

Lesson 5

Sequence mapping part II

Ultima Genomics reposted this



Ultima Genomics

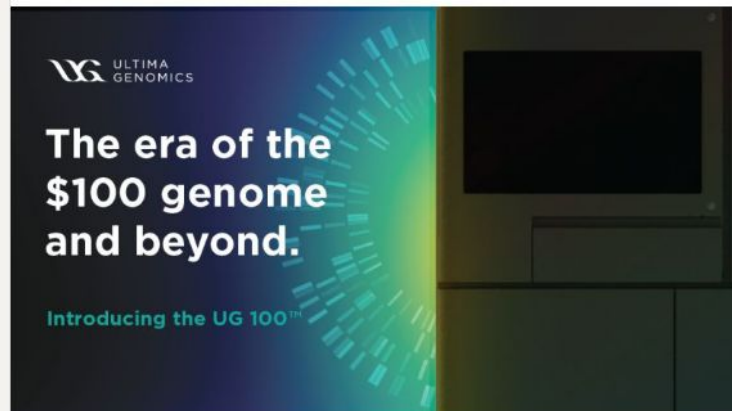
14,006 followers

2d • Edited •

Ultima is gearing up for the full commercial availability of the UG 100™ at the upcoming [AGBT - Advances in Genome Biology and Technology](#) (AGBT) on February 5, 2024. We will also be unveiling our ppmSeq™ Technology with industry-leading raw read accuracy for calling SNVs for rare event detection applications.

Hear why our early access customers are excited about the launch of the UG 100™ and the era of the \$100 genome and beyond.

https://lnkd.in/gANE_9Cv



Ultima Genomics Announces End of Early Access and Upcoming Launch of UG 100™

businesswire.com • 5 min read

David Bogumil and 224 others

9 comments • 34 reposts



Like



Comment



Repost



Send

FASTA

Ready, Steady!

