Practical Linux Examples

- Processing large text files
- Working with lots of files

Qi Sun
Bioinformatics Facility
Cornell University

Estimate the percentage of sequencing reads in the FASTQ file that contains the adapter “AGATCGGAAGAGC”.

- **Read file:**

  cat D7RACXX.fastq

- **Select lines that contain the string “AGATCGGAAGAGC”**

  grep AGATCGGAAGAGC

- **Count the selected lines**

  wc -l

```
4=DFFFFHHHHHJJJJJJJJJJJJJJGHIJFGIJEHIGEEEFDBBBBDBBBDBBDDDD@DDBDDDDDDDDDDDBBDDCDDDDDDDBDDDDDDDDBDDACC
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
CCAGTCTGCGTCCGCCTAGGCCCGGTCGACGAGCCTGGGGCCAAGGTGGCGTACCAGACCAGCGTCAGCATCACCGGCCACGTCTTCAAGGGCATCCTG
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
4=DFFFFGHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIHHHHFFCCCCC@CBCCBCCCCCCCCCCCCBBCCCCCCCCCCCCCCCCCCCCECCCCCCCCCCC
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
TACGGACAGCTCGGTGCACTGCTTCAAAAACTGCCGGATGTAGTCCCACAGGGATTCCCCTGGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAG
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
1=DDFFFHHHHHJJGHJJJJJJJJJJJJJIJJJJJJJIJJIJJIIIJIJIJJHHHHFFFFFDDEEDDDDDDDDDDDDDDDDDDCDDDDD@BCDDDDD@
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
TATTACAGCAGGCGCGCCGAGAGAGAGTGGAGAGGAGAGGACGGTGGCGATCGAGCGAGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCG
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
1=DFFFFGHHHHJJJJJJJGIGIFHFH=BGGHIIJGHGHHGHFDCDDDDDDDDBDDDDDDDDDDDDDDDDDDDDBDBDDDDDDDDDDDDDDDDDDDBDB
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
CGGATATCTGCGACAGCTTGCTGTAGTACAGTACAGGTACAGTACGTACGTTGCAGTAGAAGCAATCCATGCATAAATACGTCTGCCACACCGGAGGAG
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
1=DDDDDHHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIGEGGIIIIIIIIIIIIIIIIIHIIIIIIIIHHHHHFEEEEEEEECCCBCBCCCCBBBBBBBBBB
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
AGGACAGCCGGACTTAATGATGAAGAGATGGCTCTCATCATCAAGAGATTCAAGACGGCGCTAAAAGGTCACAAGGGGCAGAGATCGGAAGAGCGGTTC
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
1=DDDDDHHDHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIGIEEIIIIECCBBCCCBCCCCCCBBBBBBBBBCCCBB?BBBBCBBBBB
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
AGGACTGCGGCCGCCGTGAGCGTCATCAGATCGCCGGGGCTTCTGACGCCAACCGCGGCCTGCACGGCGGCGGCCACCTGGTCGTGGCGCGCCCCCGCG
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
1=DFFFFHHAHHIIIIIIIIIIIIIIIIIIIIIGIIIIHFFCCCCCCCECCCCBCBCCCCCCCCCCCCCBBCCBBBB>5@BACCACB@BBBB>>BBCCB9>B
SRR836349.18 HWI-ST1085:39:D0D7RACXX:1:1101:5308:1997 length=100
TAGCCAGCGACCGCTACGTGGCAGTCGTCGTGAGGACACCTAATCCTCAATTCGCATAAATATTCACCTACACCCCTTGCGTCCAAGGTATTTCGTCTT
SRR836349.18 HWI-ST1085:39:D0D7RACXX:1:1101:5308:1997 length=100
4=DDFFFHHHHHJJJJJJIJJJJJJJJJJIHIGIIGGIJHGHHFFFFFFEEEEDDDDDDDEEEFDEDDDDDACCBBDDDDDDDDDDDDCCEEEEDDBD?
SRR836349.19 HWI-ST1085:39:D0D7RACXX:1:1101:5732:1994 length=100
ATCTCTGCTTGATTAGAAAAAGGAGATCCACGGCCTGTGAGTAACTACGCCATGTTACGAATCACAGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGA
SRR836349.19 HWI-ST1085:39:D0D7RACXX:1:1101:5732:1994 length=100
1=DDFFFHHCACFHIJJJJJJJJJJJJJJJJJJJJJJJIJIIFHIIJIJJJJJJJJJJJJHHHGFFFFFEEEEDDDDDDDDDDDDDDDDDDDDDDDDD@
SRR836349.20 HWI-ST1085:39:D0D7RACXX:1:1101:5623:1995 length=100
TCCTGGACTGCACCCTTAGGCGACTTCTGCGCGGATAGTCTTCGCGATAGACATTGGATTTTCTCGCACTTGTTGTGCTTAGTCCGGCTGAGATCGGAA
SRR836349.20 HWI-ST1085:39:D0D7RACXX:1:1101:5623:1995 length=100
4=DDDDDHHBHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIHEECCCCCCCFFCCCCCCECCBBBBBCCCCBCBCCCCCCCCBBBBBA@:@CBBBB
SRR836349.21 HWI-ST1085:39:D0D7RACXX:1:1101:6484:1991 length=100
GAAGACTGCGACTTACCGTGTTGGAGTTGTGTAGCCACCGTGAAACACCACACTGCGCGCCGCCGCCACCACCAAAAGAGAGTGTTTCCGCCGCCGTGG
SRR836349.21 HWI-ST1085:39:D0D7RACXX:1:1101:6484:1991 length=100
4=DDFFFGHHGHJJIJJJJJJJJJJIJJIJJJJJIJJJJJJJJJJHIIJJJIJJIJHHFDDDDDDDDDDDDDDDDDDDDDDDD:ACEDDDDDDDDDDBDD
SRR836349.22 HWI-ST1085:39:D0D7RACXX:1:1101:6274:1992 length=100
ATATGCCAGCTAATCGTGTGGCTTTAGAAGCCTGTGTACAAGCTCGTAACGAAGGGCGCGATCTTGCTCGTGAAGGTAATGAAATTATCAAAGCAGAGA
SRR836349.22 HWI-ST1085:39:D0D7RACXX:1:1101:6274:1992 length=100
1=DDFFFHHHHHIJJJJJIJJJJJJFIIJJJJJJGIGIHIIJJJJJIJJJJIJIJJIHHFDDDDDDDDDD@ACBCC:ACDCDDDDDDDADDDDDCCCDC
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
GCTGATCAGCTCCCCGTTGTCGCCGGCGACGACCTGGCGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACGGATCTCGTATGCCGTGTTCTG
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
11BDDDDAFBFHHGIIAFGEFHGGHI<D<FG8',5?B(998;8@:@@BB7258<9:@5-2(:3+8?0(44::>&099(0&09CC@##############
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
CACCCAGCTTGAATCGAGGATTCACCCAACCACACTCTCAAAGAACCAACCAACACAGCCTACAATCAATCGAGGAGACCTCGAAGGGAGTAGGAGCAC
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
4=DDFFFHHHHHJJJJJJJJJJIJJJJIJIIIGHIJGIGIIJJIJJGHIJJJIIIJJJEHHHFFFFFEDCEEDDDDDDDDDDDDDDDDDDDD8ACCDDDBB
SRR836349.25 HWI-ST1085:39:D0D7RACXX:1:1101:7402:1992 length=100
```

