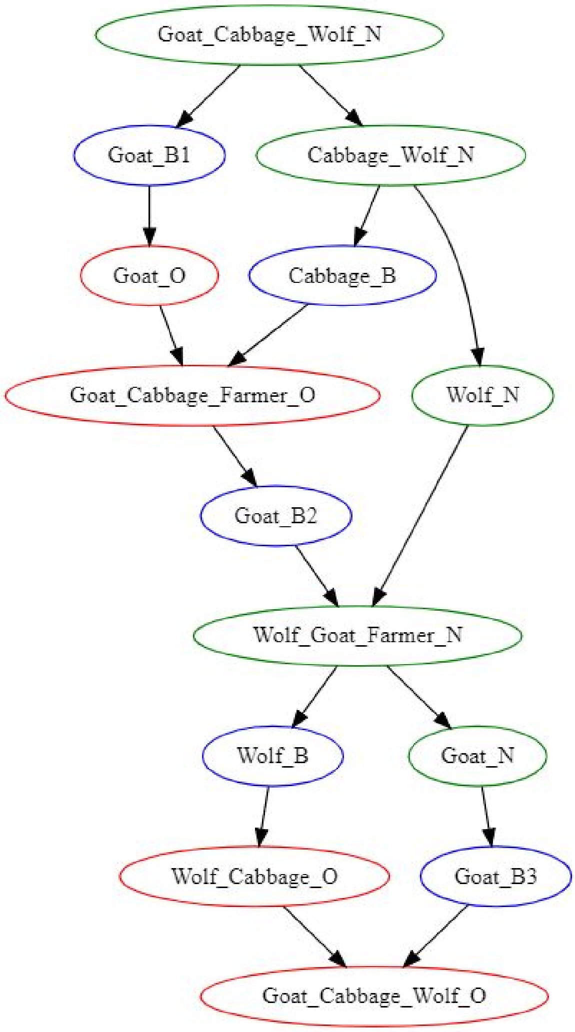
Sao Chi Pham, Johanna Spaude, Florian Murzov-Pirkl

Lab 5 - State Transition Diagrams

1. Model the modified goat/cabbage/wolf problem: The farmer is in Niederschöneweide and wants to get his goat, his cabbage and his wolf over to Oberschöneweide. Only one thing can fit in his boat at a time beside himself. He cannot leave the cabbage and the goat or the goat and the wolf alone on the same side of the river, for obvious reasons. Is it possible for him to get all three possessions across the Spree? Draw a State Transition Diagram modelling a solution to this problem.

We started this lab by thinking together how we could solve this problem. After discovering that we have to take the goat a few times in our boat we wanted to draw the state transition diagram for this problem. Therefore, we used the tool http://www.webgraphviz.com/. So, we had to understand how the tool works and how we can write nice diagrams.

Johanna then took the first task.
I understood how we could solve this problem and wanted to draw a nice diagram. My first idea was to create a diagram similar to a sequence diagram, where I would have three columns. One for the states in Niederschöneweide, one for the states in the boat and one for the states in Oberschöneweide. I tried to model this with the webgraphviz and the cluster layout. But in this layout the stages can only have short names. I would have to use many abbreviations. This did not look nice, so I tried other layouts. I finally decided to use just a simple flow diagram layout.
The states that are placed in Niderschöneweide have the letter N in their name and are painted green. States of the boat are blue and have a B in their name. States in Oberschöneweide are red and have an O in their name. This was my idea of having the difference of the places in this simple diagram. I went from the first state where everyone is in Niederschöneweide to the next states. Now, the first states of the boat (that is not empty) is, that the goat is in the boat. The state of Niederschöneweide changes to have only the cabbage and the wolf left. In this flow I draw the whole diagram. I decided to leave out the farmer in the simple states, where the sheep nor the cabbage gets eaten. But in states where the goat could eat the cabbage or the wolf could eat the goat, I included the farmer. I wanted to make sure, that it is clear to everyone that this is not a mistake in the diagram. My diagram looked like this in the end.



2. Model the states an IMI student passes through from the first until the sixth semester.

Flo took this exercise and included the feedback of Johanna and Sao Chi.

By starting this exercise, I was first writing many different possible states on paper, just barely ordering them in different groups. My first impression was, it might become a small diagram, but after a short time of brainstorming, I had already many ideas collected. And some additional ideas came from my teammates.

Right from the beginning, I had the idea in mind, that the first semester start is something special and therefore it has no loop going back to that part. Basically, everyone is more or less nervous right at the beginning. Some people get even more nervous, after they notice, what IMI is about and some students become euphoric, because it is much better than they thought it will be. Both groups of people will be overwhelmed, the one group in a good way and the other one in a bad way.

For a few, everything is just as expected and therefore they just lose their nervousness.

Sooner or later, everyone gets to the point to think about, how to deal with everything (e. g. studying, doing homework, maybe going to work, having a private life and sometimes also a rest). We thought, there’s one group which makes promises, a second group, which just do not change anything and it results in chaos and a third group which thinks about alternatives (but maybe not changing anything). Desperate students of the third group might consider to start big in McDonald’s, but they all ended up poor. But very few students made indeed breakthroughs with their founded startups and we could find no example of the ones who stayed in the university after they became rich.

