Environment setup - Windows 10

Hi guys, welcome to the 2020 edition of the *Bioinformatics* course! During the next lessons, you will learn what *bioinformatics* is about, and typical problems and tools will be introduced to you. As long as during labs, you will go deeper into the usage and development of *bioinformatics* pipelines, a suitable working environment has to be set up on your PC. In the next sections, detailed explanations are provided about how to make Windows 10 systems ready for the course. If you have any trouble feel free to write an email to `emanuele.parisi@unibo.it` for receiving additional help.

Prepare for *Python* 3 programming

Python is an interpreted, general-purpose and high-level programming language widely adopted both in academia and enterprises. It is the official programming language we will use to develop scripts and pipelines during the course, so you need to have a working *Python* 3 installation on your PC.

Install the *Python* 3 interpreter

Windows users can download the *Python* 3.8.2 installer from the official *Python* FTP server. Once it is downloaded, execute it and make sure the *"Add Python 3.8 to PATH"* box is checked. Then, click on "*Install Now*" and follow the instructions until the setup process is over. Once the installation is completed, you should be able to run the *"Python 3.8 (64-bit)"* application from the menu of Windows.

Install PyCharm

PyCharm is an integrated development environment for *Python* programming. It provides useful features such as smart code completion, code inspections, on-the-fly error highlighting and automated code refactoring. To obtain the professional version of the software, you can register on the JetBrains account website using your PoliTO email. Then, install PyCharm Professional following the official instructions JetBrains provides on this page.

Install a Unix-like environment

During lab sessions, you will be asked to run *bioinformatics* tools such as aligners or gene fusion detection pipelines which require a Unix-like environment to be used. Windows 10 users can emulate a Linux environment easily, installing the *"Ubuntu 18.04 LTS"* app from the Windows store. You can find the application on this page along with install instructions. Once the app is correctly installed, run it immediately to complete its configuration.

Get *bioinformatics* tools

Conda is an open-source, cross-platform package manager and environment management system popular among *Python* data scientists. We will use a minimal *Conda* distribution, called *Miniconda*, to quickly access *bioinformatics* tools and libraries required during labs.

Install *Conda*

Even if *Conda* is cross-platform, some of the tools you will use during laboratories are not. That's why we will install *Miniconda* inside the *"Ubuntu 18.04 LTS"* app you have just configured. First of all, open *"Ubuntu 18.04 LTS"*. Then, download, make executable and run the *Miniconda* installer with the following commands:

```
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
chmod +x Miniconda3-latest-Linux-x86_64.sh
./Miniconda3-latest-Linux-x86_64.sh
```

Follow the instructions on the screen accepting the Conda license and keeping the default *Conda* installation path the installer. When the install procedure is over disable the automatic activation of the base environment on login running:

```
conda config --set auto_activate_base false
```

For testing your *Miniconda* installation, exit the terminal and open it again. Then, check if running the command `conda --version` correctly print the *Conda* version installed. If so, *Miniconda* was successfully installed on your system.

Initialize a virtual environment

Virtual environments are useful to avoid conflicts when managing software dependencies. We will use them for installing bioinformatics tools quickly and avoiding collisions with libraries already installed on your system. Open the "*Ubuntu 18.04 LTS*" you installed *Miniconda* into, then run the following commands:

```
conda create --name bioinformatics2020
conda activate bioinformatics2020
conda install -c bioconda bwa samtools bcftools
conda deactivate
```

If no errors are printed on the screen, then your virtual environment is correctly installed. Remember that whenever you want to work with software installed in your virtual environments, you have to activate it with:

```
conda activate bioinformatics2020
```

When they are no longer needed you can deactivate the virtual environment with:

```
conda deactivate
```

At this point, if you were able to correctly run all the previous steps without errors, your system is ready for *bioinformatics* laboratories. Good job!