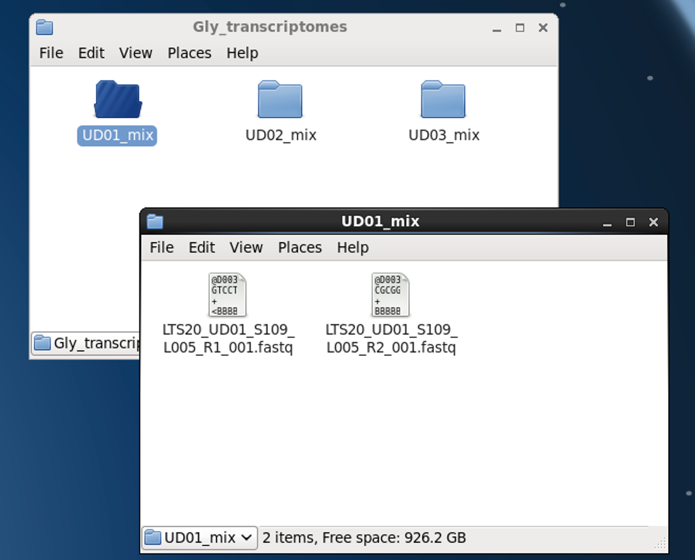
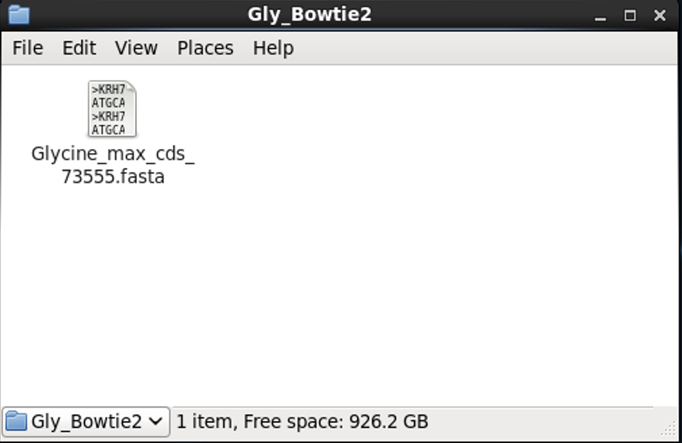
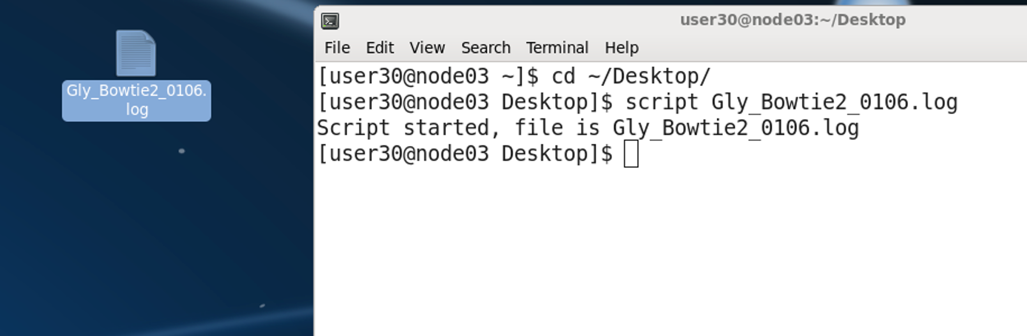
**SOP of Running Bowtie2BuildAndCal.R**

1. **Preparing transcriptome data and cds.fasta for building Bowtie2 index.**
2. Saving all transcriptomes in a folder. For example: here is the folder “Gly\_transcriptomes”.
3. Create a file with cds.fasta (without descriptions) for building Bowtie2 index. For example, here is “Gly\_Bowtie2”.
4. **Recording print out information when running Bowtie2**
5. Create a log file.

