



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Lecture 1: Introduction

Introduction for Linguists (LT2102)

Markus Forsberg
Språkbanken
University of Gothenburg

2010-09-07



Message from Martin

GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

- ▶ GSLT seminar day + post seminar
- ▶ September 13
- ▶ <http://www.gslt.hum.gu.se/current/calendar/>
- ▶ You need to sign up for the post seminar.



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

About the course

- ▶ *Introduction for Linguists* ⇒ *Introduction to computer science.*
- ▶ Compulsory for students with linguistic background.
- ▶ The main goal of the course is that you will learn how to program using the programming language *Python*.
- ▶ Note: you have very different backgrounds, so it is important that you communicate what you find most difficult.
- ▶ Note 2: learning how to program is a practical knowledge — sit in front of a computer while reading the course literature.



Schedule

GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

- ▶ Course homepage:
spraakbanken.gu.se/personal/markus/introduction_for_linguists
- ▶ We will meet at Tuesdays and Fridays:
 - ▶ lecture
 - ▶ assignment supervision
- ▶ 45 min + 15 min break + 45 min



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Course literature

- ▶ Main course book: *Python for software design, how to think like a computer scientist*, Allen B. Downey
- ▶ *Natural Language Processing with Python*, Steven Bird et al. (we will only use the first chapters, but it is the main book in the 'Programming in NLP' course)
- ▶ The books are available online for free (linked from the homepage).
- ▶ Paperbacks cost around 25 euro each.
- ▶ Python documentation at the Python website: <http://docs.python.org/release/2.6.5/>.



Assignments

GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

- ▶ 3 obligatory practical assignments.
- ▶ The assignments are done in groups of two.
- ▶ Do not make the mistake of being a passive member of a group! Switch control of the keyboard frequently!



Exam

GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

- ▶ October 27, 8-13, V30 (Viktorlagatan 30)
- ▶ Grade: Pass with distinction, Pass, or Fail



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Computer science crash course

- ▶ Computer science is the study of *computation*.
- ▶ computation = problem solving
- ▶ *Algorithm*: a detailed account of how to solve a problem.
- ▶ *Programming language*: a formal language to express computations.



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Formal vs. natural languages

- ▶ *Natural languages*: what we normally mean by languages, i.e., what people speak.
- ▶ *Formal languages*: man-made languages designed for a specific purpose, such as programming languages.
- ▶ Differences:
 - ▶ *ambiguity*
 - ▶ *redundancy*
 - ▶ *literalness*



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Programming in a nutshell

- ▶ Input (keyboard, file, other devices)
- ▶ Output (screen, file, other device)
- ▶ Math (addition, multiplication)
- ▶ Conditional execution (select what to execute based on a condition)
- ▶ Repetition (usually combined with conditionals)
- ▶ That's it!
- ▶ Or is it?
- ▶ Let us print *Hello World!* on the screen.



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Low-level language: Assembler (makes computers, not humans, happy)

```
section .data
    str:      db 'Hello world!', 0Ah
    strLen:   equ $-str
section .text
    global _start
_start:
    mov eax,4
    mov ebx,1
    mov ecx,str
    mov edx,strLen
    int 80h
    mov eax,1
    mov ebx,0
    int 80h
```



High-level language: Python

GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

```
print "Hello world!"
```



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

High-level language

- ▶ The difference is in the level of *abstraction* — details are hidden in a high-level language.
- ▶ A high-level language allows us to be much more productive.
- ▶ However, for every new programming language you need to learn the abstraction of that language.



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Standing on the shoulders of giants

- ▶ Programming is all about building on what others have done.
- ▶ Using a high-level programming language is exactly that.
- ▶ Instead of trying to reinvent the wheel, we often use code defined by others that helps us solve a particular problem.
- ▶ Python terminology: a *library* consists of *packages* that consist of *modules*. A module is a file containing code. (More about this later)
- ▶ *Python standard library* is always available with the Python program.
- ▶ However, we will actually many times reinvent the wheel in this course just for the practice.



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Some Python terminology

- ▶ *values*: basic things a program works with, like letters and numbers.
- ▶ *expression*: denotes a value, possibly after some computation (5+5 denotes 10).
- ▶ *types*: every value has a type, e.g., **2** is an *integer*, "**Hello world!**" is a *string*.
- ▶ *variables*: gives a name to a value. A variable has the same type as the value.
- ▶ *Statement*: performs an action, such as printing a string or assigning a name to a value.



Example

GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

```
$ python
...
>>> type(12)
<type 'int'>

>>> type(12+12)
<type 'int'>

>>> name = 12+12

>>> name
24

>> type(name)
<type 'int'>
```



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

When things go wrong

Syntax errors are formal errors that may be lexical or syntactical.

Runtime errors are errors, also referred to as *exceptions*, occurring while running a program.

Semantic errors are errors where the program actually runs, but fails to do what we want.



GÖTEBORGS
UNIVERSITET

Språk
BANKEN

CLT

Assignment 0: description

- ▶ Not obligatory, but highly recommended.
- ▶ A hands-on assignment, where you will be familiarized with both programming in Python and Language Technology.
- ▶ Do not expect to understand everything! Just work your way through the examples.
- ▶ Chapter 1 of the NLTK book.
- ▶ We will now spend the rest of the lecture on a live demo to get you started.