

Programmeermethoden NA

Week 1: Introductie

Kristian Rietveld

<http://liacs.leidenuniv.nl/~rietveldkfd/courses/prna/>



Universiteit
Leiden

Even voorstellen ...

- Het hoorcollege wordt verzorgd door de docent:
Kristian Rietveld
- Het werkcollege wordt verzorgd door de studentassistenten:
Leon Helwerda (hoofd), Dennis van der Zwaan, Lennard Kwakernaak.

Doelstelling

- Doel van het vak:

Leren programmeren; leren gebruiken van een programmeertaal in een wetenschappelijke context.

- Waarom leren programmeren?

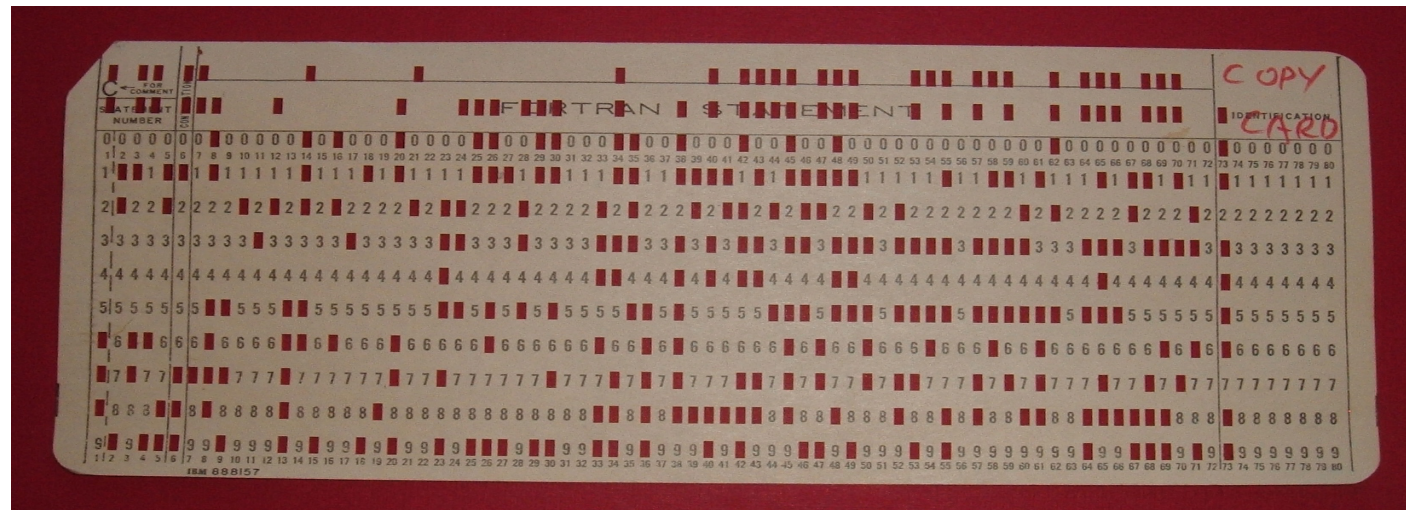
- Taken automatiseren, zoals bijv. data-analyses.
- Bedenk je dat de volumes aan data die tegenwoordig worden verzameld dermate groot zijn, dat deze niet meer met de hand kunnen worden verwerkt.
- Plots maken die mooier en complexer zijn dan Excel.
- **Alles wat je hier leert gaat je later tijd schelen!**

Programmeren?

- Computers en smartphones lijken magie, maar het zijn eigenlijk domme apparaten.
- Een computer moet worden voorzien van (overduidelijke) stap-voor-stap instructies.
 - U mag raden wie er voor deze stap-voor-stap instructies verantwoordelijk zal zijn.

De programmeertaal

- Een computer verwacht dat de uit te voeren instructies worden aangeleverd in een binaire code.
- Het met de hand schrijven van deze binaire codes is zeer tijdrovend (en heel vervelend).



De programmeertaal (2)

- Informatici hebben daar iets op verzonnen: programma's om een door de mens leesbare tekst om te zetten naar deze binaire codes.
- Deze door de mens leesbare tekst is een tekst geschreven volgens de regels van een bepaalde programmeertaal.
- Er bestaan honderden verschillende programmeertalen.
- Welke taal kiezen we dan?

