C++ and Linux tutorial

1. A Linux terminal is a text interface that allows you to write and execute commands. The goal of this exercise is to learn a few simple commands.

   a) Open a new terminal (shell)

   b) Access the LIP machines by typing `ssh -XY username@summer.ncg.ingrid.pt`. If, for some reason, you cannot access the LIP machines you can still follow this tutorial from your own computer. Check this section for instructions.

   c) Create a new directory named *test*: mkdir test

   d) Go into the directory: cd test

   e) Start a text editor software to write a program in C++ (emacs, vim or gedit): gedit test.txt

   f) Type something, save the file and exit the editor

   g) A few other useful Linux commands are:

Command	**Utility**
`pwd`	Show path of current directory
`mkdir` *name*	Create directory *name*
`cd` *name*	Go into directory *name*
`gedit`	Start the text editor
`ls`	List all files and directories in current directory
`ls -l`	Include additional information in the listing
`cp`, `mv`	Copy/move files
`man` *command*	Open the manual of a *command*
command `-h`	Get help on the syntax of a *command*
`g ++`	Compile C and C++ programs
`exit`	Exit the terminal
`tar -xvzf` *name*.tar.gz	Uncompress file with extension tar.gz
`tar -cvzf` *name*.tar.gz	Create zipped file with extension tar.gz

2. **Introduction to C++** | The goal of this task is to write a first C++ program.

   a) Open a terminal

   b) Create a new directory called *hello*

   c) Go into the directory

   d) Use a text editor to open a C++ file called `hello.cpp`

e) Type the following code:

```
#include <iostream>
using namespace std;
int main(){
   cout<<"Hello World"<<endl;
   return 0;
}
```

f) Compile the code: `g++ hello.cpp -o hello.x`

g) Run the code: `./hello.x`

h) Change the output text, compile and run the program again

3. **Another C++ example** | The goal is to write a code that generates a table with the values given by a parabola.

   a) Open a file parabola.cpp and write the following code:

```
#include <iostream>
using namespace std;
int main(){
   for(int i = 1; i<=10; i++){
      double y = i*i; // Create new variable
      cout<<i<<"\t"<<y<<endl;
   }
   return 0;
}
```

   b) Run the program saving the output to a file *parabola.dat*: `./parabola.x > parabola.dat`

4. **Simple arrays** | Implement a program that defines an array with the following values

   $\{10.5, 9.3, 11.4, 10.9, 13, 8.4, 9.2, 8.9, 10.3, 11.2, 12.1, 8.4, 9.2, 9.9, 10.1\}$

   The program should run over all values and print them to the screen. Then it should ask the user to enter a number between 1 and 15 and print the corresponding number of the array.

5. **Calculate mean values and standard deviation** | Change the program you wrote on the previous exercise to calculate the following quantities:

   a) Mean value of the numbers in the array

$$< x >= \frac{1}{N}\sum_{i=1}^{N} x_i \tag{1}$$

b) Standard deviation

$$\sigma = \sqrt{var}, \qquad var = \frac{1}{N}\sum_{i=1}^{N}(x_i - < x >)^2 \tag{2}$$

6. **Conditional statements** | Using the same program as in the previous exercises, define the following array

   $\{1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1, 0, 1, 0, 1\}$

   Loop over the entries of the array and whenever you find an entry with the value 1 print the corresponding entry of the initial array. Then for all entries marked with 0 (or 1) calculate the mean value and the standard deviation.

If you do not have acceess to the LIP machines

1. On Linux, or in a Linux terminal emulator, check if you have a `g++` compiler installed: type `dpkg --list | grep compiler` and check if `g++` shows up on the list. If yes, you are all set and can continue with the tutorial.

2. If not, install the `g++` compiler: type `sudo apt install g++` and enter your password (should be the same you use when login in). Check if it was correctly installed by following the instructions on the previous point.