Bioinformatics (MTAT.03.239) - Assignment 2

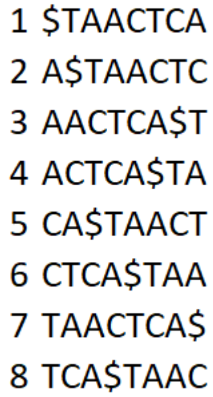
Student: ChengHan Chung

Task 1: Burrows-Wheeler Transform (BWT) (2 point)

Given the Borrows-Wheeler transformed string ‘ACTCA$TA’.

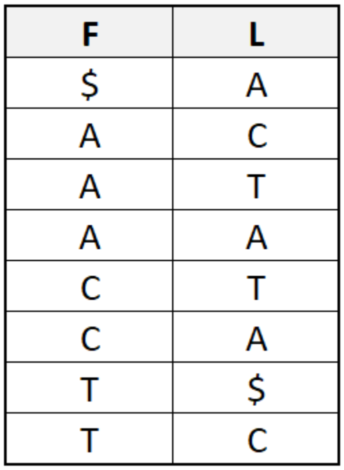
1. Construct the FM index

(A) : An FM-index is created by first taking the Burrows–Wheeler transform (BWT) of the input text. Hence, we have to build up an BWT shown as following below:



1 $TAACTCA
2 A$TAACTC
3 AACTCA$T
4 ACTCA$TA
5 CA$TAACT
6 CTCA$TAA
7 TAACTCA$
8 TCA$TAAC

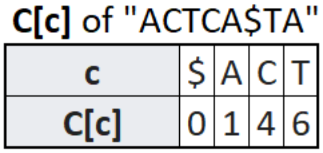
Then, we can create an FM-index, shown as following below:



F	L
$	A
A	C
A	T
A	A
C	T
C	A
T	$
T	C

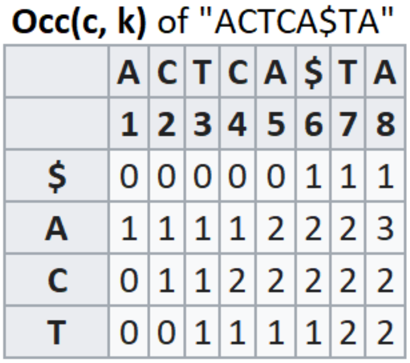
2. Show how many times the pattern ‘CA’ occurs in original string using the FM index and LF(i) (Last-to-First) function

(A) : First of all,we need to create table C[c] and a function Occ(c, k). C[c] is the table which contains the number of occurrences of lexically smaller characters in the text. The function Occ(c, k) is the number of occurrences of character c in the prefix L[1..k].



C[c] of "ACTCA$TA"

c	$	A	C	T
C[c]	0	1	4	6



Occ(c, k) of "ACTCA$TA"

	A	C	T	C	A	$	T	A
	1	2	3	4	5	6	7	8
$	0	0	0	0	0	1	1	1
A	1	1	1	1	2	2	2	3
C	0	1	1	2	2	2	2	2
T	0	0	1	1	1	1	2	2

Then, we can implement LF(i) as following below :

LF(i) = C[L[i]] + Occ(L[i], i)

Then we can get the position `[5..6]`, where the location `CA` occured once `(6-5)`.

