Algorithms

CMPT 435

– Assignment 4 - 75 points

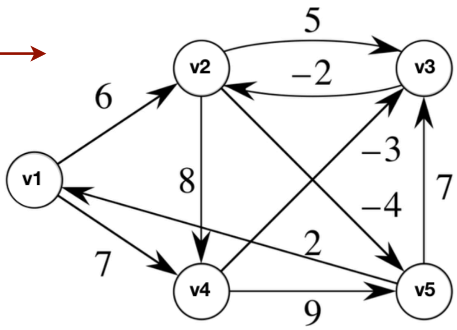
Goals

to implement the Bellman-Ford dynamic programming algorithm for Single Source Shortest Path (SSSP) on a few weighted, directed graphs; to implement a greedy solution to an intergalactic instance of the fractional knapsack problem; and to analyze these algorithms' performance in asymptotic terms.

Requirements and Notes

(1) Modify your graph implementation from assignment #4 to model directed and weighted graphs. The file graphs2.txt contains data describing multiple directed and weighted graphs. Read it and create a **linked object** representation for each graph. Then, for each graph, run SSSP (with vertex #1 as the single source) and output the results. [15 points]

Example graph →



Example output:

```
1 → 2 cost is 2; path is 1 → 4 → 3 → 2.
1 → 3 cost is 4; path is 1 → 4 → 3.
1 → 4 cost is 7; path is 1 → 4.
1 → 5 cost is -2; path: 1 → 4 → 3 → 2 → 5.
```

(2) Imagine you are traveling to Arrakis for a spice heist. You must fill a few knapsacks with as many scoops of the most valuable spice as they will hold. The file spice.txt contains the details of available spice and knapsacks you can use. Implement the fractional knapsack algorithm to maximize your take. [10 points]

Example output:

```
Knapsack of capacity 1 is worth 9 quatloos and contains 1 scoop of orange.
Knapsack of capacity 6 is worth 38 quatloos and contains 2 scoops of orange, 4 scoops of
blue.
```

(3) In your LaTeX analysis document, **explain the asymptotic running time** of both SSSP and fractional knapsack and why it is that way. And don't forget your code listings and explanations. [50 points]

As ever, your code must separate structure from presentation, be professionally formatted yet uniquely yours (keep showing personality), use and demonstrate best practices, and make me proud to be your teacher. [−∞ if not]

Note: I will run your code on different versions of the text files in the same format. In other words, there will be more or different graphs or items, but the structure of the files will be the same as those referenced above. Your code must react to the contents of the file; you cannot hard-code any aspect.

Resources

- Dynamic programming is described in the 3rd edition of our text in section 15.
- The greedy strategy is described in the 3rd edition of our text in section 16.2.

Submitting Your Work

In addition to your source code, commit your LaTeX documents to your GitHub repository. Make **many** commits to GitHub. Be sure that you make your final commit for this assignment on or before the due date. (See our syllabus for those details.)