FASTQ

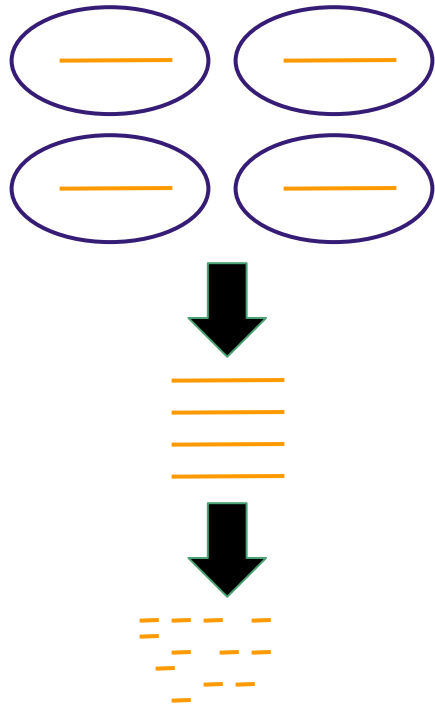
Read 1

Sequence

Q-score

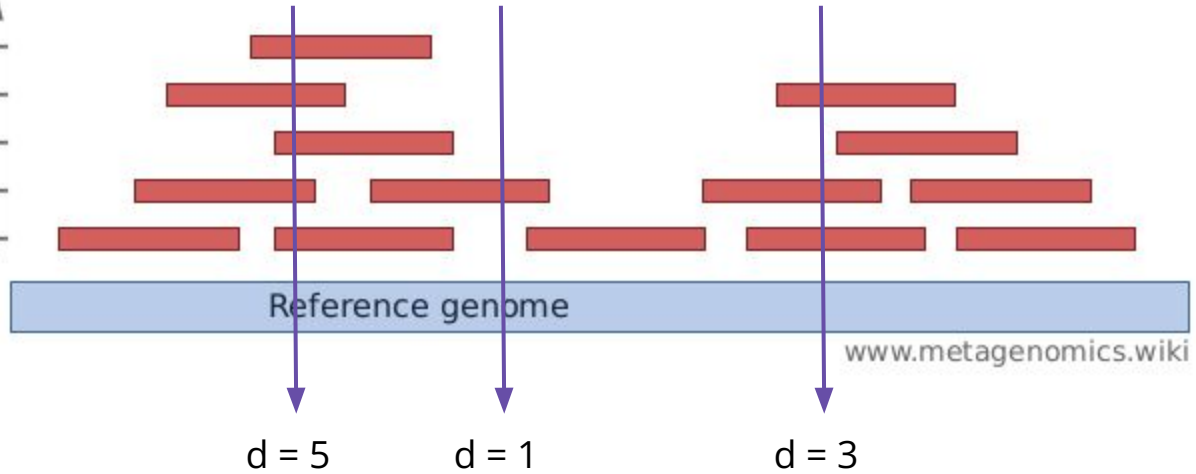
Read 2

Sequencing depth



Depth of coverage

5X
4X
3X
2X
1X



$$D = \frac{L * N}{G}$$

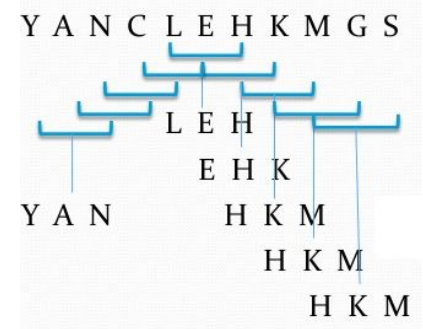
Sequence mapping - BLAST

CGTGTACAG

ACCTGAGGATCGTATACAAGTTA

C	G	T	G	T	A	C	A	-	G
C	G	T	A	T	A	C	A	A	G

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

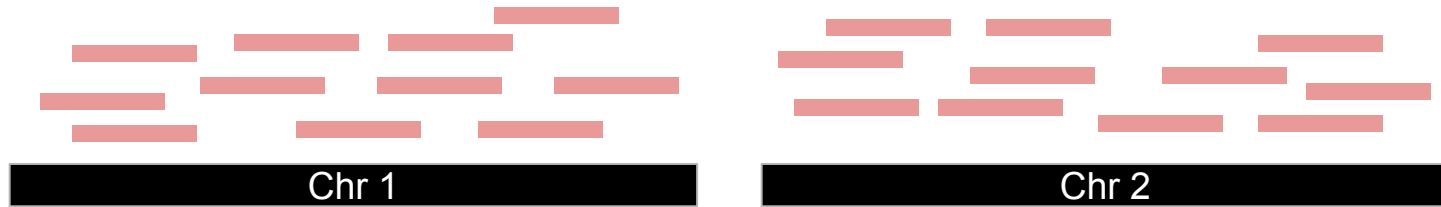


By the end of this lesson you will...

- Know how to use BWA for short read mapping
- Understand the Sequence Alignment Map (SAM) and BAM file formats
- Be able to view and manipulate SAM/BAM files using samtools
- Be familiar with the IGV genome browser and how to use it for viewing BAM files

Short read mapping

- Map (search and align) Illumina reads to a reference genome
- Find the most likely position of a read in the genome
- Probably the most common task in genomics



Short read mapping to reference genome - why?

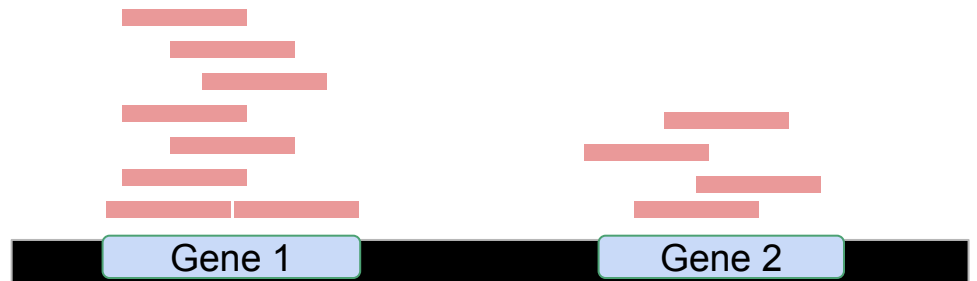
- Variant calling / genotyping

DNA



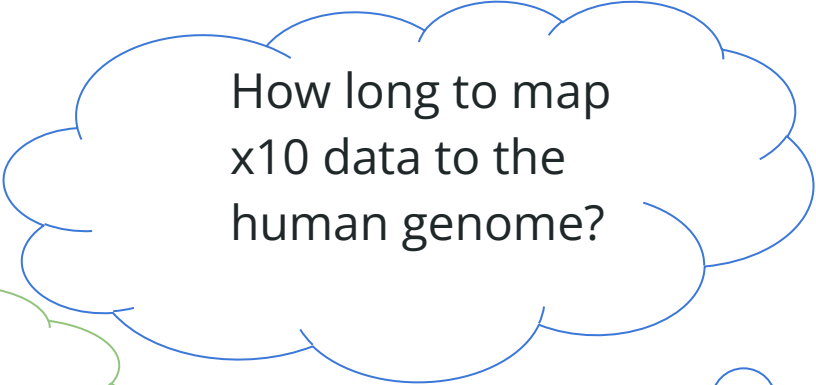
- Gene expression profiling

RNA

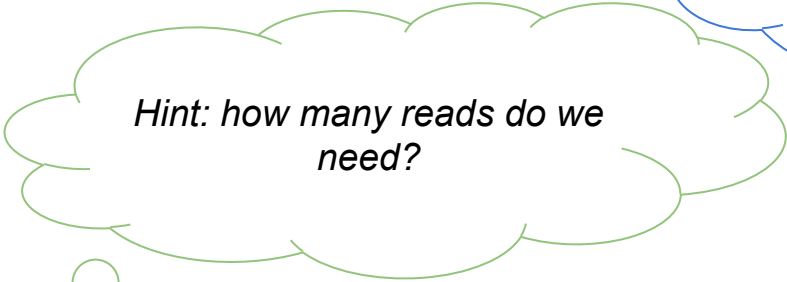


The challenge - scale and speed

- We need to map millions to hundreds of millions of reads
- **Can we use Blast?**
- Blastn - ~100 reads / sec
- Human genome - ~ 3Gb
- Assume 100bp reads



How long to map
x10 data to the
human genome?



*Hint: how many reads do we
need?*

Can we use Blast?

- Blastn - ~100 reads / sec
- Human genome - ~ 3Gb
- Assume 100bp reads

Data required:

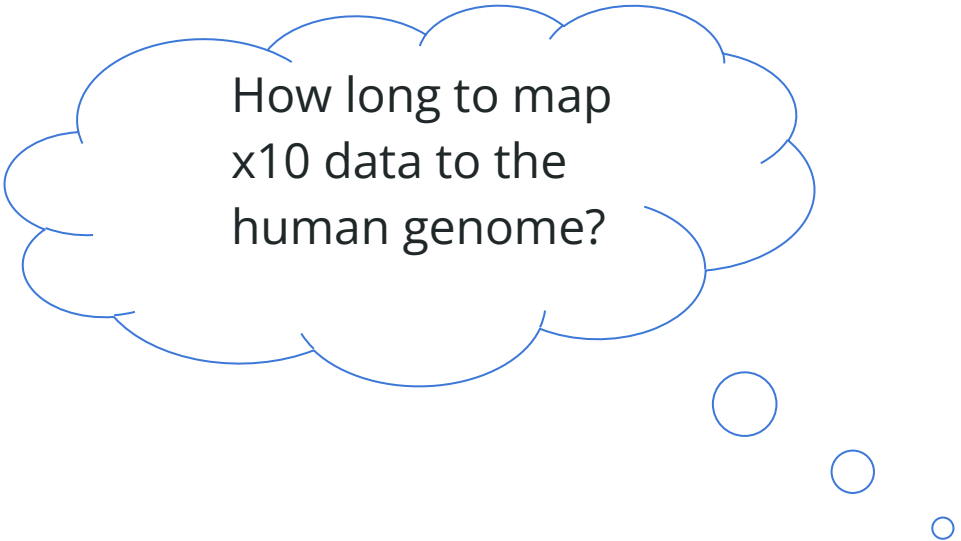
3 Gb x 10 = 30 Gb

Reads required:

30 Gb / 100 = 300 M reads

Time to map:

300 M reads / (100 reads/sec) = 3M sec
= ~ **35 days**



How long to map
x10 data to the
human genome?

BWA - Burrows-Wheeler Aligner

- Specifically designed for mapping of short reads
- Maps ~2,200 reads / sec (one CPU)
- Allows parallel computing
- Contains three algorithms - the most useful is **BWA-MEM**

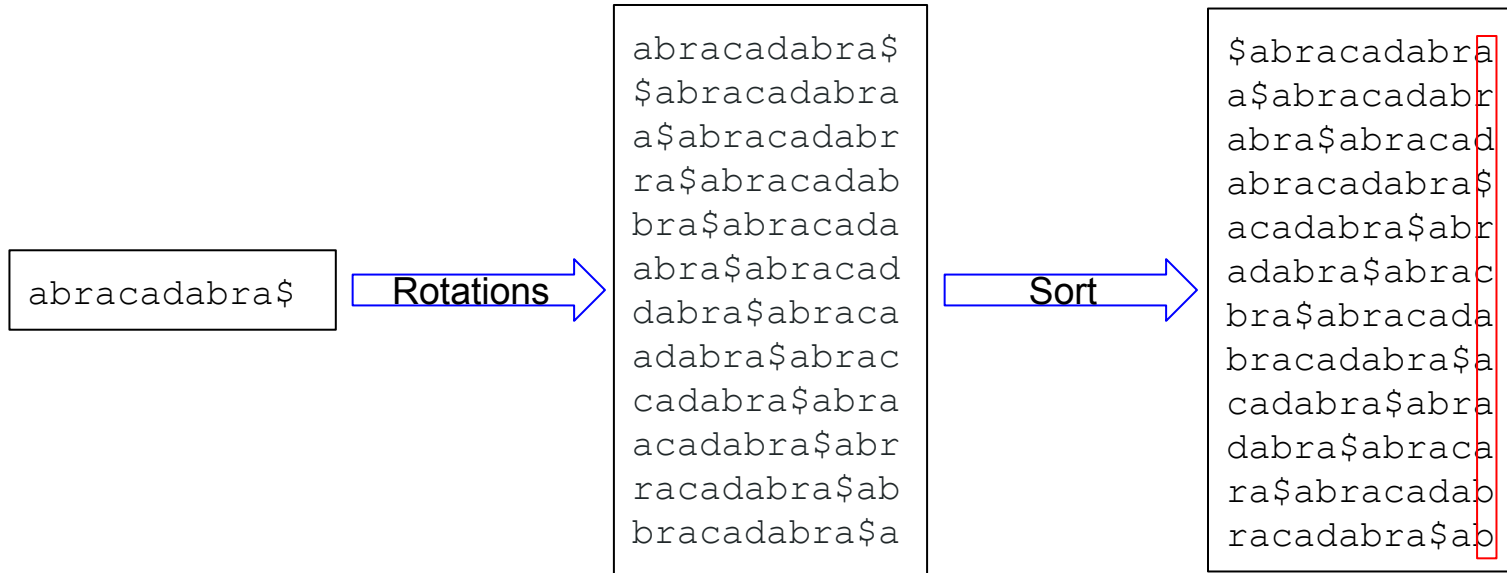
BWA - limitations

- Only works for nucleotides (usually DNA, not RNA)
- Less effective when:
 - Queries are very long
 - Reads are highly diverged from the reference
 - Reads contain lots of sequencing errors
- Usually offers a good accuracy-speed balance

BWA algorithm overview

- Step 1: Index the reference genome
- Step 2: Search for reads
- Indexing is based on the **Burrows-Wheeler's transformation**
- Index allows easy searching:
 - Quick
 - Memory efficient