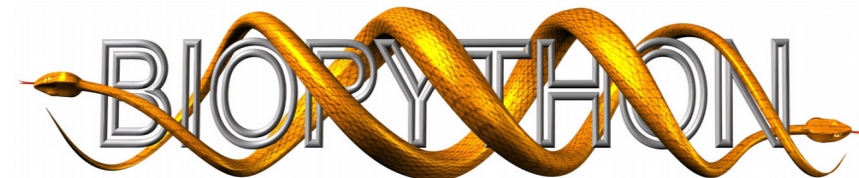
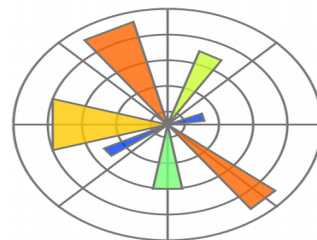
Python: wat en waarom?



- Het hele college staat in het teken van het leren van de programmeertaal Python.
 - Deze taal is ontworpen door Guido van Rossum eind jaren '80 / begin jaren '90.
 - Eenvoudig & portable (werkt op zowat alle systemen).
 - Complexe bewerkingen in maar enkele regels code -- hierdoor een ultiem gereedschap! (perfect voor luie mensen).
 - Zeer populair geworden in de laatste tien jaar, met name voor wetenschappelijke toepassingen.
- Informatica/Wiskunde werken bij het vak Programmeermethoden met de taal C++. Docent: Walter Kusters.

Waarom zo populair?

- Zeer uitgebreide standaard bibliotheek.
- Eenvoudig om uitbreidingen te schrijven en deze uit te wisselen met anderen.
- Er zijn vele modules ontwikkeld voor het doen van numeriek rekenwerk en maken van plots.
- Hierdoor zeer populair in verschillende wetenschappelijke disciplines.



Colleges & werkcolleges

➤ **Hoorcolleges:**

- Dinsdagen, 9:00 – 10:45 uur (incl. 15 minuten koffiepauze).
- Laatste keer: 28 november.
- Twee keer geen college: 3 oktober, 24 oktober.
- Locatie: De Sitterzaal.

➤ **Werkcolleges:**

- Direct aansluitend: dinsdagen, 11:00 – 12:45 uur.
- Locatie: computerzalen begane grond Snellius gebouw.
- Veelal in het teken van de programmeeropdrachten.
- We registreren aanwezigheid.

