In an effort to make sure that my program passes I've come up with some large randomly generated graphs. The first three graphs have a couple thousand vertices, the last one should have approx 10,000 vertices.

Extract and copy the files into your directory.

Test with:

```bash
./dijkstra large1.txt > myOut1.txt
./dijkstra large2.txt > myOut2.txt
./dijkstra large3.txt > myOut3.txt
./dijkstra large4.txt > myOut4.txt
diff testOut1.txt myOut1.txt
diff testOut2.txt myOut2.txt
diff testOut3.txt myOut3.txt
diff testOut4.txt myOut4.txt
```

File Attachments
1) LargeTests.zip, downloaded 29 times

Subject: Re: Large Tests (over 9000 vertices)
Posted by SSinischo on Fri, 31 Mar 2017 01:19:54 GMT

I am getting a lot of different output. It looks like you are not updating when you find a path with fewer steps but the same distance. Also there are some issues separating trees. Did you match all of my output in my other thread?

The input is so large my VM freezes up just trying to search for values.

-

Subject: Re: Large Tests (over 9000 vertices)
Posted by georgecoll on Fri, 31 Mar 2017 01:27:40 GMT

I'm matching all of these and matching all of yours @SSinischo
Subject: Re: Large Tests (over 9000 vertices)
Posted by SSinischo on Fri, 31 Mar 2017 01:32:08 GMT

here is my output for test 1 compared to yours:

https://www.diffchecker.com/tnqxUUCC

You can see that some nodes have the same distance but have paths that yield different levels/steps

this behavior is confirmed proper by jrobinson3:
@SSinischo aren't you doing the opposite that Dr. Lusth confirms in that thread then, by having the higher value as the predecessor?

IF that wasn't the case it would fail the test3 dropbox for graph50 right? The graph50 has 43 as the parent of 41, rather than 3.

Subject: Re: Large Tests (over 9000 vertices)
Posted by ncgleason on Fri, 31 Mar 2017 02:44:16 GMT
View Forum Message <> Reply to Message

In the graph50 43 is the parent because once the predecessor is set to 43, it will never be reset to 3 (which comes later). That's how I interpreted what Lusth said. Am I correct?

Subject: Re: Large Tests (over 9000 vertices)
Posted by nboltralik on Fri, 31 Mar 2017 03:14:28 GMT
View Forum Message <> Reply to Message

@SSinischo
It looks like you are the only one who has code that updates based on a route with the same weight but less steps.
My code passes test_50 in the dropbox so it doesn't appear that this kind of route update is necessary.

Also, in the thread you linked, Dr. Lusth confirms that the the tiebreaker for adding vertices to the tree is simply the one with the smaller vertex number. He also says that once the predecessor is set it in not changed unless a shorter path is found which I assume means smaller weight not less steps?

Subject: Re: Large Tests (over 9000 vertices)
Posted by SSinischo on Fri, 31 Mar 2017 03:17:36 GMT
View Forum Message <> Reply to Message

My code passes all tests as well. In the thread I linked, the TA says we should prioritize steps when weights are equal.

What happens if there's a source with a single edge to a vertex x with edge weight 100, vs. a 100 vertex path to x all with edge weights 1? Both update x's distance to 100, but clearly one path is "better". I think some input from Lusth is needed to be sure...

Edit: so, upon changing two lines of code, my results match. But still, I am unclear as to which one of us has the correct results.

-
From that thread
Quote:
It seems I get the same distances but a different path for node 39.

For instance to get to node 39 you do: 0->45->25->11->39, but I do 0->27->34->39

I think something is wrong with your bfs (unless there is something I missed in the assignment).

@jarobinson only says that the paths are different. It looks to me like the 0->27->34->39 path is correct because its parent value is smaller, not because there are fewer steps.

When you have an edge of length x and process another edges of length x you don't ever replace it is what I'm getting from Dr. Lusth's comment in the other thread.

nbolralik wrote on Thu, 30 March 2017 22:32
From that thread
Quote:
It seems I get the same distances but a different path for node 39.

For instance to get to node 39 you do: 0->45->25->11->39, but I do 0->27->34->39

I think something is wrong with your bfs (unless there is something I missed in the assignment).

@jarobinson only says that the paths are different. It looks to me like the 0->27->34->39 path is correct because its parent value is smaller, not because there are fewer steps.

The parent value (11) in my original path is smaller than what's in his path (34). I think what Lusth confirms is that parent value doesn't matter. But do number of steps? This is not defined in the specs - we desperately need his input on this.

Edit: the output in this thread is correct. Below is an email sent from Dr. Lusth.

Quote:According to dijkstra's algorithm in the book, the only time a predecessor changes is when the key value can be made smaller. The only tie breaker in the spec concerning the algorithm is when there are competing values (all with the same key) vying to become the extreme value in
the priority queue. The tie breaker in that case is the vertex number.

Does this help?

dr.j

Subject: Re: Large Tests (over 9000 vertices)
Posted by lusth on Fri, 31 Mar 2017 14:32:55 GMT
View Forum Message <> Reply to Message

nboltralik wrote on Thu, 30 March 2017 22:14@SSinischo
It looks like you are the only one who has code that updates based on a route with the same weight but less steps.
My code passes test_50 in the dropbox so it doesn't appear that this kind of route update is necessary.

Also, in the thread you linked, Dr. Lusth confirms that the the tiebreaker for adding vertices to the tree is simply the one with the smaller vertex number. He also says that once the predecessor is set it in not changed unless a shorter path is found which I assume means smaller weight not less steps?

Yes, shorter path means less weight. In the context of Dijkstra's algorithm, the number of vertices along a path is irrelevant.

Subject: Re: Large Tests (over 9000 vertices)
Posted by ngleason on Fri, 31 Mar 2017 17:09:10 GMT
View Forum Message <> Reply to Message

For clarification, the output of the 4 large tests provided in the ZIP is correct?

Subject: Re: Large Tests (over 9000 vertices)
Posted by SSinischo on Fri, 31 Mar 2017 23:00:14 GMT
View Forum Message <> Reply to Message

ngleason wrote on Fri, 31 March 2017 12:09For clarification, the output of the 4 large tests provided in the ZIP is correct?
yes.

Subject: Re: Large Tests (over 9000 vertices)
Posted by jarobinson3 on Sat, 01 Apr 2017 15:44:51 GMT
View Forum Message <> Reply to Message
Okay, so which is the correct output? testOut3.txt or myOut3.txt?

When I extract the files (don’t run anything) and do

```
diff myOut1.txt testOut1.txt
```

The two files are marked as different.

---

Subject: Re: Large Tests (over 9000 vertices)
Posted by nboltralik on Sat, 01 Apr 2017 15:59:26 GMT
View Forum Message <> Reply to Message

The files named testOutX.txt are what should be the correct output, the myOutX.txt files are blank in the zip file so you'll need to run dijkstra and output to them before trying diff.

---

Subject: Re: Large Tests (over 9000 vertices)
Posted by jarobinson3 on Sat, 01 Apr 2017 16:11:51 GMT
View Forum Message <> Reply to Message

nboltralik wrote on Sat, 01 April 2017 10:59:
The files named testOutX.txt are what should be the correct output, the myOutX.txt files are blank in the zip file so you'll need to run dijkstra and output to them before trying diff.

I see that now.

All my outputs match yours.