**Student feedback #1**

This is a really good idea! I like how you’re exploring a problem that is related to how the computer works, as a computer science major it’s nice to really see how this can all tie together. Here are my thoughts on the questions you asked about for feedback:

**How is our model complexity? We are planning to go with testrun.py as it seems more complex than the alternate version, but we are still wondering if it suffices in complexity**

The problem you are solving seems to be complex but I wonder if there are other constraints you can add? The ones you have right now are quite similar to each other. It may be a good idea to add one or more constraints/propositions into the mix to add more variability to the model.

**If testrun.py is not complex enough, what do you think of the idea of modelling deadlock?**

It sounds like an interesting idea, but I don’t feel like you need to restart your whole project. I wonder if you could use the concept of deadlock to make your existing model more complex? Maybe you can add a new constraint that will check if deadlock will occur (if so the model will be false, if not the model is true).

**Not sure what should be included in the Jape Proofs file. Are we trying to prove that our constraints are valid or that the solutions provided are valid? If we were trying to prove that how would we go about it?**

The purpose of the jape proofs is to make conclusions from the constraints that have not already been modelled (just to show your knowledge about solving proofs and natural deduction). My group is feeling a bit uncertain in this area as well, but we asked a few questions in the discussion forums and found that you can simplify your constraints quite a bit, and the proofs do not need to be related to your code. Here are some threads in the discussion forum that might help to read:

* https://onq.queensu.ca/d2l/le/398978/discussions/threads/3981269/View
* https://onq.queensu.ca/d2l/le/398978/discussions/threads/3982571/View
* The Week 8 video also had helpful information, with an example proof

Maybe you could try to prove something along the lines of: “If one resource is being used in a critical section, this does not imply another resource is also being used by that section”?

**Is this all valid exploration for our model? I know there are examples provided and we are yet to test our partial assignments thoroughly, just wanted to know if this is all valid content.**

To my understanding, this is valid. There are lots of different versions at play in the document that was shared which I found a bit confusing. I would suggest eliminating past work in your documentation and focusing more on what you have decided on or plan to do.

One other thing to consider would be how you plan to get values for each of your propositions. Our group asked about this - you could read in a file with input values, or get user input.

**How would we properly include code in the document, would you prefer a line reference to files on the repository e.g. testrun.py: line 10 - 24**

Line references would be good from my point of view! I think small screenshots to code you are referencing would work too. You’ve also included lots of comments in your code, which was helpful for understanding - I might shorten this in your final submission though

**Student feedback #2**

I think overall it’s good but it’s a little bit complex for me to read. However overall the idea is great and the theory is flawless. I would suggest to simplify your code part considering you’re going to have a lot more complex testing code.

About the jape proof I didn’t see your jape file so I assume that you’re not starting it yet. Maybe you can try to prove the constraint and your solutions are valid just like what you write in your draft.

**Student feedback #3 –This one is sometimes unnecessarily harsh. Don't take it personally**

First of all, when I started to check what goal your modelling project aims to achieve, I couldn’t understand it well. Through your project summary and description, a few lines are not enough to make people understand you. What do you mean by “the two processes” and “shared resources”? You must stand in the perspective of readers instead of yours when you write an explanation. I suggest your rewrite your description and describe all the steps you will go through in this project. However, your explanations of propositions and constraints in testrun.py are clear. Just modify your purpose.

“Testrun.py” – constrains part: there is kind of mistake you made in your first two constraints. E.add\_constraint(~q12 | ~q21) . You wrote something like this and said only one of these two can be true. Remember what we learned about disjunction, ~q12 is true and ~q21 is true, ~q12 | ~q21 is also true! You cannot just consider disjunction as “or”. You can see professor’s example constraints to find some clues. I am not able to just tell you the correct one. Actually, all of your constraints should be changed.

One more thing about your constraints is that, you need to contain all the constraints in your code. Not just a part of it. If you run your code, you cannot see a right return values with an incomplete or wrong constrains. Fulfill your constraints.

Continue with your report part of constraint, I see you do a lot of updated propositions and constraints thing at the end. Just a little suggestion that you can delete the previous version of propositions and constraints, and just leave a newest version on the report. I am kind of confused by reading it all the way to the end.

Overall, The first thing you need to do is to improve your clarity, and then make your constraints right.

Answering your questions:

* How is our model complexity? We are planning to go with testrun.py as it seems more complex than the alternate version, but we are still wondering if it suffices in complexity

I do think the complexity of the content in testrun.py is complex enough from my perspective. Just finish your testrun.py and keep improve it.

* Not sure what should be included in the Jape Proofs file. Are we trying to prove that our constraints are valid or that the solutions provided are valid? If we were trying to prove that how would we go about it?

I think you just need to go play with your constraints (as your antecedent) and prove something randomly not purposefully (as your consequent).

Other questions may can only be answered by course staff.

**TA Feedback**

Great work so far! I can see the progress and appreciate all the consideration you’re putting into your model. Here are some suggestions to consider:

* Take testrun to a slightly further extension -- proposition shouldn't represent a full schedule, but a set of propositions should. Here are some tips on how to get started with it:
  + Assume there are K steps of the schedule (think of them as time steps).
  + For each step, you should have propositions for each process that captures what the current code block is (so if there are B blocks for a process, there are K\*B such propositions)
  + Propositions that indicate if a code block requires a specific resource.
  + Constraints that say you're only in a single code block for a step.
  + Constraints saying you start at the first code block and end at the final.
  + Constraints saying that the code block of step i+1 is either the same as step i, or the following code block.
  + Constraints that forbid resource contention.
  + Propositions of the form **using\_p\_r\_k** to represent process **p** is using resource **r** at step **k** would be useful. Consider how this affects “wait”
* Jape proofs:
  + You’re basically just trying to use your explicit propositions/constraints (as premises) to prove a conclusion in your theory that didn’t need to be explicitly stated. ie. Prove to us that your model actually works.
  + Simplify the model in your minds if this feels overwhelming
* First order extension
  + Look for places in your constraints/propositions where you use English words “all”, “every”, “only”, “at least”. These translate well into universal and existential statements.
* The exploration you’ve done for the model is very detailed and looks fine for now.
  + get the implementation going with 2 processes, visualize what's going on, and start adding constraints
* Line referencing the code is perfect

Overall, I’m quite pleased with the progress you’ve made and the direction this is taking. I’m excited to see the final product!

Please reach out to me [cw143@queensu.ca](mailto:cw143@queensu.ca) (Catherine) or Prof Muise if you need more help/feedback.