Estimate the percentage of sequencing reads in the FASTQ file that contains the adapter “AGATCGGAAGAGC”.

```
4=DFFFFHHHHHJJJJJJJJJJJJJJGHIJFGIJEHIGEEEFDBBBBDBBBDBBDDDD@DDBDDDDDDDDDDDBBDDCDDDDDDDBDDDDDDDBDDACC
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
CCAGTCTGCGTCCGCCTAGGCCCGGTCGACGAGCCTGGGGCCAAGGTGGCGTACCAGACCAGCGTCAGCATCACCGGCCACGTCTTCAAGGGCATCCTG
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
4=DFFFFGHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIHHHHFFCCCCC@CBCCBCCCCCCCCCCCCBBCCCCCCCCCCCCCCCCCCCCCECCCCCCCCCCC
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
TACGGACAGCTCGGTGCACTGCTTCAAAAACTGCCGGATGTAGTCCCACAGGGATTCCCCTGGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAG
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
1=DDFFFHHHHHJJGHJJJJJJJJJJJJJIJJJJJJJIJJIJJIIIJIJIJJHHHHFFFFFDDEEDDDDDDDDDDDDDDDDDDDCDDDDD@BCDDDDD@
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
TATTACAGCAGGCGCGCCGAGAGAGAGTGGAGAGGAGAGGACGGTGGCGATCGAGCGAGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCG
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
1=DFFFFGHHHHJJJJJJJGIGIFHFH=BGGHIIJGHGHHGHFDCDDDDDDDDBDDDDDDDDDDDDDDDDDDDDBDBDDDDDDDDDDDDDDDDDDDBDB
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
CGGATATCTGCGACAGCTTGCTGTAGTACAGTACAGGTACAGTACGTACGTTGCAGTAGAAGCAATCCATGCATAAATACGTCTGCCACACCGGAGGAG
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
1=DDDDDHHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIGEGGIIIIIIIIIIIIIIIIIHIIIIIIIIHHHHHFEEEEEEEECCCBCBCCCCBBBBBBBBBB
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
AGGACAGCCGGACTTAATGATGAAGAGATGGCTCTCATCATCAAGAGATTCAAGACGGCGCTAAAAGGTCACAAGGGGCAGAGATCGGAAGAGCGGTTC
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
1=DDDDDHHDHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIGIEEIIIIECCBBCCCBCCCCCCBBBBBBBBBCCCBB?BBBBCBBBBB
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
AGGACTGCGGCCGCCGTGAGCGTCATCAGATCGCCGGGGCTTCTGACGCCAACCGCGGCCTGCACGGCGGCGGCCACCTGGTCGTGGCGCGCCCCCGCG
                                                                                                @BBBB>>BBCCB9>B
                                                                                                AGGTATTTCGTCTT
                                                                                                DDDCCEEEEDDBD?
                                                                                                AGCAGGAATGCCGA
                                                                                                DDDDDDDDDDDDD@
                                                                                                GGCTGAGATCGGAA
                                                                                                BBBBBA@:@CBBBB
                                                                                                TTCCGCCGCCGTGG
                                                                                                EDDDDDDDDDDBDD
                                                                                                GAAATTATCAAAGCAGAGA
SRR836349.22 HWI-ST1085:39:D0D7RACXX:1:1101:6274:1992 length=100
1=DDFFFHHHHHIJJJJJIJJJJJJFIIJJJJJJGIGIHIIJJJJJIJJJJIJIJJIHHFDDDDDDDDDD@ACBCC:ACDCDDDDDDDADDDDDCCCDC
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
GCTGATCAGCTCCCCGTTGTCGCCGGCGACGACCTGGCGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACGGATCTCGTATGCCGTGTTCTG
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
11BDDDDAFBFHHGIIAFGEFHGGHI<D<FG8',5?B(998;8@:@@BB7258<9:@5-2(:3+8?0(44::>&099(0&09CC@##############
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
CACCCAGCTTGAATCGAGGATTCACCCAACCACACTCTCAAAGAACCAACCAACACAGCCTACAATCAATCGAGGAGACCTCGAAGGGAGTAGGAGCAC
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
4=DDFFFHHHHHJJJJJJJJJJIJJJJIJIIGHIJGIGIIJJIJJGHIJJJIIIJJJEHHHFFFFFEDCEEDDDDDDDDDDDDDDDDDDDD8ACCDDDBB
SRR836349.25 HWI-ST1085:39:D0D7RACXX:1:1101:7402:1992 length=100
```

```
cat D7RACXX.fastq | grep AGATCGGAAGAGC | wc -l
```