Gebouwen campus FWN

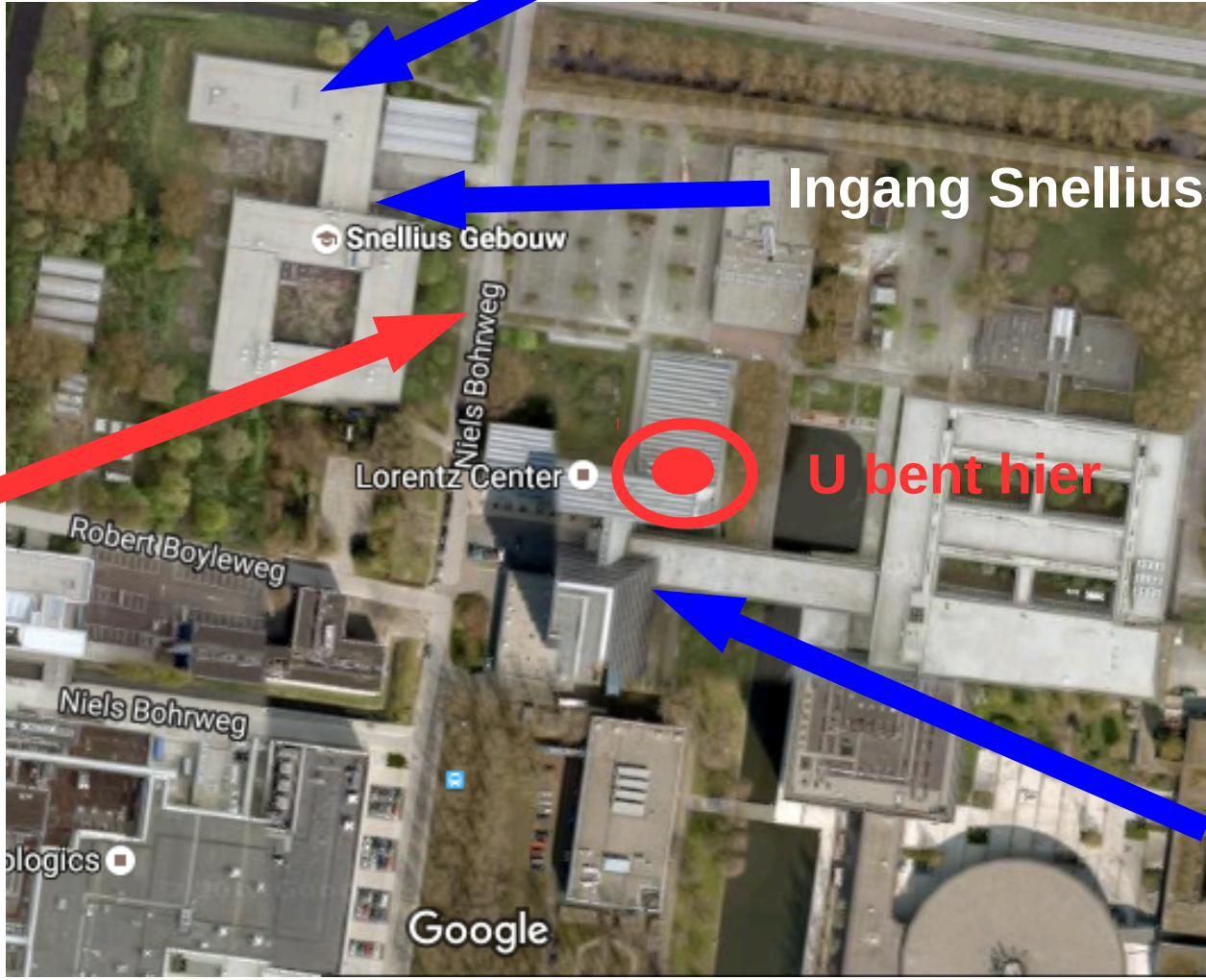
Computerzalen Snellius

Ingang Snellius

U bent hier

Hoofdingang faculteit

fietsers!



Toetsing

- De toetsing van dit vak bestaat uit twee delen:
 - 1. Een schriftelijk tentamen. $\frac{2}{3}$ van het eindcijfer.
 - 2. Een programmeerpracticum. $\frac{1}{3}$ van het eindcijfer.

- Het tentamen zal plaatsvinden op:
 - Vrijdag 8 december, 14:00 – 17:00 uur.
 - *Hertentamen*: vrijdag 2 februari 2018, 14:00 – 17:00 uur.

- Oefententamens beschikbaar op de website!

Toetsing (2)

- Het programmeerpracticum bestaat uit 3 (drie) opdrachten.
- De deadlines zijn als volgt:
 - **Opdracht 1:** maandag (!) 25 september, 13:00 uur.
 - **Opdracht 2:** vrijdag 3 november, 17:00 uur.
 - **Opdracht 3:** vrijdag 1 december, 17:00 uur.
- In te leveren:
 - Digitaal via Blackboard (code & verslag).
 - Print van het verslag (incl. code) in de doos in kamer 156 van het Snellius (postkamer informatica).

Toetsing (3)

- Hoe komen we tot een eindcijfer?
 - Alle programmeeropgaven moeten voldoende zijn (≥ 5.5).
 - Het tentamen moet voldoende zijn (≥ 5.5).
- $\text{Eindcijfer} = (2 \cdot \text{tentamen} + (\text{op1} + \text{op2} + \text{op3}) / 3) / 3$
- Het eindcijfer wordt afgerond naar het dichtstbijzijnde element uit de verzameling
 $\{1, 2, 3, 4, 5, 6, 6.5, 7, 7.5, 8, 8.5, 9, 9.5, 10\}$.
- Als het cijfer van het schriftelijk tentamen onvoldoende is, is dit meteen het eindcijfer!