The Burrows-Wheeler's transformation



BWT(abracadabra\$) = ard\$rcaaaaabb

The Burrows-Wheeler's transformation

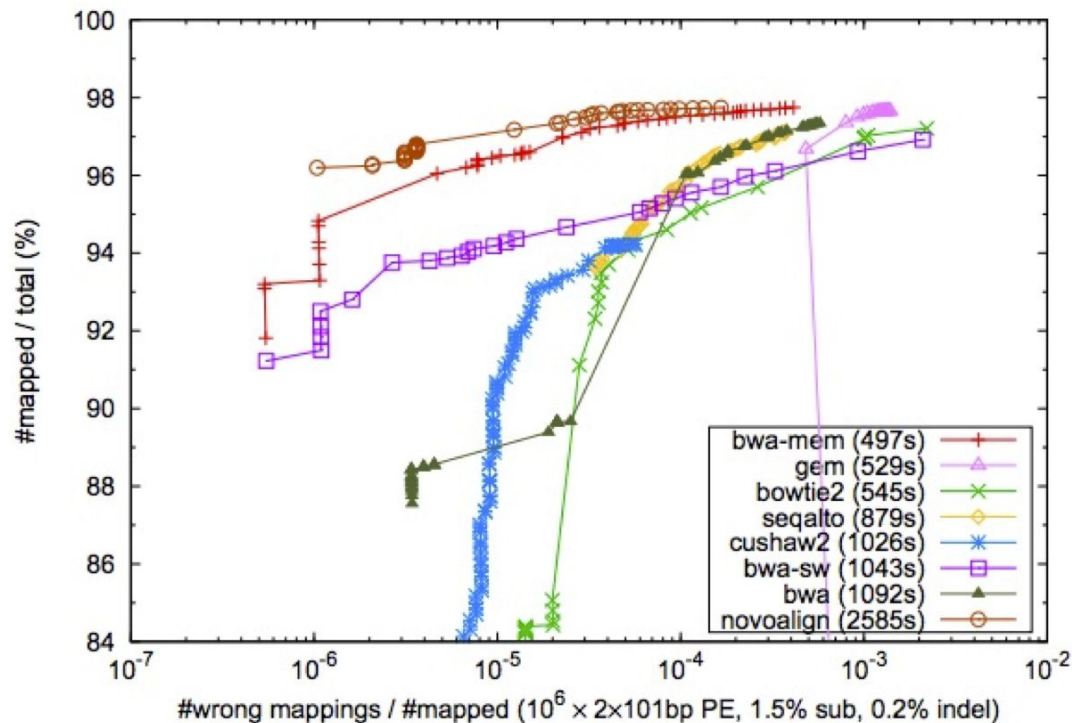
- BWT is **reversible** - we can get back from $\text{BWT}(G)$ to G
- $\text{BWT}(G)$ tends to cluster the same characters together - easy to compress
- Using some additional data structures, $\text{BWT}(G)$ can be searched efficiently

$\text{BWT}(\text{abracadabra}\$) = \text{ard\$rcaaaabb}$

Aligners Comparison

<u>Aligner</u>	<u>Index</u>	<u>Applications</u>	<u>Availability</u>
BWA-mem	Burrows-Wheeler	DNA, SE, PE	open-source
Bowtie2	Burrows-Wheeler	DNA, SE, PE	open-source
Novoalign	Hash-Based	DNA, SE, PE	propriety
TopHat	Burrows-Wheeler	RNA-seq	open-source
STAR	Hash-Based (reads)	RNA-seq	open-source
GSNAP	Hash-Based (reads)	RNA-seq	open-source

Aligners Comparison



BWA-MEM Workflow

This takes a long time, but you do it once

Create BWT of reference genome.

```
$ bwa index grch38.fa
```



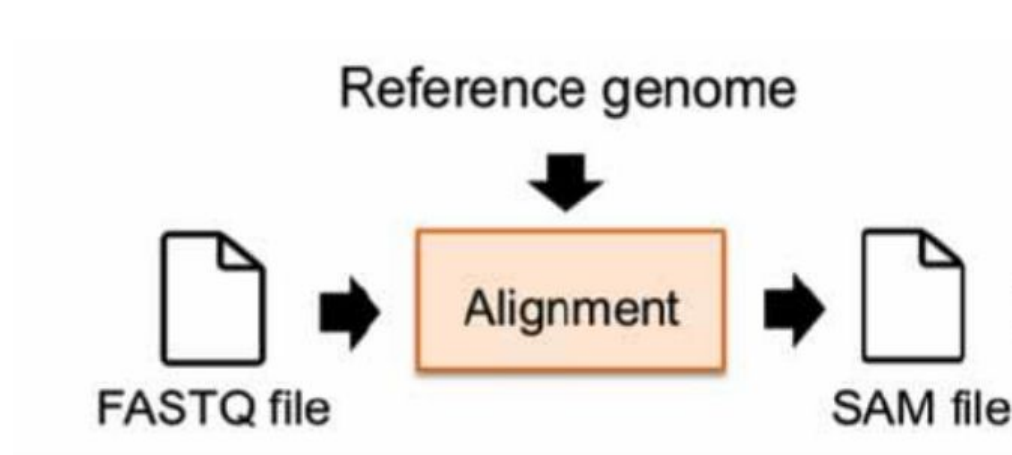
Output is in SAM format.

Use multiple threads if you have a computer with multiple CPUs.

Align paired-end FASTQ to BWT index.

```
$ bwa mem -t 16 grch38.fa 1.fq 2.fq > sample.sam
```

FASTQ to BAM



Sequence Alignment and Mapping

Sequence analysis

The Sequence Alignment/Map format and SAMtools

Heng Li^{1,†}, Bob Handsaker^{2,†}, Alec Wysoker², Tim Fennell², Jue Ruan³, Nils Homer⁴, Gabor Marth⁵, Goncalo Abecasis⁶, Richard Durbin^{1,*} and 1000 Genome Project Data Processing Subgroup⁷

¹Wellcome Trust Sanger Institute, Wellcome Trust Genome Campus, Cambridge, CB10 1SA, UK, ²Broad Institute of MIT and Harvard, Cambridge, MA 02141, USA, ³Beijing Institute of Genomics, Chinese Academy of Science, Beijing 100029, China, ⁴Department of Computer Science, University of California Los Angeles, Los Angeles, CA 90095, ⁵Department of Biology, Boston College, Chestnut Hill, MA 02467, ⁶Center for Statistical Genetics, Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA and ⁷<http://1000genomes.org>

