Linux Software Installation – Exercises 2

Part 1. Install PYTHON software with PIP

1.1 Login to the BioHPC machine and install deepTools;

Login (ssh) to the machine that you are assigned for this workshop (assigned machines: https://biohpc.cornell.edu/ww/machines.aspx?i=117).

```
pip install deepTools --user
```

1.2 Check which version of numy and deeptools;

First, start python by command “python”. Check version and file path by typing following commands at “python” prompt “>>>”:

```
import numpy

numpy.__file__

numpy.__version__
```

```
import pysam

pysam.__file__

pysam.__version__
```

```
import deeptools

deeptools.__file__
```

After you are done, press “Ctrl-D” to exit python prompt.

- It is double-underline in “__version__” .
- Check from which files each modules are loaded from;
- The deeptools module does not have the “version” property;
- An old version of pysam could cause deeptools to fail, in that case you will need to update the pysam by “--upgrade” option;

1.3 (This step and step 1.4 is optional. It shows you how to run pip in a different way. It could take several minutes to finish) Install deeptools to a specific directory by “--install-option” parameter (replace xxxxx with your user ID)

```
mkdir ~/myPythonLib


pip install deepTools \

--install-option="--prefix=/home/xxxxx/myPythonLib" --ignore-installed
```

1.4 Now check which numpy and deeptools modules are used

```
export PYTHONPATH=~/myPythonLib/lib64/python2.7/site-packages:~/myPythonLib/lib64/python2.7/site-packages
```

Start python by command “python”. Check version and file path by typing following commands at “python” prompt “>>>”:

```
import numpy

numpy.__file__

numpy.__version__

import deeptools

deeptools.__file__
```

To delete the installation, you can delete the whole directory:

```
rm -fr ~/myPythonLib
```

Part 2. Install PERL software with CPAN

2.1 Configure CPAN

```
mkdir ~/perl
export PERL5LIB=~/perl/lib/perl5
cpan
```

At cpan pompt “cpan[1]>”, type the following commands:

```
o conf makepl_arg INSTALL_BASE=~/perl
o conf mbuild_arg INSTALL_BASE=~/perl
o conf prefs_dir ~/perl /prefs
o conf commit
```

2.2 Install PERL modules with CPAN

Still at cpan pompt , type the following command to install PERL module:

```
install XML::Simple
```

This module is only accessible if you “export PERL5LIB=~/perl/lib/perl5”. Delete the whole directory if installation went wrong.

Part 3. Install C software

Normally, the software web site or the README (sometimes INSTALL) file in the source code directory provides step-by-step instruction. You can follow the instruction. As a non-root user, quite often you need to modify the instruction, e.g. adding “--prefix=~/mydirectory” at the configuring step.

3.1 Download the source code (replace xxxxx with your user ID)

```
cd /workdir/xxxxx

wget http://catchenlab.life.illinois.edu/stacks/source/stacks-
2.0b.tar.gz

tar xvfz stacks-2.0b.tar.gz
```

- If the source code is available from github, use "git clone" to get the latest source code. E.g. "git clone https://github.com/lh3/bwa.git"
- If the source code you downloaded is a *tar.bz2 file, use "tar xvfj myfile.tar.bz2" command to decompress.

3.2 Configure (replace xxxxx with your user ID)

```
mkdir ~/stacks

cd stacks-2.0b

export LD_LIBRARY_PATH=/usr/local/gcc-7.3.0/lib64:/usr/local/gcc-7.3.0/lib


export PATH=/usr/local/gcc-7.3.0/bin:$PATH

./configure --prefix=/home/xxxxx/stacks
```

- In the directory of source code, you will find an INSTALL file with step-by-step instruction how to install.
- The default gcc compiler would not work with stacks v2. If you use the default gcc, you will get an error message complaining your gcc is too old. The two commands here "export LD_LIBRARY_PATH.." "export PATH…" switch the default gcc to v7.3.0.
- The "--prefix=" parameter instruct the software to be installed in your home directory.
- After this step, a new Makerfile is created with instructions to compiler how to compile and install the software. The Makerfile is a text file. If you want, you can use any text editor to examine the file, you will find lines like "prefix = /home/xxxxx/stacks".

3.3 Compile (this step could take many minutes)

```
make
```

- After this step, several binary executable file is created

3.4 Install

```
make install
```

- After this step, you will find binary executables and libraries in your installation directory: /home/xxxxx/stacks. To run the code, you can do “export PATH=/home/xxxxx/stacks/bin:$PATH”, and run the command “sstacks”.

Part 4. Install R package

First, start R by command “R”. If you want to install R package to a different version of R, you need to modify PATH, e.g. “export PATH=/programs/R-3.4.2s/bin:$PATH”.

At R prompt “>”, type the following command to install R package “qtl”

```
install.packages("qtl")
```

Still at R prompt, load qtl package, get the version and physical path of the package

```
library(qtl)

packageVersion("qtl")

find.package("qtl")
```