Programmeeropdrachten

- mogen worden gemaakt in tweetallen.
- toegestaan: overleggen en om hulp vragen.
- niet toegestaan: overnemen/kopiëren code van anderen.
 - Hier wordt automatisch op gecontroleerd!
- zie de website voor huisregels en richtlijnen.
- deadlines zijn strict! (-1 punt per week te laat).
- deels maken tijdens werkcolleges; deels thuis.
- Vragenuren in de computerzalen: wordt nog bekend gemaakt.

De eerste colleges

- *Week 1: 4 – 8 september*
 - **Onderwerp:** Introductie, UNIX, Python
- *Week 2: 11 – 15 september*
 - **Onderwerp:** Types
- *Week 3: 18 – 22 september*
 - **Onderwerp:** Controlestructuren
 - 25 september: Deadline opdracht 1!

Literatuur

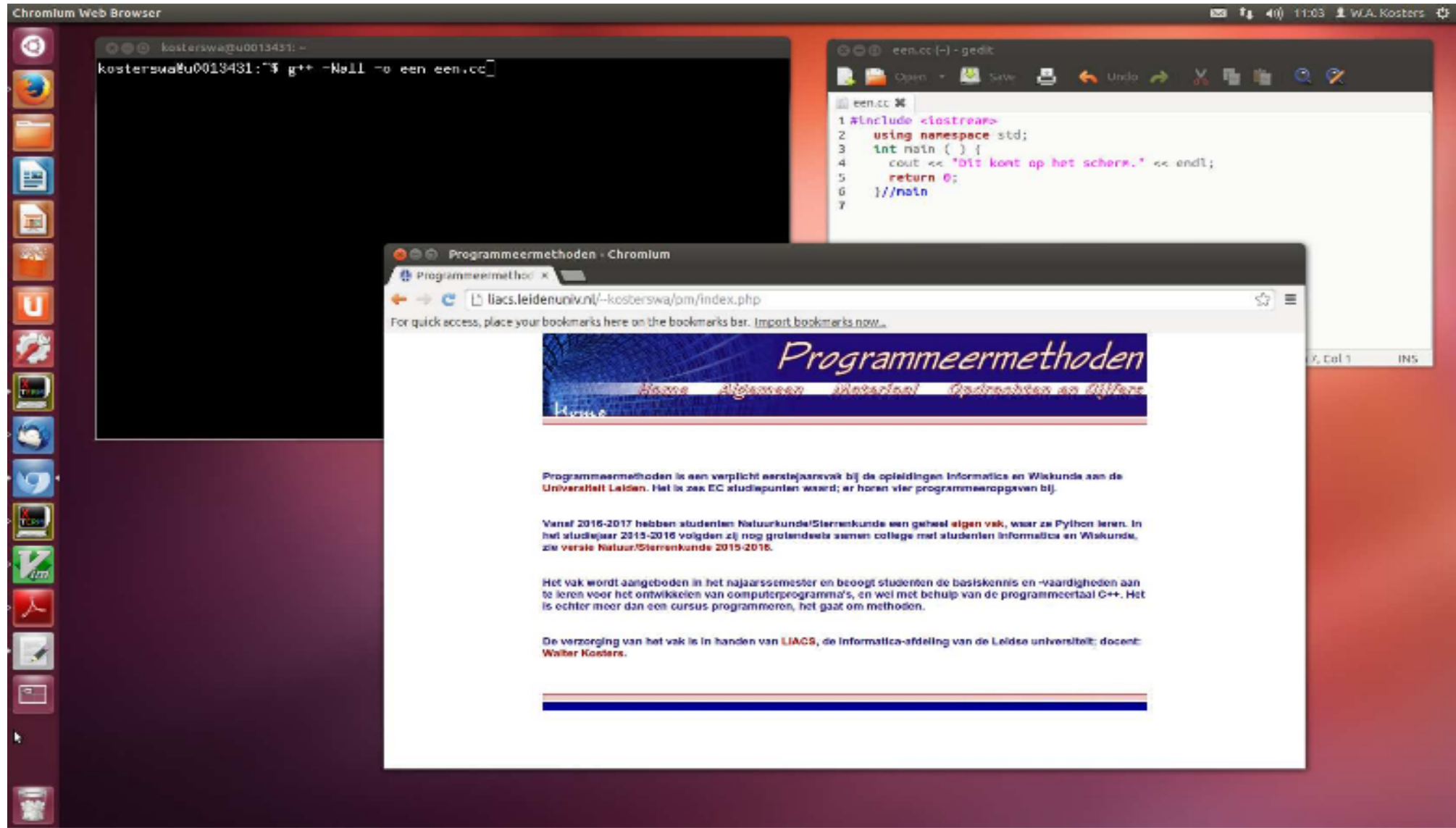
- Het vak heeft geen voorgeschreven boek.
- Er is een dictaat dat in principe alle stof behandelt, zal verkrijgbaar zijn op de website. Zelf printen of gebruik een e-reader.
- Alle slides/sheets gebruikt tijdens hoorcollege zullen op de website terug te vinden zijn.
- Verschillende andere handouts.
- Website bevat ook links naar andere bronnen:
 - Bijv. de Python tutorial.
 - 'Gratis' boeken in PDF formaat.

