Unfortunately, there are many things here and there that are vague when reading directly from the .pdf, so I have come up with this full checklist of things based on the items that were clarified on Piazza. Each checklist item will have a link to their respective Piazza post (or some other indicator if it's explained in the .pdf).

Personally, I think any item that had to be clarified on Piazza should not be subject to the same strict "no re-evaluation" @140 rule announced if that clarification was only provided less than a day or two prior to the deadline. Thankfully, it does not appear that this is the case (yet), but hopefully, TA's can understand that we need time to be able to debug and fix our code.

Last week, for example, I noticed things were going back and forth regarding the correct way to handle when to terminate our program (the .pdf said one thing and then Piazza posts were saying another). The more confused the students are, the more Piazza posts we have to scroll through, the more work TA's have to do to clarify; it's an endless cycle that does not benefit anyone.

I would greatly appreciate it if TA's (or even other students if possible) could actively update/edit this checklist as well as we near the submission deadline, so that we all have one place where we can see and double check everything, rather than having to scroll through dozens of Piazza posts (as I just did to create this checklist in the first place):

LAST UPDATE: 06:00 EST on 16/10/23

- [ ] "V E s" is given but "s" is rejected, continue reading until an accepted "s" is received. **"V" and "E" may only be redefined after each successful "s" command**. @159
- [ ] A call sequence of "V E s s" is valid and should output accordingly. @158
- [ ] A call sequence such as "V E E" should throw an error on the second 'E' call. And according to @159, we would continue reading in this case and V E would only be redefined after a successful 's' call.
- [ ] Allow/disallow negative vertex IDs. @pdf @157
- [ ] Remove warnings from code even if it is working. @156
- [ ] "V E E E E s s s s s" should only record the first non-erroneous "V" and the first non-erroneous "E". @154

In combination with @159, if only the last 's' in this sequence is valid, then you would throw errors and use the first valid 'V' and first valid 'E' for that first valid 's' call's output.

- [ ] use stderr or stdout for printing error messages. @pdf @154
- [ ] "E V s" is not ok and should not run. @153

I don't think the instructor answer "No" was clear enough here, given that the original post had several questions combined into one. I am making a bit of an assumption with this checklist item.

Calling 'E' as the first command should immediately output an error. If it's "E V E s", then you would have an error, followed by a valid "V E s" sequence that should output accordingly.

- ☐ Go over the programming style guidelines. @151
- ☐ Multiple shortest paths may exist. @pdf @120 @150
- ☐ Ignore empty line inputs, but EOF termination should still work. @148
- ☐ A single invalid Edge or Vertex in an 'E' command should invalidate the whole command. @133 @146
- ☐ Decimals will not be tested, also, the .pdf states integer for vertex IDs. @pdf @146
- ☐ V 0 input should throw an error. @pdf @146
- ☐ V 1 **(I am still double-checking if this is valid or not and how we should handle it. Not sure if I saw a post on Piazza that specifically mentioned it. Will update accordingly.)**
- ☐ 'V', 'E' or 's' command with no arguments should be an error, but will not be tested @146
- ☐ "5. Since it is an undirected graph, there is no question of a vertex mapping to itself as an edge." @ 146

I took this to mean that <1, 1> is not a valid edge. Please correct me if I am wrong.

Post @177 also seems to confirm this assumption, saying we won't be tested for this case.

- ☐ There won't be "V V E". "V E E" can occur, but your program must ignore subsequent E specifications after the first E. That is to say, pick the first E and then look for the first 's' thereafter. What occurs in between is irrelevant. @145 @154 @159
- ☐ Program terminates at EOF, not when an error is thrown, but do print the errors. @137 @141 @145
- ☐ Duplicate edges will not be tested for. <2,8> and <8,2> being a duplicate will also not be tested. We are free to account for both these scenarios and treat them however we want. @143
- ☐ "s 1 1" is valid and should output "1" for shortest path, not "1-1". A properly implemented algorithm will cover this naturally. @139
- ☐ An isolated vertex should still output the shortest path if valid, e.g. for the isolated vertex 7, "s 7 7" should output "7". A properly implemented algorithm will cover this naturally. @139
- ☐ Make sure code runs on eceubuntu servers. This will also confirm what STL/other imports are acceptable. @135

If you were having issues on eceterm, try eceubuntu (and use a VPN if off-campus)

- [ ] int is enough for number of vertices defined by V. @134
- [ ] Make sure your program can take .in file and output a .out file @122

All in all, I hope this checklist helps. I also think the TA's should start a master checklist similar to this that is updated every time some clarification on Piazza is made. If we have one trusted place that we can reference at any moment (knowing it's up-to-date), it would greatly help with future assignments.

Disclaimer: This checklist, derived from questions on Piazza, may not capture all potential uncertainties or clarifications - although it tries to. Always exercise due diligence by referring to the official assignment .pdf and checking Piazza regularly.

Download a .pdf or .md version of this checklist: https://github.com/adaniel2/ECE650-Checklists