Estimate the percentage of sequencing reads in the FASTQ file that contains the adapter “AGATCGGAAGAGC”.

```
4=DFFFFHHHHHJJJJJJJJJJJJJJGHIJFGIJEHIGEEEFDBBBBDBBBDBBDDDD@DDBDDDDDDDDDDDBBDDCDDDDDDDBDDDDDDDBDDACC
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
CCAGTCTGCGTCCGCCTAGGCCCGGTCGACGAGCCTGGGGCCAAGGTGGCGTACCAGACCAGCGTCAGCATCACCGGCCACGTCTTCAAGGGCATCCTG
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
4=DFFFFGHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIHHHHFFCCCCC@CBCCBCCCCCCCCCCCCBBCCCCCCCCCCCCCCCCCCCCCECCCCCCCCCCC
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
TACGGACAGCTCGGTGCACTGCTTCAAAAACTGCCGGATGTAGTCCCACAGGGATTCCCCTGGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAG
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
1=DDFFFHHHHHJJGHJJJJJJJJJJJJJIJJJJJJJIJJIJJIIIJIJIJJHHHHFFFFFDDEEDDDDDDDDDDDDDDDDDDDCDDDDD@BCDDDDD@
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
TATTACAGCAGGCGCGCCGAGAGAGAGTGGAGAGGAGAGGACGGTGGCGATCGAGCGAGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCG
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
1=DFFFFGHHHHJJJJJJJGIGIFHFH=BGGHIIJGHGHHGHFDCDDDDDDDDBDDDDDDDDDDDDDDDDDDDDDBDBDDDDDDDDDDDDDDDDDDDBDB
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
CGGATATCTGCGACAGCTTGCTGTAGTACAGTACAGGTACAGTACGTACGTTGCAGTAGAAGCAATCCATGCATAAATACGTCTGCCACACCGGAGGAG
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
1=DDDDDHHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIGEGGIIIIIIIIIIIIIIIIIHIIIIIIIIIHHHHHFEEEEEEEECCCBCBCCCCBBBBBBBBBB
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
AGGACAGCCGGACTTAATGATGAAGAGATGGCTCTCATCATCAAGAGATTCAAGACGGCGCTAAAAGGTCACAAGGGGCAGAGATCGGAAGAGCGGTTC
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
1=DDDDDHHDHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIGIEEIIIIECCBBCCCBCCCCCCBBBBBBBBBCCCBB?BBBBCBBBBB
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
AGGACTGCGGCCGCCGTGAGCGTCATCAGATCGCCGGGGCTTCTGACGCCAACCGCGGCCTGCACGGCGGCGGCCACCTGGTCGTGGCGCGCCCCCGCG
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
                                                                                                BACCACB@BBBB>>BBCCB9>B
                                                                                                CGTCCAAGGTATTTCGTCTT
                                                                                                DDDDDDDDDCCEEEEDDBD?
                                                                                                CGGTTCAGCAGGAATGCCGA
                                                                                                DDDDDDDDDDDDDDDDDDD@
                                                                                                TAGTCCGGCTGAGATCGGAA
                                                                                                CCCCCCBBBBBA@:@CBBBB
                                                                                                GAGTGTTTCCGCCGCCGTGG
                                                                                                DDD:ACEDDDDDDDDDDBDD
                                                                                                AATGAAATTATCAAAGCAGAGA
SRR836349.22 HWI-ST1085:39:D0D7RACXX:1:1101:6274:1992 length=100
1=DDFFFHHHHHIJJJJJIJJJJJJFIIJJJJJJGIGIHIIJJJJJIJJJJIJIJJIHHFDDDDDDDDDD@ACBCC:ACDCDDDDDDDADDDDDCCCDC
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
GCTGATCAGCTCCCCGTTGTCGCCGGCGACGACCTGGCGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACGGATCTCGTATGCCGTGTTCTG
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
11BDDDDAFBFHHGIIAFGEFHGGHI<D<FG8',5?B(998;8@:@@BB7258<9:@5-2(:3+8?0(44::>&099(0&09CC@##############
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
CACCCAGCTTGAATCGAGGATTCACCCAACCACACTCTCAAAGAACCAACCAACACAGCCTACAATCAATCGAGGAGACCTCGAAGGGAGTAGGAGCAC
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
4=DDFFFHHHHHJJJJJJJJJJIJJJJIJIIGHIJGIGIIJJIJJGHIJJJIIIJJJEHHHFFFFFEDCEEDDDDDDDDDDDDDDDDDDD8ACCDDDBB
SRR836349.25 HWI-ST1085:39:D0D7RACXX:1:1101:7402:1992 length=100
```

```
cat DeD7RACXX.fastq | head -n 40000 | grep AGATCGGAAGAGC | wc -l
```

To get a quick answer, you can estimate the percentage based on the first 40,000 lines

```
cat DeD7RACXX.fastq | head -n 40000 | grep AGATCGGAAGAGC
```

VS

```
cat DeD7RACXX.fastq | grep AGATCGGAAGAGC | head -n 40000
```

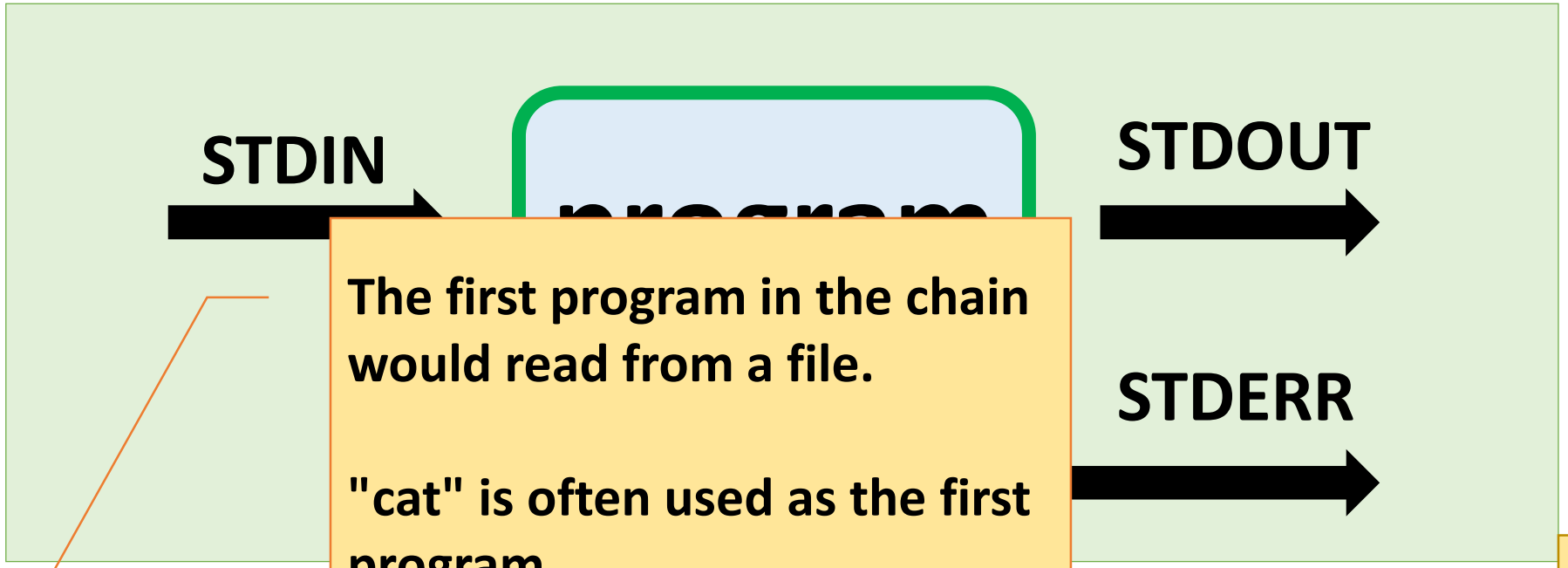
Estimate the percentage of sequencing reads in the FASTQ file that contains the adapter “AGATCGGAAGAGC”.

```
cat D7RACXX.fastq | \
head -n 40000 | \
grep AGATCGGAAGAGC | \
wc -l
```

