`print('Hello Python!')`

Basic Programming in Python

Sebastian Höffner Aline Vilks

Wed, 05 Apr 2017

Programming is easy

But it takes a lot of time to become good at it.

```
print('Hello Python!')
```

Programming is easy

Programming is easy

But it takes a lot of time to become good at it.

- It's like sports: easy to run, but it takes a lot of practice to run a marathon.
- You need ten thousands of hours to master a skill – we can only present you with your first 50 or so.

Why programming?

Programming in academia

Programming outside of academia

```
print('Hello Python!')
```

Why programming?

- Course work focuses more on contents, less on tools
- Data analysis becomes more fluent
- Theses and papers become easier
- You learn to automate things to be more productive

- Understanding the principles helps with every day tasks:
  - using office programs
  - filing your tax returns
  - understanding insurance policies
  - ...
- Understanding technology makes things easier and less magic
- Basically every job for academic people involves code

Cliqr

Figure 1: We will use Cliqr throughout the class. Please bookmark http://vt.uos.de/bufuv

```
print('Hello Python!')
```


Figure 1: We will use Cliqr throughout the class. Please bookmark http://vt.uos.de/bufuv

Responses: https://goo.gl/VhpXY1

Questionnaire results: speed

- This class is slow, so no worries. For a faster class, consider Brian's on Thursdays 12:00 - 14:00. We target mostly master students with a non Computer Science background who did not program before.
- Yes, there will be a lot of work – but we try to help you. And it's okay if you don't do some, but we recommend it.

Questionnaire results: support

- Almost 50 % said they think feedback sessions are important.
- We can handle at most 12 feedback session groups, so here is our offer:

1. Form groups of sizes 3-4 if you want to have feedback sessions and sign up for groups on Stud.IP which have a time slot.
2. Form groups of sizes 2-4 if you do not want to have feedback sessions and sign up for groups without.
3. If there are not enough groups, please get in touch with us.

Questionnaire results: contents

- We did not plan to go too deep into data analysis and applications (Brian will hopefully cover that), but as it is a popular request in write-ins, we will try to push the emphasis more towards that.
- Some people stated they did not own a laptop to work on. Please identify yourself after class so we can find a solution.

Lecture

- Time: Wednesday, 14:00 - 16:00, c.t.
- Room: 93/E15

(Complete list in Stud.IP)

Homework

- One sheet per week ($\sim$ 13 sheets in total)
- Deadline is before Monday morning, 08:00
- In groups of 2–4 students (as discussed, either with or without feedback sessions)
- If you work in groups, try to tackle to problems together, don't split the tasks among you.
- Sign up on Stud.IP

Feedback sessions

- Once per week
- Homework will be checked and commented on
- Not a bad thing, but an opportunity!

Grading

- Only Fail/Pass
- To Pass: Present at least nine homework assignments to your tutor (They do not need to be perfect, but you should have worked on it)
- Note: There will be small projects at the end which will be split among several homework sheets, they thus count as two to three assignments

Let's learn together

Ask questions

- Ask questions in **class**
- Ask questions in the forum[1]
- Ask questions per mail

Share your knowledge

- Collect your error messages and the code which produces here[2]
- Try to solve them
- We will have a session soon (in May) where we discuss different errors

[1] https://studip.uos.de/plugins.php/coreforum/index/index?cid=e7eca86bfdacf12717540d75bb2fcb47

[2] https://docs.google.com/document/d/1heObG6cQhuub8hgcTDVB4Pty71woSrX8-SF4p2OAqTs/edit?usp=sharing

Let's program!

We will often see Pseudocode: algorithms written down in a concise way, but close to natural language.

Pseudocode example

```
If it is sunny
    I like to go swimming
If it is rainy
    I like to play in puddles
Otherwise
    I stay at home
```

Pseudocode example

```
For 10 apples inside the crate:
    Take it out
    Put it into your shopping cart
Move your shopping cart to the cash point
Start a receipt with 0 EUR
For each apple inside your shopping cart:
    Take it out
    Weigh it
    Get the price for the weight
    Increase the receipt with its price
    Put it into your shopping bag
Pay the sum on your receipt
```

Pseudocode example

Write a little pseudocode yourself! For example:

- How to pass this class?
- What to wear? Red or blue T-shirt?
- . . . ?

Hello World!

The starting program for almost every programming language is a `Hello World!` program. It is a program which somehow prints[3] a friendly message:

```
Hello World!
```

[3] "printing" means to output something, usually on the terminal. Don't bring out your printers and throw stacks of paper at us. We have nothing to throw back.