The ones with the promises, might throw them already after a short time and it results in chaos. Same with the ones who's strategy it is to just cut sleeping, because it will not work and therefore chaos is waiting for them, too. The ones who came up with good promises and keeping them, eventually became good students. The ones which had the promise to go to the IMI semester party have a head start to become good students, but still have a chance to end up in chaos as well (this follows the logic: "High risk, high reward"). Interesting is also, that a not underestimated number of students decided to stop attending the lectures and do not prepare (much) for the exams and therefore lower their chaos and exam stress, but still manage to pass (some) exams (although many do fail). Maybe a few even enjoy not doing anything for the university too much and do not even try the exams until they completely drop it or get dropped, due the three semester rule.
Beside the last group, everyone else will have more or much more exam stress and hope for the best result(s).

The group who failed the exams (or became poor with alternative ideas) are for sure first devastated and maybe quit or must quit due to the three semester rule. But the ones who stay, must go back to think about their strategies.

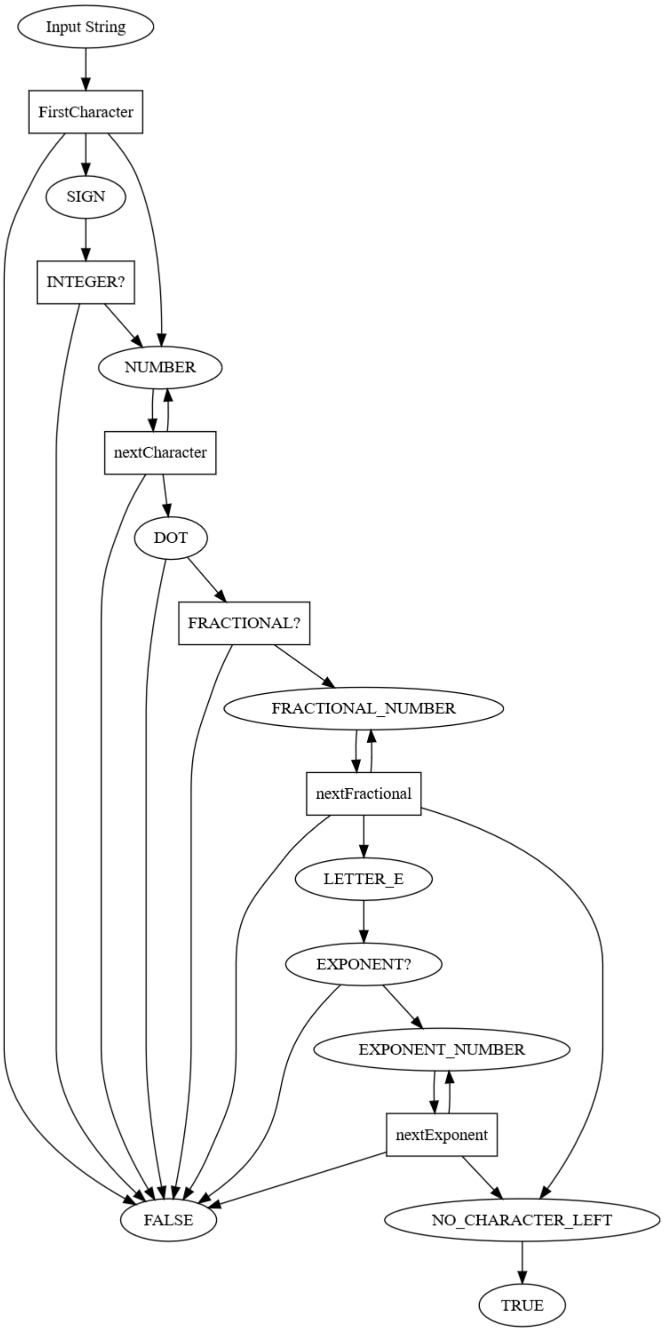
Students, who successfully passed their exam(s), they either will party or relax for sure. Eventually, each semester (break) comes to an end. As long as they did not graduate (or planning to continue studying), another (normal) semester or the internship is ahead of them. For sure it will be emotionally again, because the student might be just bored or motivated to go into it, or maybe depressed, because the break is over and that might even make some consider to quit their studies. The ones who decided to go into the new semester, they got to think about what to do with it and evaluate their strategies and see, if they are working (again).

Looking at our diagram, it became clear to us, somewhat in the middle of the student's life stands "Pure chaos".

3. Model an algorithm that determines if a given string is a proper floating-point number, i.e.
`[sign] integerpart dot fractional part [E exponent].`

The algorithm itself is actually quite simple since we only have to validate the String and return either false or true by going through each character sequentially. However, the first iteration was really messy. The first version resembled more a flow chart like the example image in the wikipedia entry for flow charts. The state was the index of the character of the input String but this proved as problematic with a graph because pointing to the incrementing the index node would lead to arrows coming from everywhere. The graph did not convey the sequential steps properly, so we tried to make an activity diagram.
This way an action would be taken depending on what a character is at each given step. The action we take in this case is always looking at the next character and determine what "state" the character should be.