Use “\” to separate the command into multiple lines. Make sure no space after “\”

```
4=DFFFFHHHHHJJJJJJJJJJJJJJGHIJFGIJEHIGEEEFDBBBBDBBBDBBDDDD@DDBDDDDDDDDDDDBBDDCDDDDDDDBDDDDDDDDBDDACC
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
CCAGTCTGCGTCCGCCTAGGCCCGGTCGACGAGCCTGGGGCCAAGGTGGCGTACCAGACCAGCGTCAGCATCACCGGCCACGTCTTCAAGGGCATCCTG
SRR836349.12 HWI-ST1085:39:D0D7RACXX:1:1101:4020:1994 length=100
4=DFFFFGHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIHHHHFFCCCCC@CBCCBCCCCCCCCCCCCBBCCCCCCCCCCCCCCCCCCCCECCCCCCCCCCC
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
TACGGACAGCTCGGTGCACTGCTTCAAAAACTGCCGGATGTAGTCCCACAGGGATTCCCCTGGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAG
SRR836349.13 HWI-ST1085:39:D0D7RACXX:1:1101:4438:1991 length=100
1=DDFFFHHHHHJJGHJJJJJJJJJJJJJIJJJJJJJIJJIJJIIIJIJIJJHHHHFFFFFDDEEDDDDDDDDDDDDDDDDDDCDDDDD@BCDDDDD@
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
TATTACAGCAGGCGCGCCGAGAGAGAGTGGAGAGGAGAGGACGGTGGCGATCGAGCGAGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACCG
SRR836349.14 HWI-ST1085:39:D0D7RACXX:1:1101:4582:1998 length=100
1=DFFFFGHHHHJJJJJJJGIGIFHFH=BGGHIIJGHGHHGHFDCDDDDDDDDBDDDDDDDDDDDDDDDDDDDDBDBDDDDDDDDDDDDDDDDDDDBDB
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
CGGATATCTGCGACAGCTTGCTGTAGTACAGTACAGGTACAGTACGTACGTTGCAGTAGAAGCAATCCATGCATAAATACGTCTGCCACACCGGAGGAG
SRR836349.15 HWI-ST1085:39:D0D7RACXX:1:1101:4981:1992 length=100
1=DDDDDHHAHHIIIIIIIIIIIIIIIIIIIIIIIIIIIGEGGIIIIIIIIIIIIIIIIIHIIIIIIIIHHHHHFEEEEEEEECCCBCBCCCCBBBBBBBBBB
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
AGGACAGCCGGACTTAATGATGAAGAGATGGCTCTCATCATCAAGAGATTCAAGACGGCGCTAAAAGGTCACAAGGGGCAGAGATCGGAAGAGCGGTTC
SRR836349.16 HWI-ST1085:39:D0D7RACXX:1:1101:4834:1996 length=100
1=DDDDDHHDHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIGIEEIIIIECCBBCCCBCCCCCCBBBBBBBBBCCCBB?BBBBCBBBBB
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
AGGACTGCGGCCGCCGTGAGCGTCATCAGATCGCCGGGGCTTCTGACGCCAACCGCGGCCTGCACGGCGGCGGCCACCTGGTCGTGGCGCGCCCCCGCG
SRR836349.17 HWI-ST1085:39:D0D7RACXX:1:1101:5373:1997 length=100
1=DFFFFHHAHHIIIIIIIIIIIIIIIIIIIIIGIIIIHFFCCCCCCCECCCCBCBCCCCCCCCCCCCCBBCCBBBB>5@BACCACB@BBBB>>BBCCB9>B
SRR836349.18 HWI-ST1085:39:D0D7RACXX:1:1101:5308:1997 length=100
TAGCCAGCGACCGCTACGTGGCAGTCGTCGTGAGGACACCTAATCCTCAATTCGCATAAATATTCACCTACACCCCTTGCGTCCAAGGTATTTCGTCTT
SRR836349.18 HWI-ST1085:39:D0D7RACXX:1:1101:5308:1997 length=100
4=DDFFFHHHHHJJJJJJIJJJJJJJJJJIHIGIIGGIJHGHHFFFFFFEEEEDDDDDDDEEEFDEDDDDDACCBBDDDDDDDDDDDDCCEEEEDDBD?
SRR836349.19 HWI-ST1085:39:D0D7RACXX:1:1101:5732:1994 length=100
ATCTCTGCTTGATTAGAAAAAGGAGATCCACGGCCTGTGAGTAACTACGCCATGTTACGAATCACAGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGA
SRR836349.19 HWI-ST1085:39:D0D7RACXX:1:1101:5732:1994 length=100
1=DDFFFHHCACFHIJJJJJJJJJJJJJJJJJJJJJJJIJIIFHIIJIJJJJJJJJJJJHHHGFFFFFEEEEDDDDDDDDDDDDDDDDDDDDDDDD@
SRR836349.20 HWI-ST1085:39:D0D7RACXX:1:1101:5623:1995 length=100
TCCTGGACTGCACCCTTAGGCGACTTCTGCGCGGATAGTCTTCGCGATAGACATTGGATTTTCTCGCACTTGTTGTGCTTAGTCCGGCTGAGATCGGAA
SRR836349.20 HWI-ST1085:39:D0D7RACXX:1:1101:5623:1995 length=100
4=DDDDDHHBHHIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIHEECCCCCCCFFCCCCCCECCBBBBBCCCCBCBCCCCCCCCBBBBBA@:@CBBBB
SRR836349.21 HWI-ST1085:39:D0D7RACXX:1:1101:6484:1991 length=100
GAAGACTGCGACTTACCGTGTTGGAGTTGTGTAGCCACCGTGAAACACCACACTGCGCGCCGCCGCCACCACCAAAAGAGAGTGTTTCCGCCGCCGTGG
SRR836349.21 HWI-ST1085:39:D0D7RACXX:1:1101:6484:1991 length=100
4=DDFFFGHHGHJJIJJJJJJJJJIJJIJJJJJIJJJJJJJJJJHIIJJJIJJIJHHFDDDDDDDDDDDDDDDDDDDDDDDD:ACEDDDDDDDDDDBDD
SRR836349.22 HWI-ST1085:39:D0D7RACXX:1:1101:6274:1992 length=100
ATATGCCAGCTAATCGTGTGGCTTTAGAAGCCTGTGTACAAGCTCGTAACGAAGGGCGCGATCTTGCTCGTGAAGGTAATGAAATTATCAAAGCAGAGA
SRR836349.22 HWI-ST1085:39:D0D7RACXX:1:1101:6274:1992 length=100
1=DDFFFHHHHHIJJJJJIJJJJJJFIIJJJJJJGIGIHIIJJJJJIJJJJIJIJJIHHFDDDDDDDDDD@ACBCC:ACDCDDDDDDDADDDDDCCCDC
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
GCTGATCAGCTCCCCGTTGTCGCCGGCGACGACCTGGCGCTGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACGGATCTCGTATGCCGTGTTCTG
SRR836349.23 HWI-ST1085:39:D0D7RACXX:1:1101:6437:1995 length=100
11BDDDDAFBFHHGIIAFGEFHGGHI<D<FG8',5?B(998;8@:@@BB7258<9:@5-2(:3+8?0(44::>&099(0&09CC@##############
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
CACCCAGCTTGAATCGAGGATTCACCCAACCACACTCTCAAAGAACCAACCAACACAGCCTACAATCAATCGAGGAGACCTCGAAGGGAGTAGGAGCAC
SRR836349.24 HWI-ST1085:39:D0D7RACXX:1:1101:7042:1998 length=100
4=DDFFFHHHHHJJJJJJJJJJIJJJJIJIIIGHIJGIGIIJJIJJGHIJJJIIIJJJEHHHFFFFFEDCEEDDDDDDDDDDDDDDDDDDDD8ACCDDDBB
SRR836349.25 HWI-ST1085:39:D0D7RACXX:1:1101:7402:1992 length=100
```