Operating Systemen

- Naast Microsoft Windows kunnen PCs ook draaien onder *Linux*, een operating systeem (OS) uit de UNIX-wereld (vergelijk Android).
- De besturingssystemen van Apple (macOS, iOS) zijn ook gebaseerd op UNIX.
- De computers in het Snellius draaien zowel Windows als Linux (“dual-boot”). Keuzemenu bij het opstarten.
- Onder Linux staat alles wat je voor Python nodig hebt al klaar.



Ubuntu



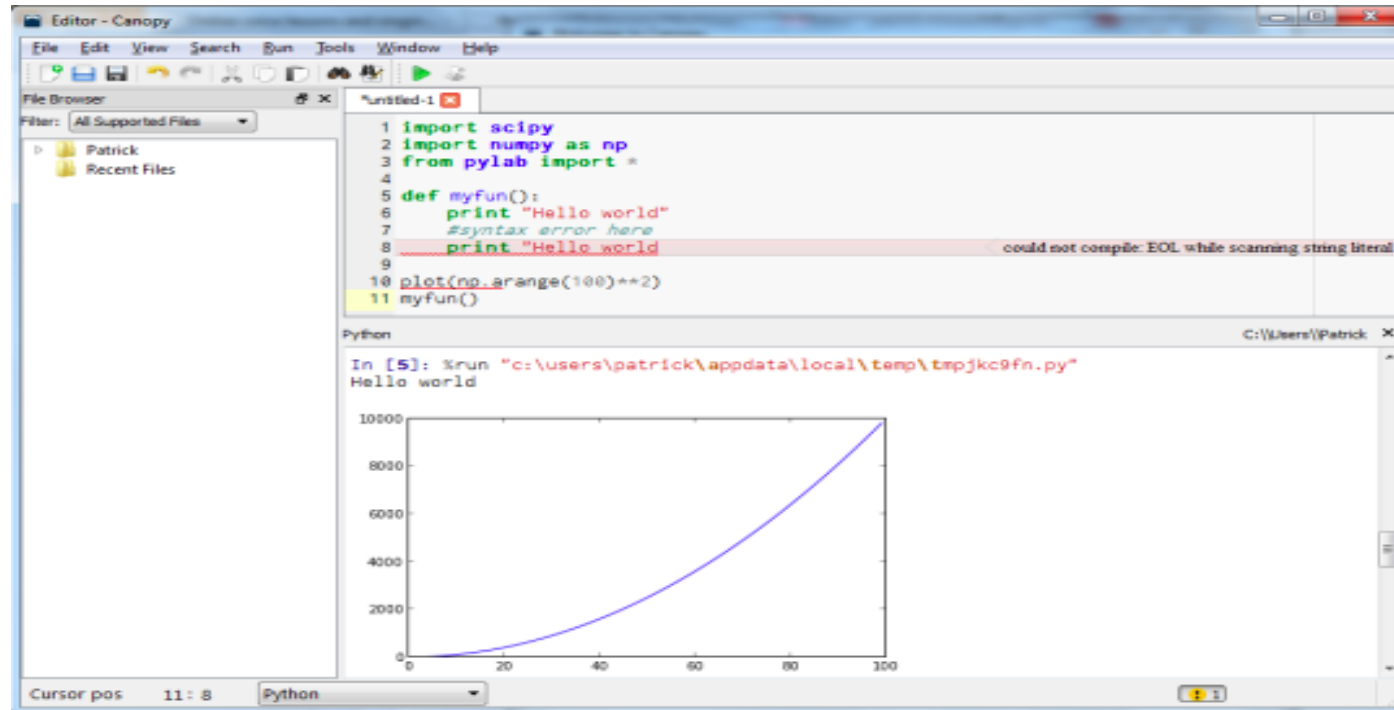
Thuis programmeren

- En wat nu als je thuis wilt werken aan het practicum?
 - Heb je een Mac? Alles staat in principe al klaar.
 - Linux?
 - Haal bijvoorbeeld Ubuntu van <http://www.ubuntu.org>
 - Huidige stabiele versie 16.04.
 - Met een Ubuntu Live-CD/DVD/USB start je PC meteen op in Unity, een grafische windows-omgeving bovenop Linux. Of gebruik KDE of GNOME.
 - Let er op dat je op een verstandige plaats (USB/e-mail/cloud) je bestanden moet opslaan. (Beter: dual boot).
 - Er zijn vele andere distributies (SUSE, Fedora Core, Debian, Arch, Gentoo, ...) -- maar dan moet je er meer vanaf weten.
 - (Of koop een “Raspberry Pi” ?)

Thuis programmeren (2)

– Windows?

- Standaard geen Python meegeleverd, download en installeer een “Python distributie”.
- Bijv. Enthought Canopy Express (<https://www.enthought.com/canopy-subscriptions/>)
- Komt inclusief handige code editor!



Bron: <https://xcorr.net/2013/04/30/canopy-scientific-python-editor-for-windows/>

Ons eerste programma

- Een werkend Python-programma maken gaat als volgt:
 - 1. Schrijf in een *editor* Python-code, het “bron-programma”.
 - 2. Sla het programma op, bijvoorbeeld “eerste.py”.
 - 3. Draai (= run) dit programma (of ook wel script genoemd) vanuit het OS: `python eerste.py`
 - 4. Er kunnen verschillende fouten optreden, ga terug naar 1. om deze te verhelpen.
 - Herhaal deze cyclus zo vaak als nodig.

Ons eerste programma (3)