Received on April 28, 2009; revised on May 28, 2009; accepted on May 30, 2009

Advance Access publication June 8, 2009

Associate Editor: Alfonso Valencia

Table 1. Mandatory fields in the SAM format

No.	Name	Description
1	QNAME	Query NAME of the read or the read pair
2	FLAG	Bitwise FLAG (pairing, strand, mate strand, etc.)
3	RNAME	Reference sequence NAME
4	POS	1-Based leftmost POSition of clipped alignment
5	MAPQ	MAPping Quality (Phred-scaled)
6	CIGAR	Extended CIGAR string (operations: MIDNSHP)
7	MRNM	Mate Reference NaMe ('=' if same as RNAME)
8	MPOS	1-Based leftmost Mate POSition
9	ISIZE	Inferred Insert SIZE
10	SEQ	Query SEQUENCE on the same strand as the reference
11	QUAL	Query QUALity (ASCII-33=Phred base quality)

The SAM format sections

- Header
 - Lines start with '@'
 - Meta-data - General information about the file
- Alignments
 - Contains the actual read mapping information
 - Each line has 11 mandatory fields (columns)
 - Additional fields may be included
 - Fields are separated by tabs

What command will fetch only header lines?

```
@SQ      SN:SL4.0ch00    LN:9643250
@SQ      SN:SL4.0ch01    LN:90863682
@SQ      SN:SL4.0ch02    LN:53473368
@SQ      SN:SL4.0ch03    LN:65298490
@SQ      SN:SL4.0ch04    LN:64459972
@SQ      SN:SL4.0ch05    LN:65269487
@SQ      SN:SL4.0ch06    LN:47258699
@SQ      SN:SL4.0ch07    LN:67883646
@SQ      SN:SL4.0ch08    LN:63995357
@SQ      SN:SL4.0ch09    LN:68513564
@SQ      SN:SL4.0ch10    LN:64792705
@SQ      SN:SL4.0ch11    LN:54379777
@SQ      SN:SL4.0ch12    LN:66688036
@PG      ID:bwa   PN:bwa   VN:0.7.12-r1039 CL:bwa mem -t 10 /groups/itay_mayrose/nosnap/liorglic/Projects/GPAD/data/S_lycopersi
cum_chromosomes.4.00.fa /groups/itay_mayrose/nosnap/liorglic/Projects/GPAD/data/SRR1572628_1.fastq /groups/itay_mayrose/nosn
ap/liorglic/Projects/GPAD/data/SRR1572628_2.fastq
SRR1572628.1   83       SL4.0ch05        25938772         60       100M     =       25938399       -473       GTGAAGGCGTTTGACTGG
GCATGGAGTTTAAGAGTGAACAAACACTTTTTGAACTTAAGGTCTAAAATAAGTCAGAAATTGTGTGGCTGTAAATCA CCCCDBDDDDDEEFFFFHGHJFIIGIIEIJJJJJJI
HIHFBDJBjjjiiijigijjjijjjjjjjjjjjjjjjjjjjjjjjjiHHHHHFFFFFCC NM:i:0 MD:Z:100 AS:i:100 XS:i:21
SRR1572628.1   163      SL4.0ch05        25938399         60       100M     =       25938772       473       CACAGATGCTGGGAGATCTA
TTTCATATCCCATGCAGTAAAATCAAACATGGGCAGAAACCAAATCAATCTTGCCTCGACCCAAGTTCTCATCAGTATC CCCFFFHHHHHHJJJJJJJJIIJJJJJJJJGIJ
JJJJJJJJJJHJJJJJJJJJJJJJJJJJJJJJJJJIIHHHHFFFDDDBBCCDEEDEEDD@CDD NM:i:0 MD:Z:100 AS:i:100 XS:i:0
SRR1572628.2   83       SL4.0ch11        27210743         60       100M     =       27210394       -449      TCACAAGCATGACGGACCAT
CACAGAATGCCGTAAACCCACACTTAGTCAGACATCCCTATCTTCAATCAGCAACTGCACTAAGCTGCAACCTACGGACCG 3A<A@<CC?895@>3@CC@;(DPC?:5GGGC=C
=GFAGFDHHD?D3DIHFB?DEFDEG?GGFHf@CA?CBIIHIFGGEC+CHFCADDDD@@@ NM:i:0 MD:Z:100 AS:i:100 XS:i:65
SRR1572628.2   163      SL4.0ch11        27210394         60       100M     =       27210743       449      CTCCAAACTTCATGACGAAC
CTACAGGACGGACTGTCGTAGACTTTACGGACCATCGTGGACTCCGTCACCCCCACACTTAGACTTACCCATCTTCCTTCA @@@FFFFFH HHHGIII6FGGGIIG>BGHFH:FBFIB=
CG>FGIGHGGAHFECDF?BCBECDGB?ABD@BB7<8?<CCDCC@@A>39CCC@CDD<CCC NM:i:0 MD:Z:100 AS:i:100 XS:i:25
SRR1572628.3   83       SL4.0ch01        55877048         60       100M     =       55876674       -474      TCTATCATCTTCGTGTTTAG
GATTATTGGTGAAATAGTACTACCACATAGAACTTTTTGGCAACACGCCTACTGTTCTTTTTTCAACGCCTATTTGATCCT CEDEDDDDDDDDDDDDDCDDDEDDBCDEDEDEFEFC@a@F
CAHE?HEEIJIHJJJJJJJJIGFIIJJIIHEJJJJJJJJJIHF@IHHGH HHHFFFFFFCC NM:i:0 MD:Z:100 AS:i:100 XS:i:0
SRR1572628.3   163      SL4.0ch01        55876674         60       100M     =       55877048       474      TAGTAGACGTAAACAAAAAG
GAAGAAAATGATCCCAAATAGACATAACATCAACCGAGTAGATAATTTTTCTCCATTCAAGCTAGGATGAAGAATGTT BCCFFDFFGHFHJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
JBFHCHIGHGECHIJJIIICEHIJGIHFEGHHFFFFFFFFEECECCCD DDDDDDDDDCCCCAC NM:i:0 MD:Z:100 AS:i:100 XS:i:0
```

