HUMBOLDT-UNIVERSITÄT ZU BERLIN
BERNSTEIN CENTRE FOR COMPUTATIONAL
NEUROSCIENCE

Bernstein Center for Computational Neuroscience
Humboldt-Universität zu Berlin
Philippstr. 13 House 6

Phone: 030/2093-9110
Fax: 030/2093-6771
webpage: http://www.bccn-berlin.de/

Models of Neural Systems I, WS 2007/08
Computer Practical 1

Discussed on 22 Oct 2007

1. Unix basics

In the first part of the practical you will get familiar with the Unix operating system including a graphical user enviroment (GNOME) and an interactive shell (Bash). At the end of the class you should be able to use basic Unix skills, such as loging in and out, editing files, accessing filesystem, modifying access permissions, mounting external volumes and working in a shell.

(a) Create an account and change your password. If you are a BCCN guest log in to the guest account.
(b) Create a text file with your favourite editor. Open a terminal and make the file readable to everyone.
(c) Make a new directory and copy the newly created file to this directory.
(d) If you have a memory stick plug it in, copy the created files and then unmount the volume.
(e) List the contents of your HOME directory together with access permissions in a reverse order. Save the output in a file.
(f) (*Optional*) Add a newly created directory to your default path.

2. First steps in Python

Python is a modern high-level programming language. In contrast to C/C++ and Java, Python is an interpreted language which means that Python programs (also called scripts) don't have to be compiled. It makes programming Python very quick and interactive. It also allows you to see immediately the results of all operations, analyse them or further process them. Here you will learn the basic concepts of Python.

(a) Open an interactive shell. Implement a simple (!) calculator:

```
a = 3
b = 4
print 'a+b=',a+b
```

Open your favourite editor, paste the code and save it as `calc.py`. Run the script from the Unix shell with the command: `python calc.py`.
(b) **Data structures**. Create variables containing: a number, a string and a list. Using introspection find out what methods are available for these structures.
(c) **Slices**. Create a string and then print: first 3 characters, last 3 characters and a reversed string.
(d) **List comprehensions**. Generate a list containing first 10 elements of a power sequence: $\{n^2\}_{i=1}^{10}$
(e) **Functions**. Define a function printing elements of Fibbonacci series smaller than N:

$$F(n) := \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F(n-1) + F(n-2) & \text{if } n > 1. \end{cases}$$

Write a documentation for your function (docstring) and test it. Can you generate Fibbonacci series by recursion?

3. Sieve of Eratosthenes

The first Python script you are going to write will implement a well-known algorithm for finding prime numbers. The goal of this excercise is to practice basic Python skills you have learnt so far. There are many ways to solve this problem but please try to follow the Python philosophy (to list its main principles type `import this` in an interactive shell).

The algorithm of finding prime numbers smaller than n (called Sieve of Eratosthenes) proceeds as follows:
(a) Generate a sequence of integers from 2 to n.
(b) Take the first element of the sequence (lets call it A).
(c) Remove from the sequence all of the numbers which are divisible by A.
(d) A is a prime number - copy it to the output.
(e) Repeat steps (b)-(d) until the sequence is empty.

The task is to write a function implementing Sieve of Eratosthenes for any given n and returning a list of prime numbers.

Please DO NOT turn off the computer

Contact

Jan Benda (room 1301)	Phone: 2093-8652	Email: j.benda@biologie.hu-berlin.de
Robert Schmidt (room 2316)	Phone: 2093-8926	Email: r.schmidt@biologie.hu-berlin.de
Bartosz Telenczuk (room 1309)	Phone: 2093-8838	Email: b.telenczuk@biologie.hu-berlin.de