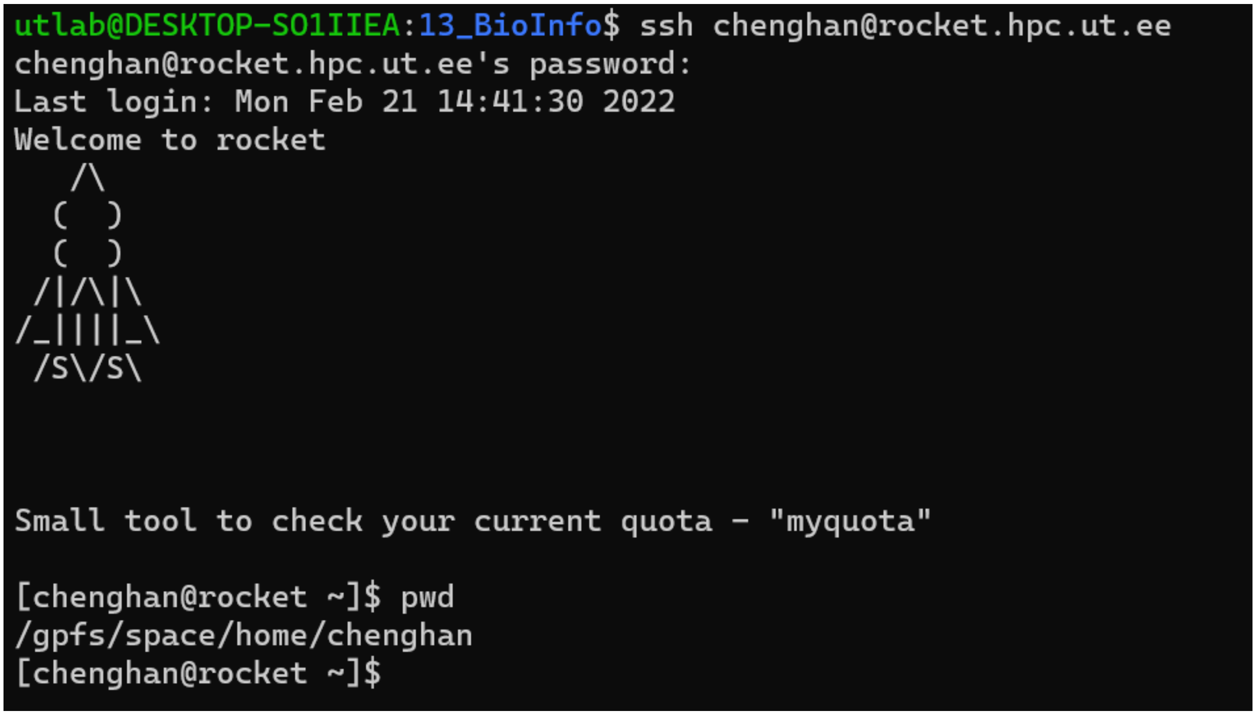
3. Does the pattern 'CATTA' appear in original string?

(A) : According to the algorithm in 1.2, there is no pattern `CATTA` in string.

Task 2: Using the High Performance Computing Center (1 point)

1. If you have never used the HPC before, go through the introductory slides to learn what it's all about.
2. Log into the head node of the rocket cluster using ssh. If you are using Mac or Linux, you can do it straight from the command line: ssh @rocket.hpc.ut.ee. On Windows you might need to install Putty. More instructions are available here.

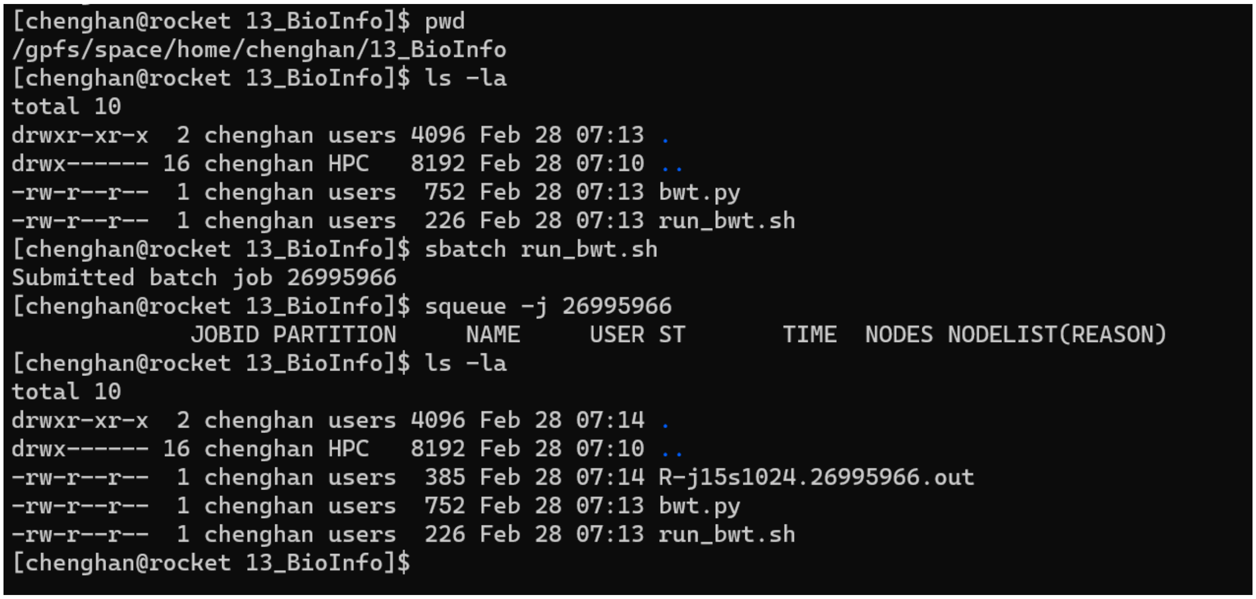
(A) :



```
utlab@DESKTOP-SO1IIEA:13_BioInfo$ ssh chenghan@rocket.hpc.ut.ee
chenghan@rocket.hpc.ut.ee's password:
Last login: Mon Feb 21 14:41:30 2022
Welcome to rocket
   /\
  (  )
  (  )
 /|/\|\
/_||||_\
 /S\/S\



Small tool to check your current quota - "myquota"

[chenghan@rocket ~]$ pwd
/gpfs/space/home/chenghan
[chenghan@rocket ~]$
```

