

Introduction to Programming and Computational Physics

(Computerorientierte Physik)

Presentation of the course

Ciro Pistillo

Bern, February 23rd, 2016

Description of the course

The course is aimed to provide the students with a basic knowledge of the C programming language and of its employment to address physics-related questions such as numerical integration, solving of differential equations and the Monte Carlo method for simulations of physical processes.

The students are not expected to have any background in programming.

Program

Part I: Introduction to Programming

Algorithms

Operating systems

Introduction to C language

- Variables and data types
- Arithmetical and logical operators
- Functions
- Selection
- Iteration
- Arrays
- Data I/O
- Pointers
- Dynamic memory allocation
- Strings
- Structures

Histograms and functions plotting

Program

Part II: Computational Physics

Root finding in one dimension

Numerical integration:

- Trapezoidal rule
- Simpson rule

Ordinary differential equation:

- Second order Runge-Kutta method

Monte Carlo method:

- Random numbers
- Monte Carlo integration
- Simulation of physical processes

Suggested books and learning material

G. Krüger

Go To C-Programmierung

B. W. Kernighan, D. M. Ritchie

The C programming language

W. H. Press, B. P. Flannery, S.A. Teukolsky, W. V. Wetterling

Numerical recipes in C, the art of scientific computing

<http://www.nrbook.com/a/bookcpdf.php>

W. Benz

Numerical Methods – Lecture notes (2007/08)

J. E. Gentle

Random Number Generation and Monte Carlo Method

Where we are

Dr. Ciro Pistillo

Room: ExWi 132

Email: ciro.pistillo@lhep.unibe.ch

Luc Grosheintz

Room: ExWi 009

Email: martti.nirkko@lhep.unibe.ch

Konrad Nesteruk

Room: ExWi A50

Email: konrad.nesteruk@lhep.unibe.ch

web repository

<http://www.physik.unibe.ch> → Studium → Bachelorprogramm →
Computerorientierte Physik (4)

Organization of the course

Frühjahrssemester 2016	22.02.2016 - 03.06.2016
Ferien ab Karfreitag	25.03.2016 - 03.04.2016

Theory: Tuesday 14-15 in **B6**, Exercises: Friday 10-12 in **A95**

Next Friday only: 1 hour theory (in B78) and 1 hour exercises

Exam (June 24th, 09:00-12:00)

Written exam, solving 2-3 exercises and handing in source code

Exam

In terminal room A95 (+A94 if needed)

Open books

No internet access / Skype / Messenger/ SmartPhone ...

Exams done in past years available upon request

Evaluation criteria

The program(s) must *compile* and *return* the correct result(s)

A program is well-written when it...

- Optimizes the usage of CPU time and memory
- Can be easily understood and extended by the author (one year later...) or other programmers
- Exploits the appropriate resources provided by the language

-...

Access to terminal room

Rooms A95 / A94 are reserved for us on Friday 10-12

You can use them at any time (if not reserved)

Your campus accounts are valid to log in but you're not allowed to use the printers and you don't have a permanent home directory (→ **save your files to a USB drive**)

If you would like to have a permanent home and use the printers, please contact Mr. Peppo Brambilla. (brambi@iam.unibe.ch)

All you need is...

- A **computer** (A95 terminal room with Linux/Unix OS or your laptop)
- An **editor** (emacs or kate or jedit on Linux/UNIX OS, Notepad++ on Windows OS)
<http://notepad-plus-plus.org/>
- A **compiler** (We will use gcc, which is probably the most universal one)

Installation of gcc for Windows users

Download the software : “gcc installer” and execute it
(you need to have a network access to download packages during installation)

You could get this message from Windows Firewall:
gcc_installer.exe is not commonly downloaded and could harm your computer **Actions->More options->Run anyway**

Leave all the settings as they are...

Then try
you should get the answer

```
C:\MinGW\bin\gcc.exe  
gcc.exe: no input files
```

How to set an environment variable (Windows 7)

If you don't want to be forced to give the full path `C:\MinGW\bin\gcc.exe` you need to set an *environment variable*.

Open

Control Panel -> System and Security -> System -> Advanced system settings -> Environment variables

Among **System variables** select **Path** and then click **Edit**

Leave the content as it is and add at the end **`;C:\MinGW\bin\`** and then click **OK**

Then you should be able to start gcc simply typing `gcc.exe`

NB: This operation is dangerous...be careful !!!

Linux users: gcc is already installed !

Installation of gcc for Mac users

1. Open the App Store
2. Search for Xcode
3. Click on Install

It should be possible to install gcc without full Xcode installation → ask an expert

FAQ

Can I use my own laptop at the exam?

No, you can't

Should I hand in all my solutions to the exercises?

No, exercises are meant only to learn programming

Is the attendance to the lessons mandatory?

No, it isn't

I'm already a good C programmer, what would you suggest me?

You can skip the first lessons on programming, but you better attend the lessons on computational physics

I don't know C but I know other program languages, may I use them at the exam?

No, only C is allowed (C++ could be an exception)

Can I use another compiler?

It's not forbidden but you should ensure yourself that it has a *high compatibility* with gcc

UNIX/Linux commands

The general syntax of a UNIX command can be summarized as:

```
command [-option] [arguments]
```

Example: obtain the list of the files stored in the /home/pistillo directory

```
> ls -l -h /home/pistillo
```



prompt: it says to the user that the shell is ready to receive commands (sometimes used "\$")

Many other options available. Try the command:

```
> man ls
```

Files and directories

A *file system* is a method for storing and organizing computer files and the data they contain to make it easy to find and access them.

Directories are organized in a tree structure and the command to move inside is `cd`

```
> cd /home/pistillo/data/MC/numuCC/run20/
```

Basics commands for file manipulations are:

<code>cp</code>	(copy)
<code>mv</code>	(transfer or rename)
<code>rm</code>	(delete)
<code>less / more</code>	(show the content)
<code>mkdir</code>	(create a directory)
<code>rmdir</code>	(delete a directory)

Other commands

<code>passwd</code>	(change password)
<code>whoami</code>	(says who is the current user)
<code>pwd</code>	(current folder)
<code>top</code>	(running processes and <i>CPU</i> /memory usage)
<code>ps</code>	(running processes)
<code>kill</code>	(kill a process)
<code>clear</code>	(clean up the terminal)