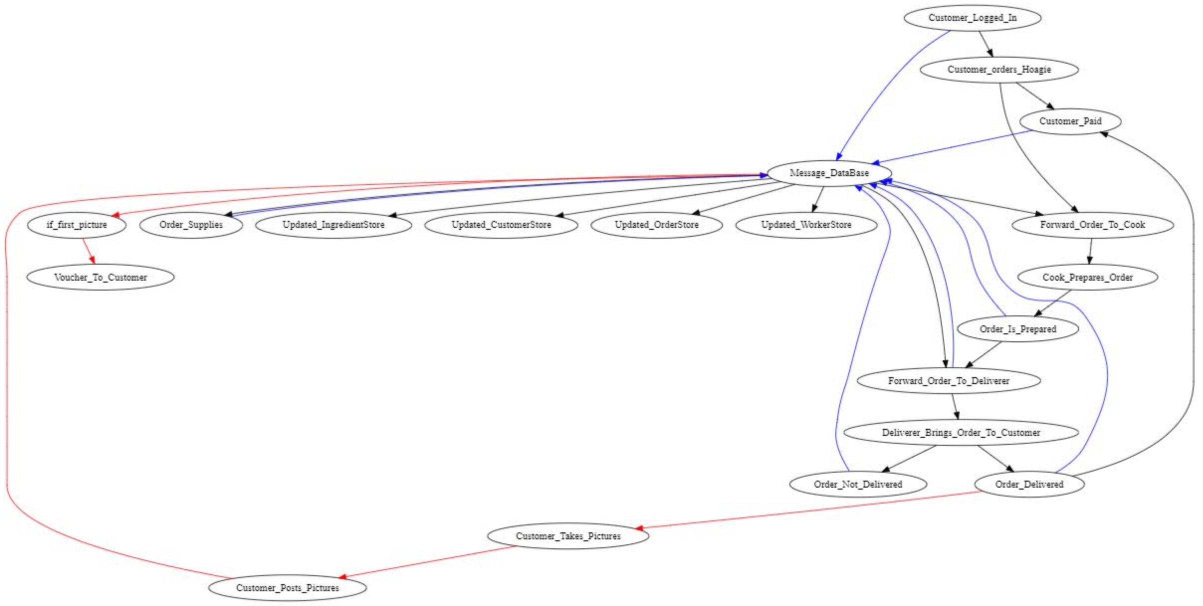
Now, the order is much clearer, e.g. a sign can never appear twice in a floating point, only as the first character optionally. Only when the conditions are true, the algorithm can keep processing, else it will just return false.



4. Model the states for the Hoagie To-Go Webshop for which you designed the class diagrams.

We did this last task together what took us quite some time. We were a little bit confused because we had a big class diagram from last week. In the diagram we had many little classes and interfaces. We had to think about how we could include all the classes in our state diagram. We decided to draw one big state diagram because we would have get very small diagrams if we had modeled one for each class. Our first state in this diagram is that the customer is logged in into our system. Now we described the states of an order. We thought that these states would represent our Hoagie Shop the best. We worked our way through the whole process of ordering, preparing and delivering an order. Afterwards we had to include things like the posting of pictures on our website or ordering new supplies. The diagram was really long, but still nice at some point. We also coloured the arrows blue that pointed to the database to get a better overview.

Unluckily we had to include the payment in the end of our work. These states completely changed the order of our diagram. Now it looked even bigger and it was not easy to get an overview anymore.



Evaluation

Johanna:
I am getting a little bit confused each time we have to deal with a new kind of diagrams. I sometimes think in the way I would have to think for another diagram. This is the reason why my diagrams are not always as clear as they could be. The state transition diagram is not hard to model as soon as you know the states of your process or your classes. Nevertheless it really remind my of sequence diagrams. I think I would use another tool if I had to do state diagrams another time. It is not very easy to work with Webgraphviz sometimes.
Time: The time in the lab for the Cabbage-Goat-Wolf diagram plus a little overview of task two and three. Two additional hours for the HoagieShop diagram and the report. (210 min in total)

Flo:
This exercise was very enjoyable and especially with such kind of fun diagrams as the one for the IMI students. I was changing, removing, adding so many times, but I did not lose motivation to continue working on it. But the other exercises were interesting and very helpful, too, because it showed me for how many different things state transition diagrams can be used for.
Time: For the IMI student diagram I spent about 100 minutes (most of it in the lab), another 30 minutes for the HoagieShop and about 10 minutes each for reviewing and giving feedback for task one and three. I used another 60 minutes for writing the report (about 210 minutes as well in total).

Sao Chi:
I felt that the state diagram was really confusing because the bigger it got, the more complex it became to handle and actions and state can be confused too. In the case of the algorithm I prefer pseudo code, however, it was also good at showing the logic. So, I think it is good when you try to explain a complex process to somebody else.

Time: The time in the lab plus one and a half hour for finishing the algorithm graph. Additionally, the time for writing the lab was half an hour.