SAM Format

Col #	Name	Meaning	Example
1	QNAME	Read or Pair name	HWI:ST156_1:278:1:1058:4544:0
2	FLAG	Bitwise FLAG	<i>soon!</i>
3	RNAME	Reference sequence name	chr1
4	POS	1-based alignment start coordinate	8,724,005
5	MAPQ	Mapping quality	<i>soon!</i>
6	CIGAR	Extended CIGAR string	<i>soon!</i>
7	MRNM	If paired, the mate's reference seq.	chr1
8	MPOS	If paired, the mate's alignment start	8,724,505
9	ISIZE	If paired, the insert size	562
10	SEQ	The sequence of the query/mate	ACAAATTCAG...
11	QUAL	The quality string for the query/mate	HHH\$^^%\$\$\$...
12	OPT	Optional Tags	XA:i:2, MD:Z:0T34G15

SAM Format

[illegible]

MAPQ

MAPQ - mapping quality

Definition: $-10 \log_{10} \Pr\{\text{mapping position is wrong}\}$

The higher - the better

Usually between 0 and 60

Calculation of MAPQ is differ between aligners

It considers alignment score, Phred score and alternative mappings

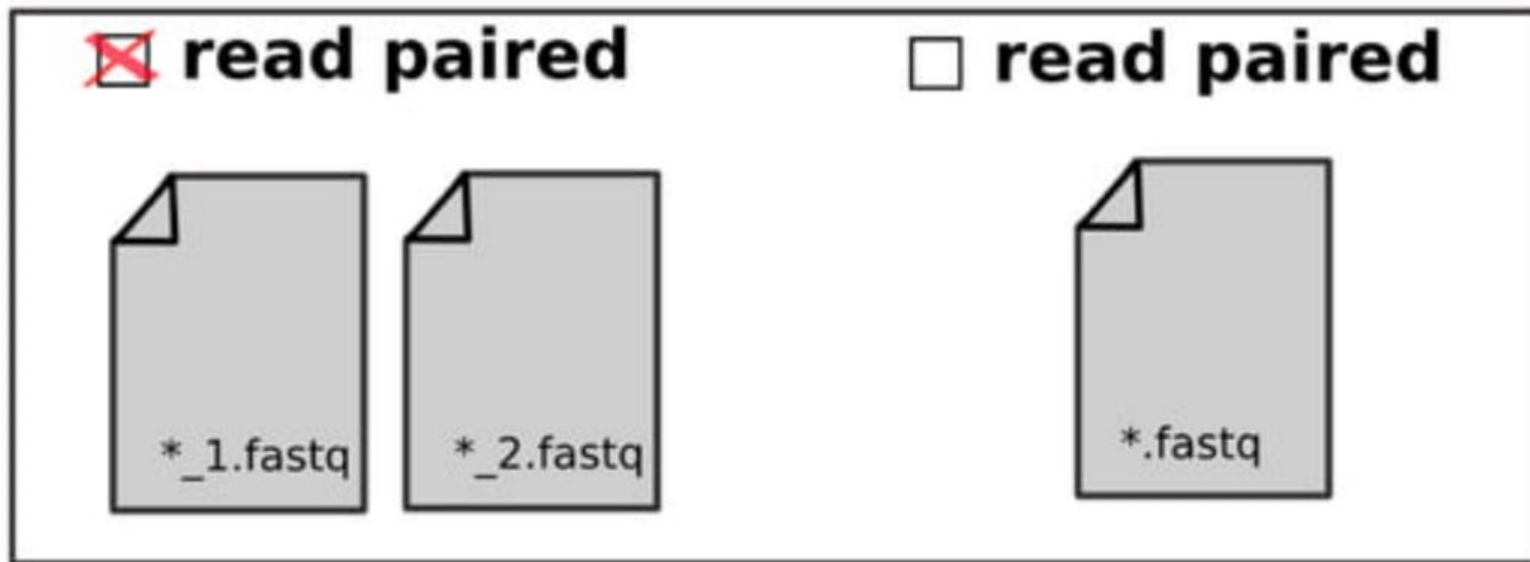
As a rule of thumb:

- MAPQ > 30 is considered a good mapping
- MAPQ 0 usually means ambiguous mapping

SAM Flag

base2	base10	base16	Meaning	Applies to:
000000000001	1	0x0001	The read originated from a paired sequencing molecule	Both
000000000010	2	0x0002	The read is mapped in a proper pair	Pairs only
000000000100	4	0x0004	The query sequence itself is unmapped	Both
000000001000	8	0x0008	The query's mate is unmapped	Pairs only
000000010000	16	0x0010	Strand of the query (0 for forward; 1 for reverse strand)	Both
000000100000	32	0x0020	Strand of the query's mate	Pairs only
000010000000	64	0x0040	The query is the first read in the pair	Pairs only
000100000000	128	0x0080	The read is the second read in the pair	Pairs only
001000000000	256	0x0100	The alignment is not primary	Both
010000000000	512	0x0200	The read fails platform/vendor quality checks	Both
100000000000	1024	0x0400	The read is either a PCR duplicate or an optical duplicate	Both

SAM Flag

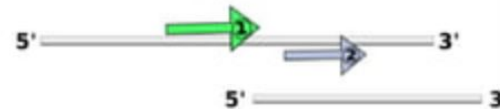
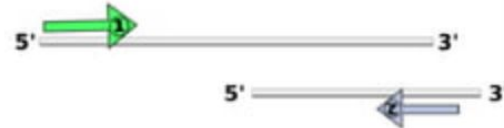


