the beginning of class on the due date, submit paper copies of
  the graphs.