3. Submit your first jobs to the cluster by following the SLURM submit job tutorial and look at it's output.

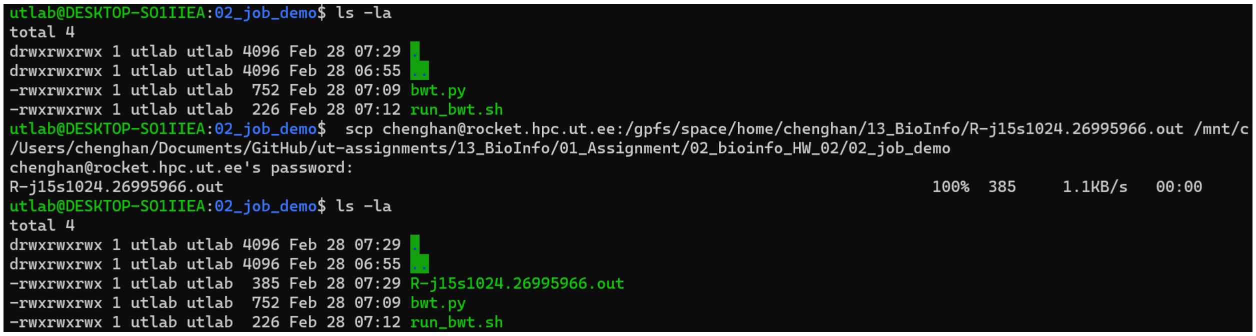
(A) :



```
[chenghan@rocket 13_BioInfo]$ pwd
/gpfs/space/home/chenghan/13_BioInfo
[chenghan@rocket 13_BioInfo]$ ls -la
total 10
drwxr-xr-x  2 chenghan users 4096 Feb 28 07:13 .
drwx------ 16 chenghan HPC   8192 Feb 28 07:10 ..
-rw-r--r--  1 chenghan users  752 Feb 28 07:13 bwt.py
-rw-r--r--  1 chenghan users  226 Feb 28 07:13 run_bwt.sh
[chenghan@rocket 13_BioInfo]$ sbatch run_bwt.sh
Submitted batch job 26995966
[chenghan@rocket 13_BioInfo]$ squeue -j 26995966
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
[chenghan@rocket 13_BioInfo]$ ls -la
total 10
drwxr-xr-x  2 chenghan users 4096 Feb 28 07:14 .
drwx------ 16 chenghan HPC   8192 Feb 28 07:10 ..
-rw-r--r--  1 chenghan users  385 Feb 28 07:14 R-j15s1024.26995966.out
-rw-r--r--  1 chenghan users  752 Feb 28 07:13 bwt.py
-rw-r--r--  1 chenghan users  226 Feb 28 07:13 run_bwt.sh
[chenghan@rocket 13_BioInfo]$
```

4. Learn how to transfer files between your computer and the HPC system. On Mac I prefer to use the Cyberduck sftp client and it might work on Windows as well. Another option is FileZilla, which should also work on all three platforms. If you prefer command line over Graphical User Interfaces please check out the scp usage.