SAM Flag

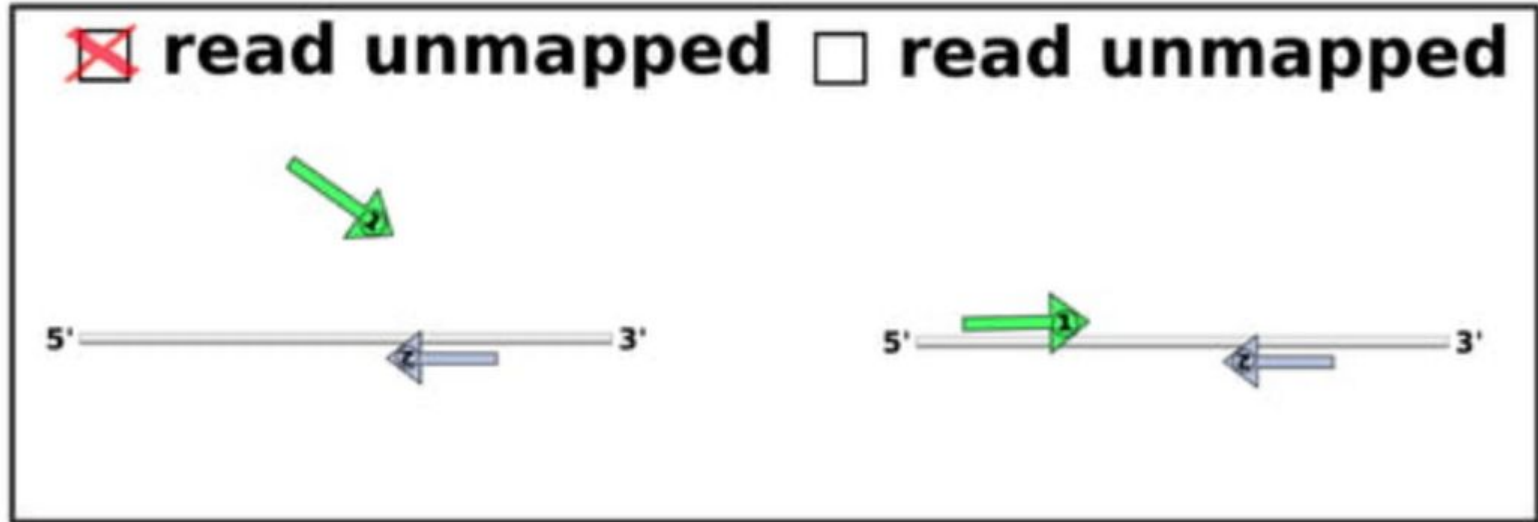
☒ read mapped
in proper pair



☐ read mapped
in proper pair



SAM Flag

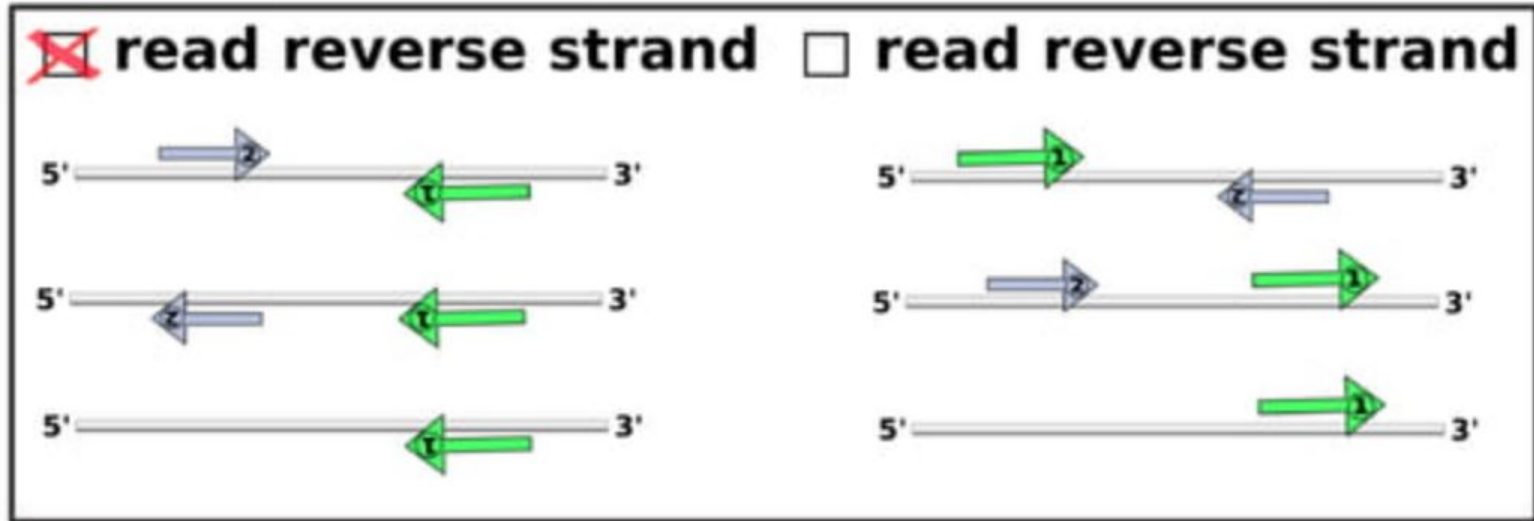