Three streams for a standard Linux program



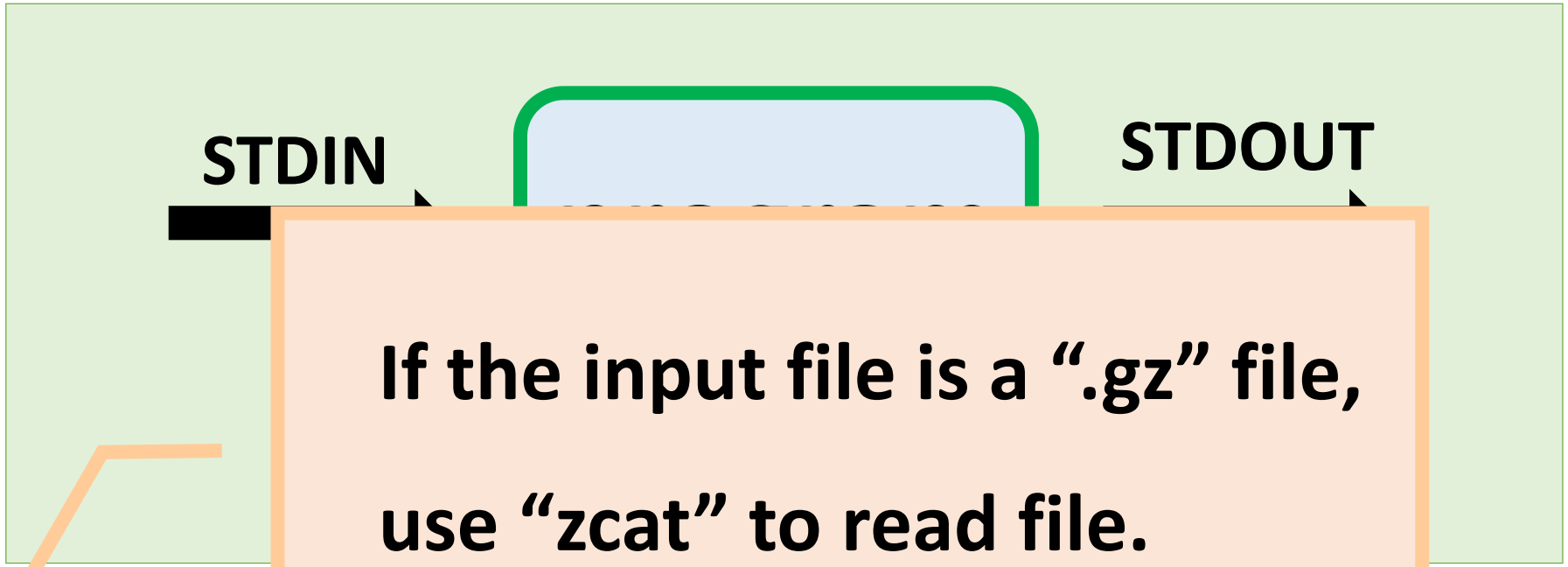
The first program in the chain would read from a file.

"cat" is often used as the first program.

Write to file

One liner

Three streams for a standard Linux program



If the input file is a “.gz” file,

use “zcat” to read file.