(A) :



```
utlab@DESKTOP-SO1IIEA:02_job_demo$ ls -la
total 4
drwxrwxrwx 1 utlab utlab 4096 Feb 28 07:29 .
drwxrwxrwx 1 utlab utlab 4096 Feb 28 06:55 ..
-rwxrwxrwx 1 utlab utlab  752 Feb 28 07:09 bwt.py
-rwxrwxrwx 1 utlab utlab  226 Feb 28 07:12 run_bwt.sh
utlab@DESKTOP-SO1IIEA:02_job_demo$  scp chenghan@rocket.hpc.ut.ee:/gpfs/space/home/chenghan/13_BioInfo/R-j15s1024.26995966.out /mnt/c
/Users/chenghan/Documents/GitHub/ut-assignments/13_BioInfo/01_Assignment/02_bioinfo_HW_02/02_job_demo
chenghan@rocket.hpc.ut.ee's password:
R-j15s1024.26995966.out                                                                       100%  385     1.1KB/s   00:00
utlab@DESKTOP-SO1IIEA:02_job_demo$ ls -la
total 4
drwxrwxrwx 1 utlab utlab 4096 Feb 28 07:29 .
drwxrwxrwx 1 utlab utlab 4096 Feb 28 06:55 ..
-rwxrwxrwx 1 utlab utlab  385 Feb 28 07:29 R-j15s1024.26995966.out
-rwxrwxrwx 1 utlab utlab  752 Feb 28 07:09 bwt.py
-rwxrwxrwx 1 utlab utlab  226 Feb 28 07:12 run_bwt.sh
```

5. Demonstrate the you have managed to successfully execute your first job by copying the contents of the SLURM output file into your report.

(A) : About job and result, please refer to folder `02_job_demo`.

Task 3: RNA-seq alignment (2 points)

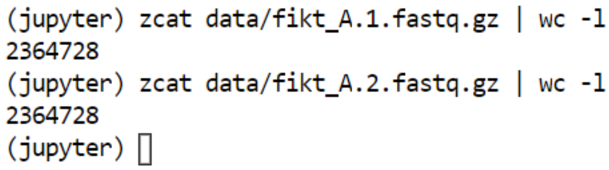
Using the RNA-seq alignment tutorial, answer the following questions:

1. How many reads are there in the `fikt_A.1.fastq.gz` and `fikt_A.2.fastq.gz` FASTQ files?

(A) : By using combination of `zcat` and `wc`, we can count the number of line in those two files, the reult shown as following below:

`**fikt_A.1.fastq.gz :** 2364728`

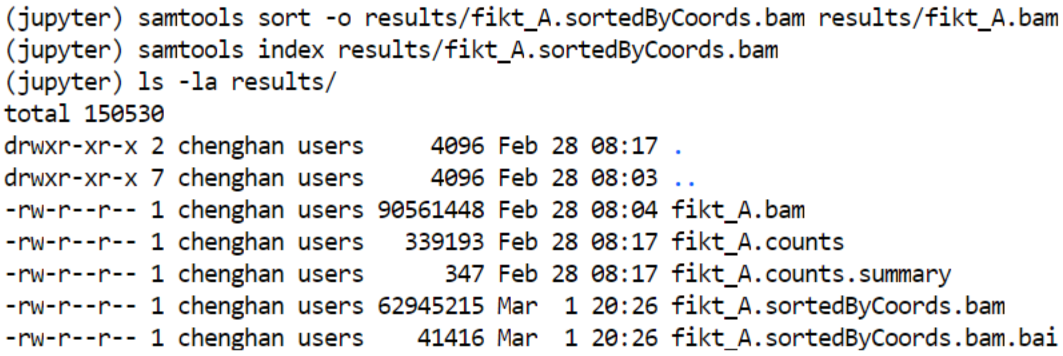
`**fikt_A.2.fastq.gz :** 2364728`



```
(jupyter) zcat data/fikt_A.1.fastq.gz | wc -l
2364728
(jupyter) zcat data/fikt_A.2.fastq.gz | wc -l
2364728
(jupyter) 
```

2. Following the instructions, align the FASTQ files to the reference genome. Sort the alignments by position and create the index.