Hello World Pseudocode

```
print "Hello World!"
```

Hello World in Python

```
print("Hello World!")
```

Hello World in other programming languages

MATLAB

```
disp('Hello World!')
```

Prolog

```
message('Hello World!').
```

Hello World in other programming languages

Java

```java
class Main {
    public static void main(String... args) {
        System.out.println("Hello World!");
    }
}
```

C++

```cpp
#include <iostream>
int main()
{
    std::cout << "Hello World!" << std::endl;
}
```

Hello World in other programming languages

Arnold.C

```
IT'S SHOWTIME
TALK TO THE HAND "Hello World!"
YOU HAVE BEEN TERMINATED
```

Brainfuck

```
++++++++++[>+++++++>++++++++++>+++>+<<<<-]>++.>+.+++++++
..+++.>++.<<+++++++++++++++.>.+++.------.--------.>+.>.
```

Hello World!

`Hello World!` programs give us a first impression of the language of a syntax. There are other demo programs but we will take a look into some later.

Keep in mind: the concepts are always very very similar!

Back to Python

```
# This prints Hello World!:
print('Hello World!')
```

Output:

```
Hello World!
```

```
print('Hello Python!')
```


Back to Python

```
# This prints Hello World!:
print('Hello World!')
```

Output:

```
Hello World!
```

You can use comments in your code: just start a line with # and it will be "ignored" by Python.

Your first homework

Setup your laptop to run Python (we will discuss this in a minute). Write your own `Hello World!` program. Draw a little St. Nicholas' house. (Not on paper, of course.)

Installing Python

Miniconda is a package management system which allows us to keep the administrative overhead of installing Python to a minimum.

- Download Miniconda (Python 3.6) from https://conda.io/miniconda.html.
- Install it. Make sure it is in your path.
- Open your terminal / command line and run the following to install an IDE[4] we will and packages we might use:

```
conda -y install pip spyder numpy matplotlib scipy
```

- For stuff used in e.g. Neuroinformatics, Machine Learning, Computer Vision, or other classes, run additionally:

```
conda -y install pandas jupyter scikit-learn scikit-ima
```

[4] Integrated Development Environment

The last slide

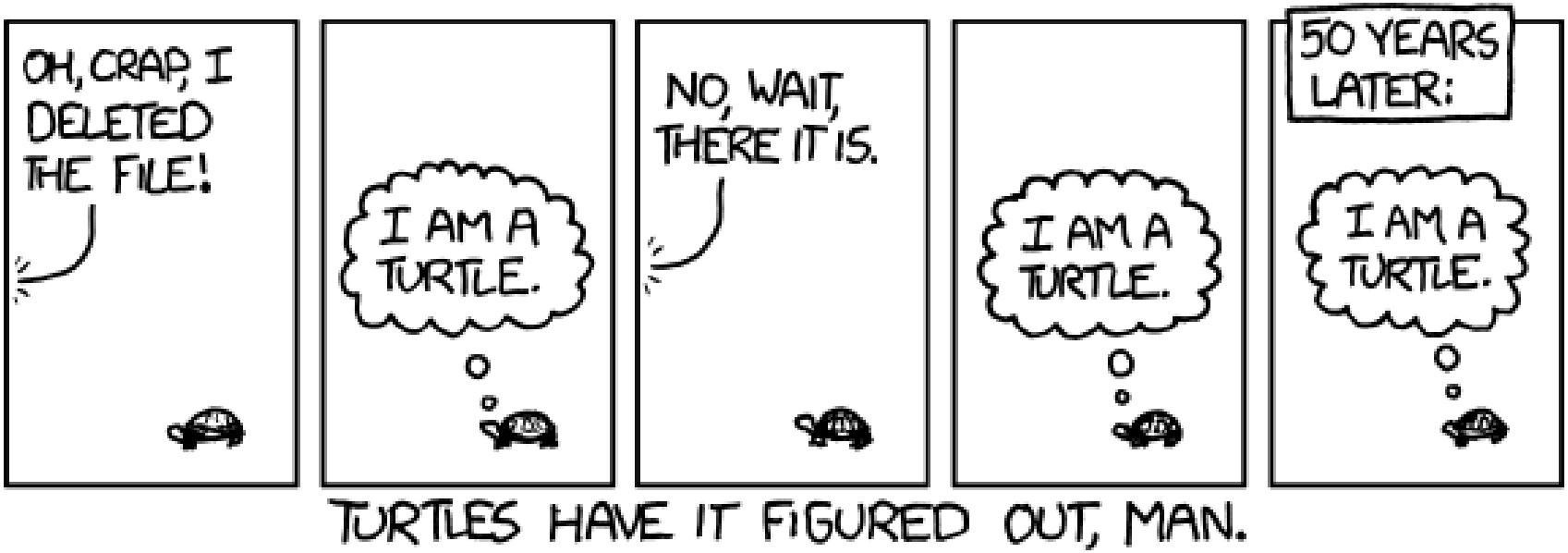

Figure 2: You're a turtle! (Munroe 2011)

References

Munroe, Randall. 2011. “Turtles.” *Xkcd. A Webcomic of Romance, Sarcasm, Math, and Language.*, no. 889 (April).