One line:

grep

Search for a pattern and output matched lines

$ cat mydata.txt

```
AAGATCAAAAAAGA
ATTTACGAAAAAAGA
ACCTGTTGGATCCAAAGTT
AAACTTTCGACGATCT
ATTTTTTTAGAAAGG
```

$ cat mydata.txt | grep '[AC]GATC'

```
AAGATCAAAAAAGA
AAACTTTCGACGATCT
```

These two commands are same

```
cat mydata.txt | grep 'GATC'
```

=

```
grep 'GATC' mydata.txt
```

* Functions like sort, wc, gunzip, head, tail, uniq can process either a file or data from STDIN.

grep -v

Filter out certain lines

$ cat mydata.txt

```
#seq1
AAGATCAAAAAAGA
#seq2
ACCTGTTGGATCCAAAGTT
```

$ grep -v ‘^#'

```
AAGATCAAAAAAGA
ACCTGTTGGATCCAAAGTT
```

- “-v” to select non-matching lines
- “^” beginning of a line; “$” end of line

wc -l Count the number of lines

$ cat mydata.txt

```
AAGATCAAAAAAGA
ATTTACGAAAAAAGA
ACCTGTTGGATCCAAAGTT
AAACTTTCGACGATCT
ATTTTTTTAGAAAGG
```

$ cat mydata.txt | grep '[AC]GATC' | wc -l

```
2
```

sort

Sort the text in a file

$ sort myChr.txt

```
Chr1
Chr10
Chr2
Chr3
Chr4
Chr5
```

$ sort -V myChr.txt

```
Chr1
Chr2
Chr3
Chr4
Chr5
Chr10
```

$ sort -n myPos.txt

```
1
2
3
4
5
10
```

sort

Sort the text by multiple columns

```
$ sort -k1,1V -k2,2n myChrPos.txt
```

```
Chr1	12
Chr1	100
Chr1	200
Chr2	34
Chr2	121
Chr2	300
```

Locale and sorting

Computer English

LC_ALL=C

Alphabetic order
A
B
C
...
X
Y
Z
a
b
c
...
x
y
z

US English *

LC_ALL=US_en

Alphabetic order
a
A
b
B
c
C
...
x
X
y
Y
z
Z

* On Linux, US English locale sorting also ignore the non-alphanumerical characters. This could cause problems.

Use this Linux command to find out the locale setting on your server:

locale

Locale setting on BioHPC

Use the “locale” command to check locale setting:

```
[qisun@cbsum1c2b010 ~]$ locale
LANG=en_US.UTF-8
LC_CTYPE="C"
LC_NUMERIC="C"
LC_TIME="C"
LC_COLLATE="C"
LC_MONETARY="C"
LC_MESSAGES="C"
LC_PAPER="C"
LC_NAME="C"
LC_ADDRESS="C"
LC_TELEPHONE="C"
LC_MEASUREMENT="C"
LC_IDENTIFICATION="C"
LC_ALL=C
```

If you see errors like this when running software:

RuntimeError:

Python 3 was configured to **use ASCII as encoding** for the environment. Consult https://click.palletsprojects.com/en/7.x/python3/ for mitigation steps.

You need to change the locale:

```
export LC_ALL=en_US.utf-8
export LANG=en_US.utf-8
```

Some extra parameter to set for the “sort” command

LC_ALL=C sort -S 4G -k1,1 myChr.txt

LC_ALL=C vs LC_ALL=US_en

LC_ALL=C	LC_ALL=US_en
AA	aa
BB	a.a
a.a	AA
aa	bb
bb	BB

* BioHPC default locale is C. So you can skip this parameter

-S 4G

Use RAM as buffer 4G

uniq -c

Count the occurrence of unique tags

$ cat mydata.txt

```
ItemB
ItemA
ItemB
ItemC
ItemB
ItemC
ItemB
ItemC
```

$ cat mydata.txt | sort | uniq -c

```
1    ItemA
4    ItemB
3    ItemC
```

Mark sure to run "sort" before "uniq"

Merging files:

cat f1 f2 vs **paste** f1 f2 vs **join** f1 f2

File 1:

Item1

Item2

File2:

Item3

Item4

cat File1 File2 > mergedfile1

Item1
Item2
Item3
Item4

paste File1 File2 > mergedfile2

Item1	Item3
Item2	Item4

* Make sure that that two files has same number of rows and sorted the same way. Otherwise, use “join”

join

Merging two files that share a common field

File 1:

Gene1	DNA-binding
Gene2	kinase
Gene3	membrane

File2:

Gene2	764
Gene3	23
Gene4	34

join -1 1 -2 1 File1 File2 > mergedfile

Gene2	Kinase	764
Gene3	membrane	23

join -1 1 -2 1 -a1 File1 File2 > mergedfile

Gene1	DNA-binding	
Gene2	Kinase	764
Gene3	membrane	23

cut

Output selected columns in a table

$ cat mydata.txt

```
Chr1 1000  2250 Gene1
Chr1  3010 5340 Gene2
Chr1  7500 8460 Gene3
Chr2  8933 9500 Gene4
```

$ cat mydata.txt | cut -f 1,4

```
Chr1  Gene1
Chr1 Gene2
Chr1 Gene3
Chr2  Gene4
```

sed

Modify text in a file

$ cat mydata.txt

```
Chr1 1000  2250 Gene1
Chr1  3010 5340 Gene2
Chr1  7500 8460 Gene3
Chr2  8933 9500 Gene4
```

$ cat mydata.txt | sed "s/^Chr//"

```
1       1000 2250 Gene1
1       3010 5340 Gene2
1       7500 8460 Gene3
2       8933 9500 Gene4
```

awk

Probably the most versatile function in Linux

$ cat mydata.txt

```
Chr1 1000  2250 Gene1
Chr1  3010 5340 Gene2
Chr1  7500 8460 Gene3
Chr2  8933 9500 Gene4
```

$ cat mydata.txt |
awk '{if ($1=="Chr1") print $4}'

```
Gene1
Gene2
Gene3
```

awk vs cut

Input data file: mydata.txt

Chr1	1000	2250	Gene1
Chr1	3010	5340	Gene2
Chr1	7500	8460	Gene3
Chr2	8933	9500	Gene4

cut -f2,1 mydata.txt

Chr1	1000
Chr1	3010
Chr1	7500
Chr2	8933

Same order as the input file

```
awk 'BEGIN{FS="\t"; OFS="\t"} {print $2,$1;}' \
mydata.txt
```

1000	Chr1
3010	Chr1
7500	Chr1
8933	Chr2

Customized order

A Good Practice:
Create a shell script file for the one liner

```
cat D7RACXX.fastq | \
head -n 40000 | \
grep AGATCGGAAGAGC | \
wc -l
```