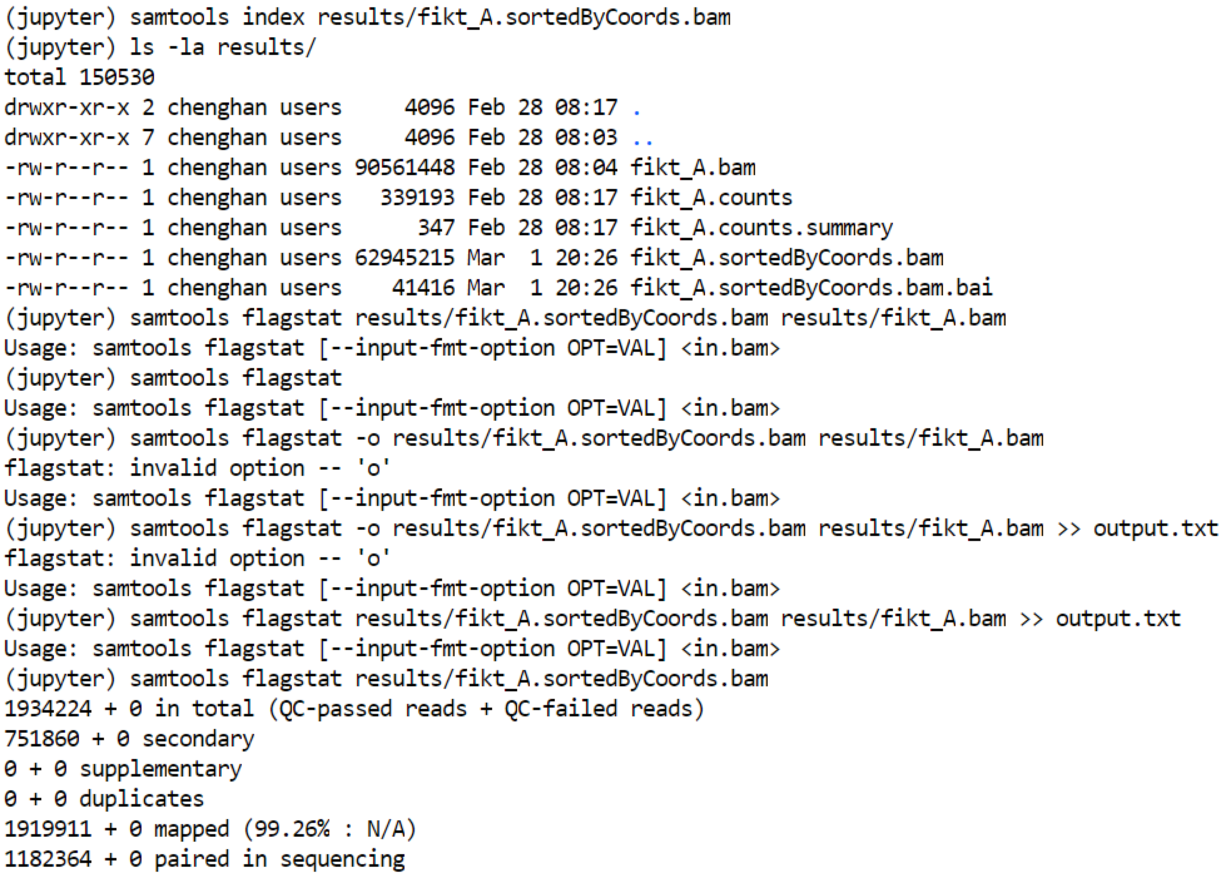
(A) :



```
(jupyter) samtools sort -o results/fikt_A.sortedByCoords.bam results/fikt_A.bam
(jupyter) samtools index results/fikt_A.sortedByCoords.bam
(jupyter) ls -la results/
total 150530
drwxr-xr-x 2 chenghan users     4096 Feb 28 08:17 .
drwxr-xr-x 7 chenghan users     4096 Feb 28 08:03 ..
-rw-r--r-- 1 chenghan users 90561448 Feb 28 08:04 fikt_A.bam
-rw-r--r-- 1 chenghan users   339193 Feb 28 08:17 fikt_A.counts
-rw-r--r-- 1 chenghan users      347 Feb 28 08:17 fikt_A.counts.summary
-rw-r--r-- 1 chenghan users 62945215 Mar  1 20:26 fikt_A.sortedByCoords.bam
-rw-r--r-- 1 chenghan users    41416 Mar  1 20:26 fikt_A.sortedByCoords.bam.bai
```