```
print "Dit komt op het scherm."
```

- Dit programma zet alleen een tekstje op het beeldscherm.
- Let op de “syntax-highlighting”.
- Bij “print” **alleen** kleine letters gebruiken! (Anders: `SyntaxError`).

Ons tweede programma

```
# dit is een simpel programma  
getal = 42 # een variabele declareren en initialiseren  
print "Geef een geheel getal ..",  
getal = int(raw_input())  
print "Kwadraat is:", getal * getal
```

Ons derde programma

```
# Dit is een regel met commentaar ...
import math # voor de "pi" constante
print "Geef straal, daarna Enter ..",
straal = float(raw_input())
if straal > 0:
    print "Oppervlakte:",
    print math.pi * straal * straal
else:
    print "Niet zo negatief ..."
print "Einde van dit programma."
```


Ons derde programma

```
# Dit is een regel met commentaar ...  
import math # voor de "pi" constante  
print "Geef straal, daarna Enter ..",  
straal = float(raw_input())  
if straal > 0:  
    print "Oppervlakte:",  
    print math.pi * straal * straal  
else:  
    print "Niet zo negatief ..."  
print "Einde van dit programma."
```

- Commentaarregels

Ons derde programma

```
# Dit is een regel met commentaar ...  
import math # voor de "pi" constante  
print "Geef straal, daarna Enter ..",  
straal = float(raw_input())  
if straal > 0:  
    print "Oppervlakte:",  
    print math.pi * straal * straal  
else:  
    print "Niet zo negatief ..."  
    print "Einde van dit programma."
```

- “Keywords”

Ons derde programma

```
# Dit is een regel met commentaar ...
import math # voor de "pi" constante
print "Geef straal, daarna Enter ..",
straal = float(raw_input())
if straal > 0:
    print "Oppervlakte:",
    print math.pi * straal * straal
else:
    print "Niet zo negatief ..."
print "Einde van dit programma."
```

- Inspringen (indentation).
- Dit moet consistent gebeuren!