Run the shell script

```
sh checkadapter.sh
```

Debug a one-liner

```
zcat human.gff3.gz | \
awk 'BEGIN {OFS = "\t"}; {if ($3=="gene") print $1,$4-1,$5}' | \
bedtools coverage -a win1mb.bed -b stdin -counts | \
LC_ALL=C sort -k1,1V -k2,2n  > gene.cover.bed
```

```
zcat human.gff3.gz | head -n 1000 > tmpfile
```

```
cat tmpfile| \
awk 'BEGIN {OFS = "\t"}; {if ($3=="gene") print $1,$4-1,$5}' | head -n 100
```

Many bioinformatics software support STDIN instead of input file

Run “BWA” without pipe:

```
bwa mem ref.fa reads.fq > tmp.sam
samtools view -b tmp.sam > out.bam
```

Create a temporary SAM file

With pipe:

```
bwa mem ref.fa reads.fq | samtools view -bS - > out.bam
```

Use "-" to specify input from STDIN instead of a file

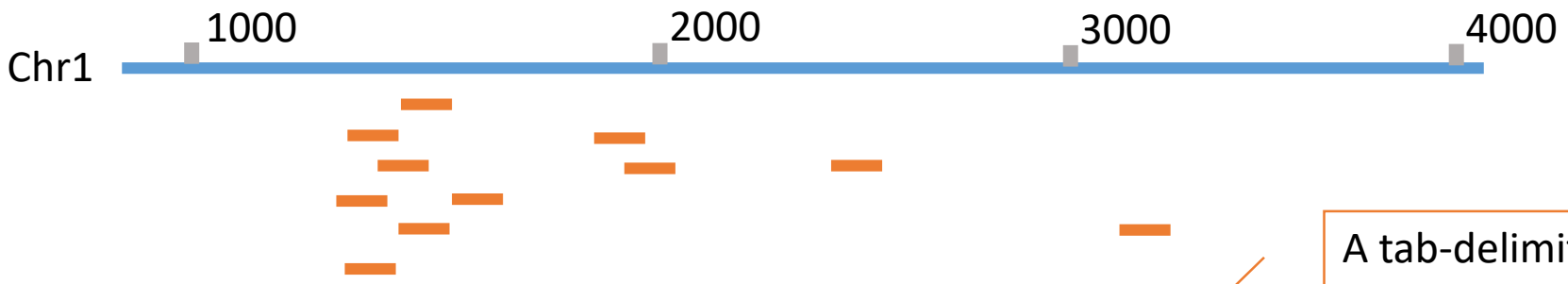
Using pipe with bed tools:

The bedtools takes in two input files, you need to specify which file from stdin

```
…… | bedtools coverage -a FirstFile -b stdin
```

Using BEDtools to process genomics data files

An example: Count the number of reads in each 1kb sliding window of the genome



A tab-delimited text file:

Chr1	21234023
Chr2	19282343
Chr3	15845499

```
bedtools makewindows -g genome.txt -w 1000 -s 1000  >  win1000.bed

bedtools coverage -abam Sample.bam -b win1000.bed -counts> coverage.bed
```

When working with text files from Windows (“/r/n”):

```
dos2unix myfile.sh
```

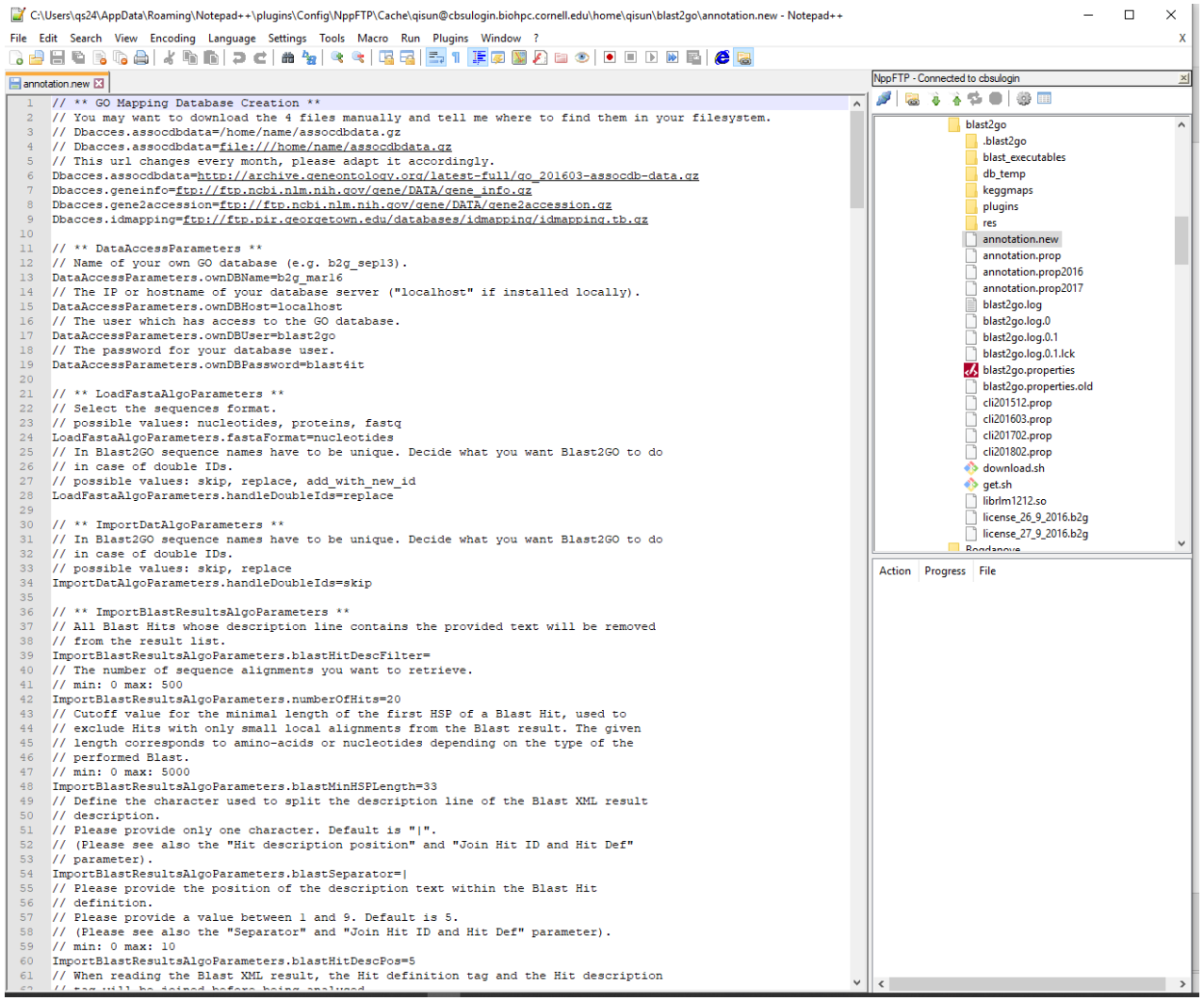
When working with text files from Mac:

- Mac files in general are same as Linux files;
- Only exception is text file saved from Excel (“/r”):

```
mac2unix myfile.txt
```

Edit text file without using “vi”

Software to directly open/edit/save files on a remote Linux server



Windows: Notepad++
(https://notepad-plus-plus.org/)

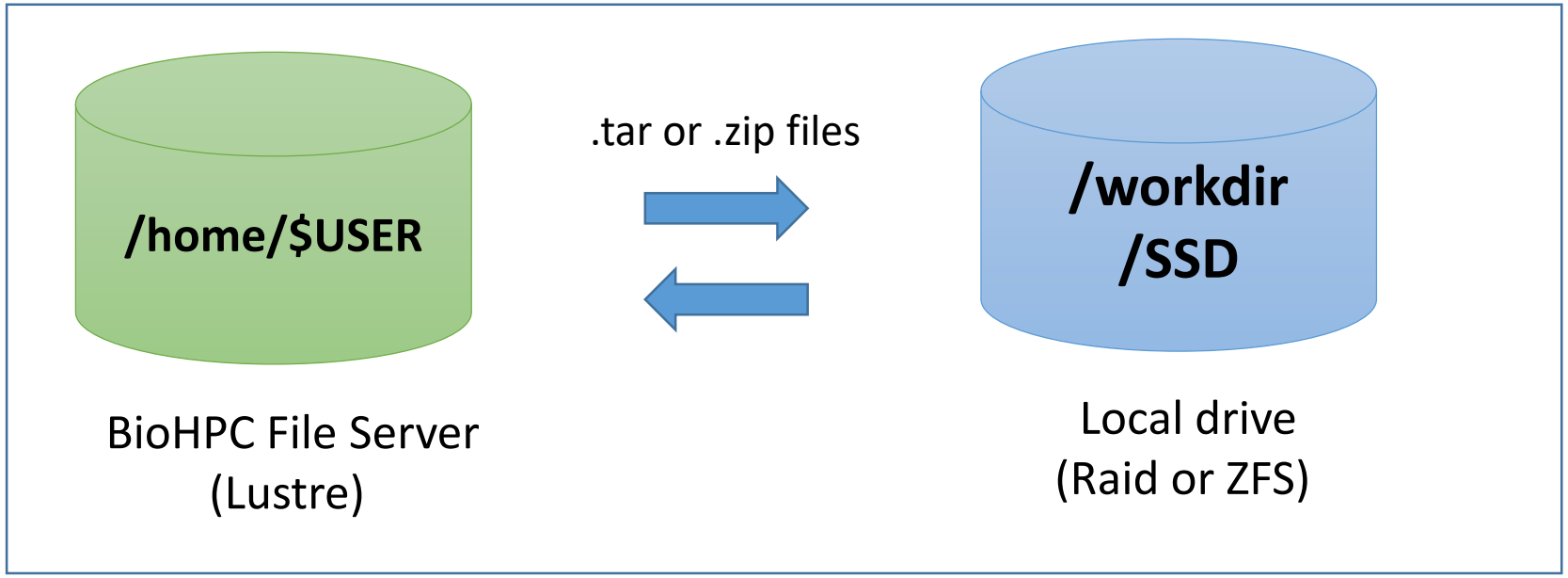
- Install Notepad++
- Click Plugins->Plugin Admins and install NPPFTP
- In the NPPFTP, click “Setting”-> “Profile Setting”, set new profile with connection type “SFTP”.

Mac: BBEdit (free mode)
(https://www.barebones.com/products/bbedit/)

- Open remote file from SFTP Server from the File menu

Dealing with large number of files

Data storage/transfer



Avoid large number of file:

- File system is good for large size, bad for large file numbers;
- Very slow data transfer ;

Solution: Archive into one single file

tar

- Commonly used in Linux;
- Slow to check content or access single file;
- Compression rely on other technology;

```
# archive
tar cf myDataDir.tar myDataDir

#unarchive
tar xf myDataDir.tar
```

- Use “-z” to compress. E.g.
tar cfz myDataDir.tar.gz myDataDir

zip

- Not commonly used in Linux;
- Very fast to check content or access single file;
- Built-in compression;

```
# archive
zip -0 -r myDataDir.zip myDataDir

#unarchive
unzip myDataDir.zip
```

- “-0” suppresses compression, good to use if the files were already compressed.

Accessing files in archive

tar: very slow

```
# list content
tar --list --file=myDataDir.tar

#extract a file
tar --extract --file=myDataDir.tar
myFile
```

* Use --to-stdout to stream to stdout

zip: very fast

```
# list content
less myDataDir.zip
zipinfo myDataDir.zip dataset1/*

# extract a file
#to a new file
unzip -j myDataDir.zip <path/>myFile
#to stdout
unzip -p myDataDir.zip <path/>myFile
```

A few suggestions:

- Organize files into subdirectories. Too many files (>~1000) in a single directory would be difficult to “ls”.
- Too big tar file (>100G) could be difficult to transfer through network.
- Compression saves storage, but slow to process. Do not compress compressed files, or over-compression (set compression level).

Run software on large number of files

Create a task list

```
ls  *fastq.gz | while read file;  \
do echo “STAR --runThreadN 4 --genomeDir REFERENCE   --readFilesIn ${file}; \
done;
```

A text file will be created

```
STAR --runThreadN 4 --genomeDir REFERENCE   --readFilesIn sample1.fastq.gz
STAR --runThreadN 4 --genomeDir REFERENCE   --readFilesIn sample2.fastq.gz
STAR --runThreadN 4 --genomeDir REFERENCE   --readFilesIn sample3.fastq.gz
```

Process the tasks in parallel

```
export PATH=/programs/parallel/bin:$PATH
parallel --jobs 24 --timeout 1h < tasks
```