3. What fraction of the reads mapped to the reference genome? (HINT: use samtools flagstat).

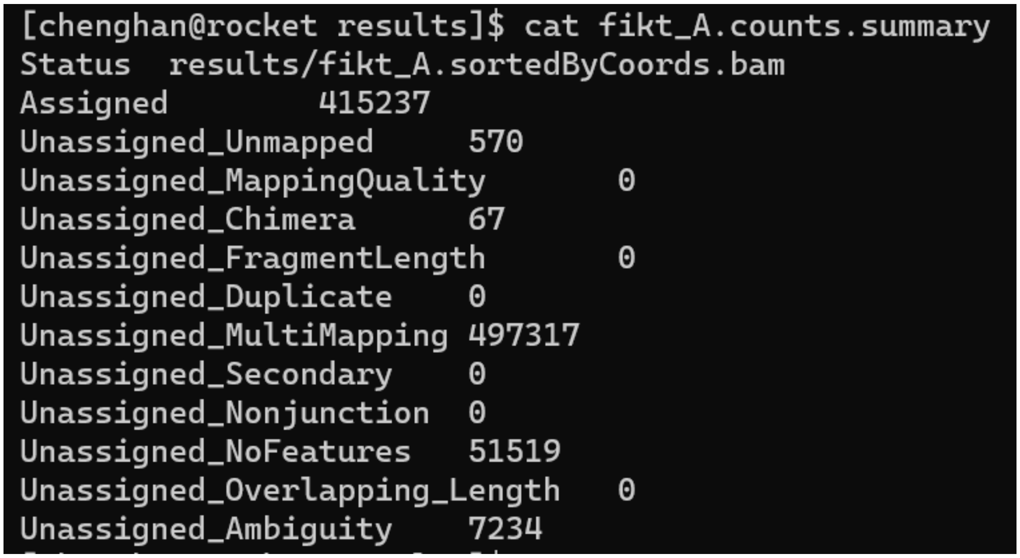
(A) : 150530



```
(jupyter) samtools index results/fikt_A.sortedByCoords.bam
(jupyter) ls -la results/
total 150530
drwxr-xr-x 2 chenghan users     4096 Feb 28 08:17 .
drwxr-xr-x 7 chenghan users     4096 Feb 28 08:03 ..
-rw-r--r-- 1 chenghan users 90561448 Feb 28 08:04 fikt_A.bam
-rw-r--r-- 1 chenghan users   339193 Feb 28 08:17 fikt_A.counts
-rw-r--r-- 1 chenghan users      347 Feb 28 08:17 fikt_A.counts.summary
-rw-r--r-- 1 chenghan users 62945215 Mar  1 20:26 fikt_A.sortedByCoords.bam
-rw-r--r-- 1 chenghan users    41416 Mar  1 20:26 fikt_A.sortedByCoords.bam.bai
(jupyter) samtools flagstat results/fikt_A.sortedByCoords.bam results/fikt_A.bam
Usage: samtools flagstat [--input-fmt-option OPT=VAL] <in.bam>
(jupyter) samtools flagstat
Usage: samtools flagstat [--input-fmt-option OPT=VAL] <in.bam>
(jupyter) samtools flagstat -o results/fikt_A.sortedByCoords.bam results/fikt_A.bam
flagstat: invalid option -- 'o'
Usage: samtools flagstat [--input-fmt-option OPT=VAL] <in.bam>
(jupyter) samtools flagstat -o results/fikt_A.sortedByCoords.bam results/fikt_A.bam >> output.txt
flagstat: invalid option -- 'o'
Usage: samtools flagstat [--input-fmt-option OPT=VAL] <in.bam>
(jupyter) samtools flagstat results/fikt_A.sortedByCoords.bam results/fikt_A.bam >> output.txt
Usage: samtools flagstat [--input-fmt-option OPT=VAL] <in.bam>
(jupyter) samtools flagstat results/fikt_A.sortedByCoords.bam
1934224 + 0 in total (QC-passed reads + QC-failed reads)
751860 + 0 secondary
0 + 0 supplementary
0 + 0 duplicates
1919911 + 0 mapped (99.26% : N/A)
1182364 + 0 paired in sequencing
```

