Tutorial Slide

Week 1

IN0003

Jigao

TUM

October 23, 2018

About Me

Jigao
Major: Information System in 5th semester
Homeland: China

About You

- How many of you are Master?
- How many are studying Informatics?
- How many are studying Games?
- How many are studying Winfo?

Organizational

- I have only one time slot. ....
- And the weekly Assignment is **optional**. You do not have to speed too much time on a "Grade bonus".

The "Weakest Precondition" for this lecture

You should have lernt before

- discrete mathmatics
- a program language (maybe imperative)
- **Recursion!!!**
- data structure and algorithms
- It would be better, if you know **lambda expression**.

Some useful things

- the lecture website `https://www.in.tum.de/i02/lehre/wintersemester-1819/vorlesungen/functional-programming-and-verification/`
- the lecture videos (but in german) of last year: `http://ttt.in.tum.de/lectures/index_ws1718.php`
- **the central tutorial about WP:**
  the video with Nr 7 at 2017.11.24 in the link above
- all the exercise sheets and exam from last year:
  (we don't publish the solution of exam): Maybe at studystuff

Some useful things (continue)

- WP-Tool: `http://www2.in.tum.de/wp-tool/`
- A Online course about SML (a variant of ocaml): `https://www.coursera.org/learn/programming-languages/home/welcome`
- Piazza `piazza.com/tum.de/fall2018/in0003/home`
- My slides: `https://github.com/cakebytheoceanLuo/IN0003_Tutorial`
- My E-mail: jigao.luo@tum.de

Questions

- When should we start? At 2:00 or 2:15 ...
- Any other questions?
- This time I will just mainly use the Exercise Sheet.
- I hope you can install the ocaml compiler at next time.

Installtion

- Which OS do you have?
  For Linux and Mac that will be easy. For Windows much harder. I would not recommand you program ocaml on a windows, especially you want the "grade bonus".
- It needs some time. I just show you the needed instruction. You should install that after the tutorial.
- If you have question, ask Piazza!
- What I tried: Utop(Recommaned), Visual Code Studio(Recommanded), Emacs and Vim (Not Recommanded)

Way1 - Refer to the Ocaml website

- `https://ocaml.org/docs/install.html` shows how to install ocaml and OPAM on different OSs.
- `https://try.ocamlpro.com/` give beginners possibility to try ocaml in web. (Not good enough for Exercise)
- You just get the compiler! After that you can write ocaml in your SHELL.
- Maybe you need a formated text editor, which acquires a lot time to configure. (Next Page)

Way2 - Refer to the Piazza from last year

- I just copied that from Piazza and let it remains in German.
- Don't worry they re-publish that in English a month later.

From Ralf Vogler

Wir empfehlen folgendes Vorgehen:
1. Opam installieren (macOS: **brew install m4 opam**, Ubuntu/Windows: **sudo apt install m4 opam**)
2. **opam init –comp=4.05.0** und mit **y** bestaetigen damit eure bash/zsh-Config angepasst wird
3. **eval 'opam config env'** oder neue Shell oeffnen
4. **opam update**
5. **opam install utop ocp-indent merlin**

From Ralf Vogler

6. Visual Studio Code installieren (macOS: brew cask install visual-studio-code), OCaml Extension. installieren. Alternativ Vim or Emacs einrichten.
7. Test dass alles funktioniert:

```
1 echo 'print_endline "hallo"' > test.ml
2 ocamlbuild test.native
3 ./test.native
```

1. $x = 1 \Rightarrow 0 < x$
   - How do I consider that?
   - $x = 1$ will statisfy the left side (it just itself)
   - So if left is correct, what will the right be?
   - The right is also correct.
   - The Implication holds!

2. $x < 6 \Rightarrow x = 3$
   - How do I consider that?
   - all the $x < 6$ will statisfy the left side
   - So if left is correct, will the right one also correct?
   - No. $\neg\exists x < 6 \wedge x = 3$
   - The Implication doesn't hold!

- Do you have any inspiration or Idee, about how the Implication works?

- Do you have any inspiration or Idee, about how the Implication works?
- It it always **the stronger one implys the weaker one**.
- In our case **the smaller set implys the bigger set, which contains the smaller set as a subset**
- So to say $\{x|x = 1\} \in \{x|0 < x\}$ and $\{x|x < 6\} \notin \{x|x = 3\}$

6. $x = 1 \Rightarrow x \leq 3 \wedge y > 0$
   - How do I consider that?
   - the $x = 1$ will statisfy the left side
   - So if left is correct, will the right one also correct?
   - Or is the left one a subset of the right one?
   - No! the left one is equals $x = 1 \wedge \forall y \in \mathbb{Z}$
   - The left one only have some common part with the right one. But not a subset-relation!
   - The Implication doesn't hold!

10 and 11

- Important here!!!
- *true* is the weakest, because it require nothing to be a *true*, which always is statisfied and independent on the variable-choosing (whatever a variable you choose, it will have a *true*)
- *false* is the strongest, because it can never be statisfied. So it implies everything!
- Want more information? You could see the Assignment 1.3 or the Aufgabe1.2 from last year!

15 and 16

- Remark: $A \Rightarrow B \equiv \neg A \vee B$
- Or you could do a logic-table.
- Or just intuitively think and come to answer.

- Assertion: a statement about the state (the values of variables) at a program point
- Can be exactly precise, or can be weaker...
- More in Assignment 1.3
- I think only the precise assertion would make sense mostly, and that is more intuitive (easy?) to come to.
- Hint: What does this program do? Then you will the point C.
- I will give you time to exercise.