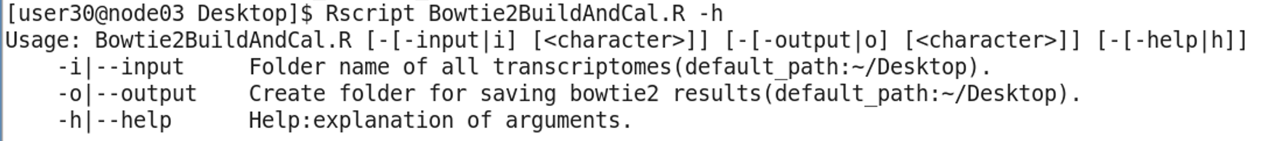
* Code: script [file\_name].log



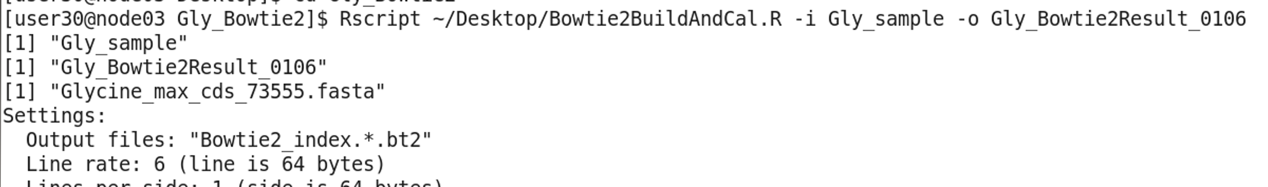
1. **Running Bowtie2BuildAndCal.R**
2. Go to the file which is going to build Bowtie2 index.

* Code: cd [path of filename]

1. See the usage and argument of Bowtie2BuildAndCal.R

* Code: Rscript [path of Bowtie2BuidAndCal.R] -h

1. Start running Bowtie2BuildAndCal.R

* Code: Rscript [path of Bowtie2BuildAndCal.R] -i [folder name of transcriptome data] -o [folder name of Bowtie2 result]

1. **Finishing and checking Bowtie2 record**
2. Finishing record

* Code: exit

1. Checking Bowtie2 record, including Bowtie2 build index and total alignment rate of each sample.

* Code: more -n/UD01 Bowtie2\_Result/Gly\_Bowtie2\_record.log
  + **Code explanation:**

1. Showing the folder name of transcriptome data and the folder name which is going to saving Bowtie2 result(A).
2. Create the folder which named “Gly\_Bowtie2Result\_0106” on the Desktop for saving Bowtie2 result(B).
3. Showing the CDS fasta file name it catches (C) and starting to build index for calculation (D).
4. Grab all transcriptome samples’ names (E).
5. In the for loop, for each sample: first, grab R1.fastaq and R2.fastaq files and save them as R1 and R2 (F); second, show the id of each samples (G); third, run Bowtie2 by put in variables including index name, cds, id, R1, R2 (H); fourth, create a file named with its id and move the Bowtie2 result to this file (I&J) .

I & J

H

A

B

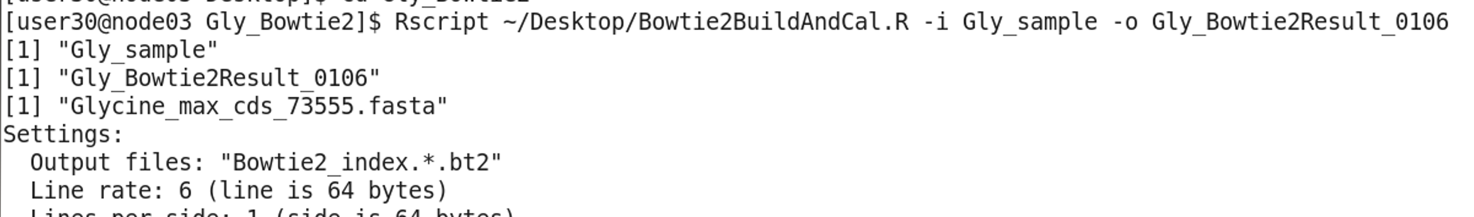
C

D

F

G

E



C

A