4. What fraction of the paired-end fragments were assigned to genes? (HINT: You can find this from the summary file created by featureCounts)

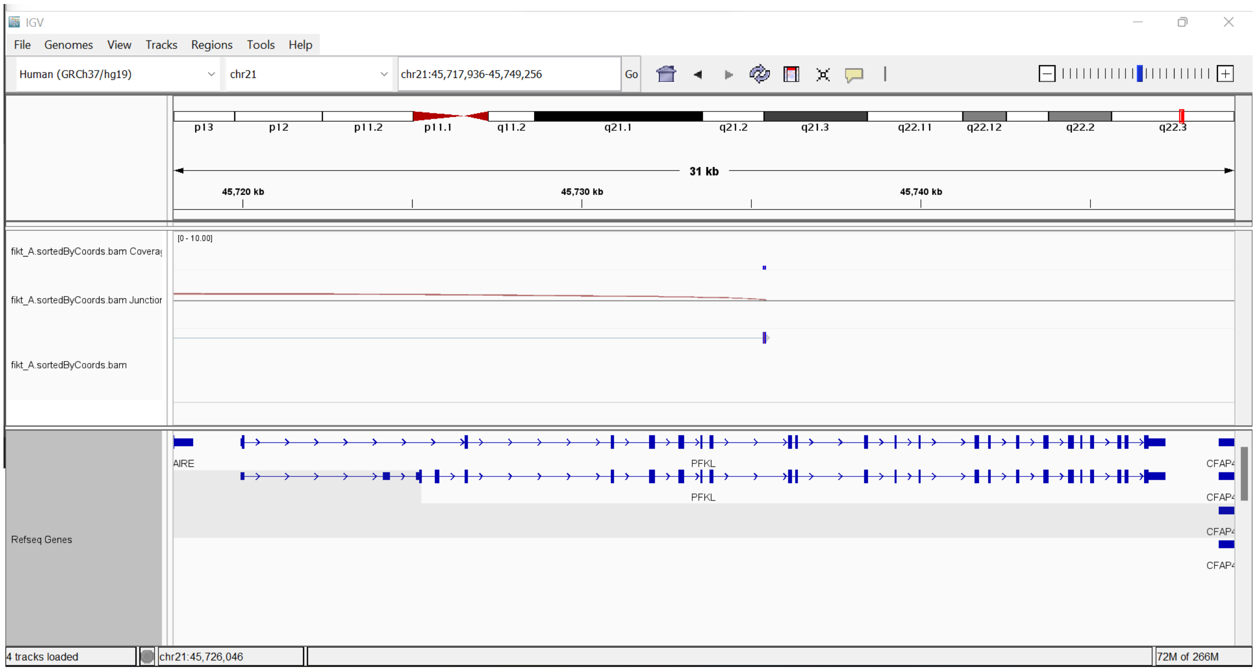
(A) :



```
[chenghan@rocket results]$ cat fikt_A.counts.summary
Status   results/fikt_A.sortedByCoords.bam
Assigned        415237
Unassigned_Unmapped     570
Unassigned_MappingQuality       0
Unassigned_Chimera      67
Unassigned_FragmentLength       0
Unassigned_Duplicate    0
Unassigned_MultiMapping 497317
Unassigned_Secondary    0
Unassigned_Nonjunction  0
Unassigned_NoFeatures   51519
Unassigned_Overlapping_Length   0
Unassigned_Ambiguity    7234
```

