

hw3

Due Feb 5, 2016 by 5pm	Points 10	Submitting a file upload	File Types zip
-------------------------------	------------------	---------------------------------	-----------------------

Due: 5 PM on Friday, Feb. 5.

You should complete this homework with your chosen partner. Information on how to register your partnership is TBD. (I believe there's a way to do it on Canvas, but I need to figure out how it works.)

For this homework, you will implement a graph data structure and graph search algorithm. The purpose is for you to gain experience implementing data structures in Rust.

The deliverable

The purpose of graph is to find paths in graphs. It reads a graph specification from a file, and then answers routing queries typed by the user.

A graph specification file represents an undirected graph as an association list of nodes, written as tokens. In particular, each line is a list of words, where the first word names some node in the graph and the remaining words enumerate its neighbors. Every node mentioned as a neighbor must start a line as well, and no node may start more than one line.

The user enters queries on stdin, one at a time. A query consists of two node names, a starting node and an ending node. The program then prints out a path between the nodes, or a message that no such path exists.

Here is a simple example:

```
$ cat graph.dat
```

```
a b d
```

```
b a d
```

```
c
```

```
d c
```

```
$ cargo run graph.dat
```

```
-> a d
```

```
a b d
```

```
-> a b
```

```
a b
```

```
-> a c
```

```
a b d c
```

(User input appears in ***bold italics***.)

List any assumptions that you need to make where this specification is incomplete, and be sure to test thoroughly.

Evaluation

Your grade will be based on:

- correctness (how closely your program adheres to this specification),
- thoroughness of testing,
- completeness of documentation (including assumptions),
- style (not expecting the most idiomatic Rust at this point, but I'll be looking for good factoring—don't put everything in main), and
- efficiency (no need to benchmark or profile, but do choose sensible data structures and avoid needless copying).

How to submit

3/28/2017

hw3

Please submit a tar or zip archive of your Cargo project directory here on Canvas. Name your archive `NW1114-hw3.{zip,tgz}` (e.g., `jat489-hw3.tgz`).