Ons derde programma

```
# Dit is een regel met commentaar ...
import math # voor de "pi" constante
print "Geef straal, daarna Enter ..",
straal = float(raw_input())
if straal > 0:
    print "Oppervlakte:",
    print math.pi * straal * straal
else:
    print "Niet zo negatief ..."
print "Einde van dit programma."
```

- Manier om geen nieuwe regel op het beeldscherm “af te drukken”.

Mogelijke fouten

- Wat voor fouten kunnen er optreden bij het draaien van een programma?
- Bij het inlezen van het programma:
 - “*SyntaxError*”: de syntax van het programma klopt niet, er staat bijv. een haakje verkeerd, “elze” in plaats van “else”, etc.
 - “*IndentationError*”: er is verkeerd ingesprongen (komt later aan bod).
- Bij het uitvoeren van het programma:
 - “*NameError*”: er worden variabelen gebruikt die niet zijn gedefinieerd,
 - “*ValueError*”: ongeldige conversie,
 - delen door 0.
 - enz.

Eerste programmeeropgave

- De eerste programmeeropgave heet “Wortels”.
<http://liacs.leidenuniv.nl/~rietveldkfd/courses/prna2017/opdr1.html>
- Eerste kennismaking met programmeren en Python.
- Je moet onder meer voor een gegeven datum de dag van de week uitrekenen.
 - Bijvoorbeeld op basis van het gegeven dat 1-1-1901 een dinsdag was.
 - Hoe zou je dat uit je hoofd doen?
 - Wanneer is een jaar een schrikkeljaar?
- Naast het programmeerwerk dient er ook een kort verslagje te worden geschreven met behulp van LaTeX.

Stoomcursus UNIX

- In een grafische Linux omgeving zoals KDE, GNOME of Unity, start je een of meer “terminals”.
- Een “terminal” is een window waarin je tekst-georiënteerde opdrachten kunt geven aan het operating systeem.
- Bijvoorbeeld:
 - `gedit eerste.py &` - edit je eerste Python-programma; open hiertoe een “edit-window”
 - `python eerste.py` - draai het programma met de interpreter
 - `ls -lrt` - overzicht van je bestanden
- De ampersand (&) zorgt er voor dat je in het oorspronkelijke terminal-window kunt doorwerken.
- Het overzicht van bestanden ziet eruit als:

```
-rw----- 1 kris  staff  20 Sep  2 14:27 eerste.py
-rw----- 1 kris  staff  83 Sep  2 14:27 tweede.py
```

De Python interpreter

- De Python interpreter kan ook interactief worden gebruikt.
 - Handig om dingen snel uit te proberen.
 - Of als “superrekenmachine”.

- *(Demo)*

Inloggen in het Snellius

- Hoe kunnen we inloggen op de computers in het Snellius?
- Iedereen gebruikt zijn/haar *ULCN-account*.
- Hoe kom je vanaf huis bij je bestanden?
 - Antwoord: “ssh-en” en “scp-en” naar:
sXXXXXXXX@sshgw.leidenuniv.nl
 - en dan naar “homedir”.
- Of een USB stick, e-mail, cloud,
- Meer informatie: dictaat “Computers en programmeren” op de website.

Aansluitend werkcollege

- 11:00 - 12:45 uur
- Snelliusgebouw: alle computerzalen begane grond.
- Aanwezigheidsregistratie.
- Doel: UNIX, hello world, ...

<http://liacs.leidenuniv.nl/~rietveldkfd/courses/prna2017/prna-wc1.html>

Afsluitend

- Hello world in Python.
- Dictaat: versie 2016 beschikbaar, update volgt snel.
- ULCN-account.
- Stoomcursus UNIX (volgende week meer).
- Software voor thuis.
- De eerste programmeeropgave.
- Website: <http://liacs.leidenuniv.nl/~rietveldkfd/courses/prna/>

Programmeermethoden NA

Week 1



Universiteit
Leiden