5. Copy to sorted BAM file together with the index from the HPC to your own environment (See Task 2). Open the BAM file in IGV. Zoom into the PFKL gene on chromosome 21. You should be able to see individual reads mapping to the exons of the gene. Now move to the neighbouring AIRE gene. What do you see? Make IGV screenshots for both genes and include them into your report.

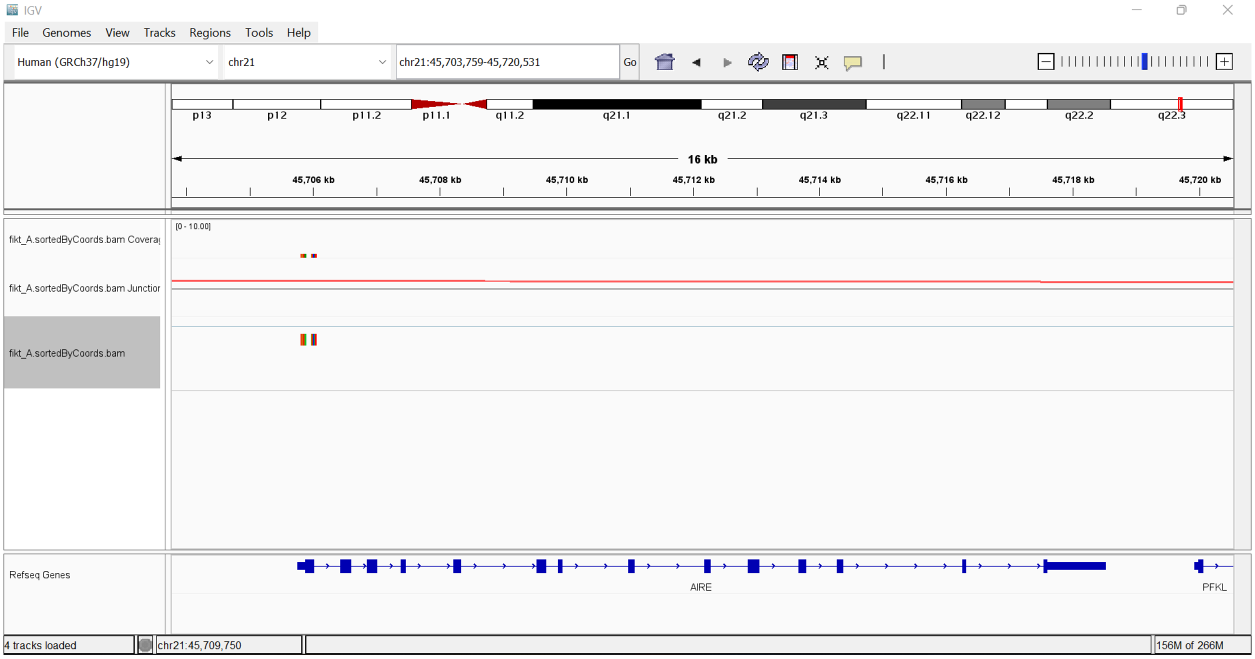
PFLK



Human (GRCh37/hg19) chr21 chr21:45,717,936-45,749,256

4 tracks loaded chr21:45,726,046 72M of 266M

AIRE



Human (GRCh37/hg19)

chr21

chr21:45,703,759-45,720,531

chr21:45,709,750

6. Find the number of paired-end fragments overlapping the PFKL and AIRE genes from the featureCounts file (last column). Do these broadly match what you observed in the IGV? (No need to count fragments manually from IGV) (HINT: You can use the search box on the Ensembl website to find the gene ids for both gens).

7. Repeat the same processing steps on all of the four samples (fikt_A, fikt_C, eipl_A, eipl_C) found in Zenodo. Report the paired-end fragment counts for PFKL and AIRE genes in all four samples.