SAM Flag

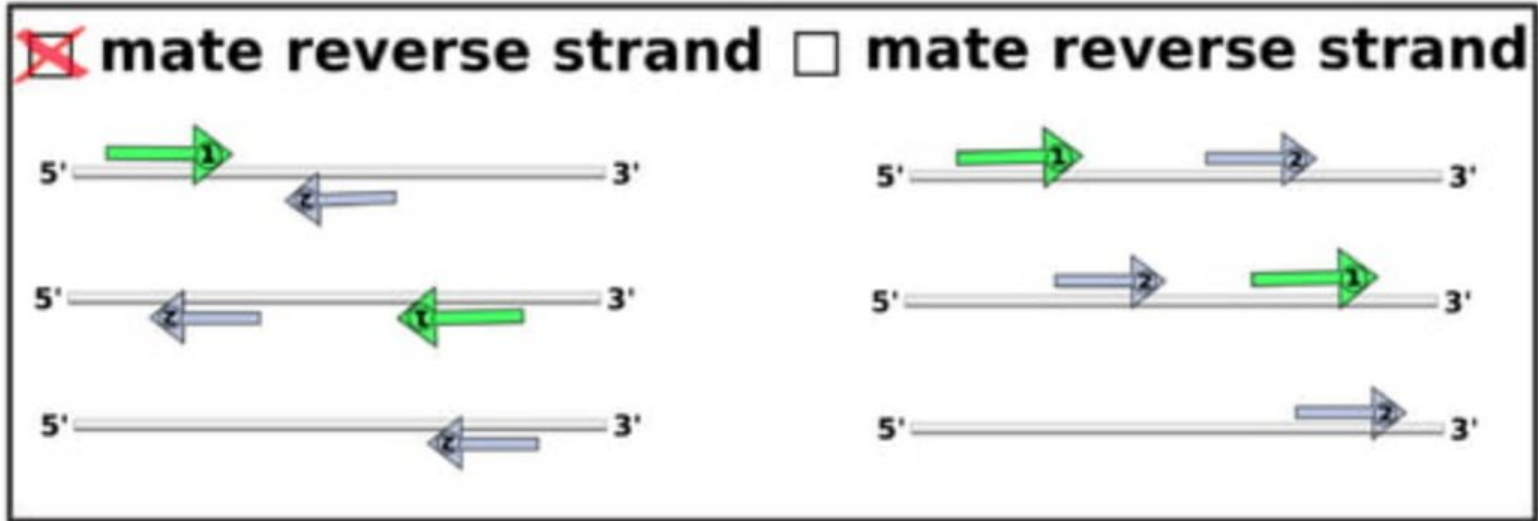
☒ mate unmapped ☐ mate unmapped



SAM Flag



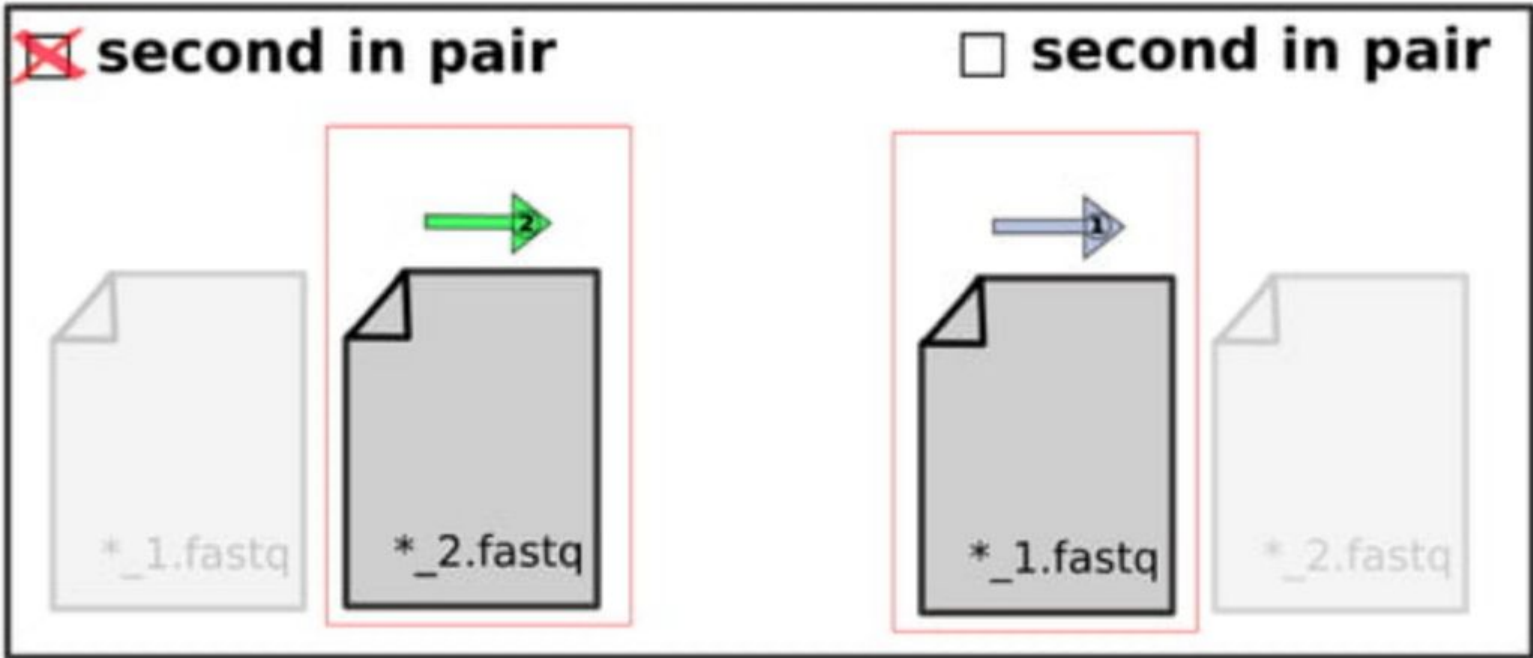
SAM Flag



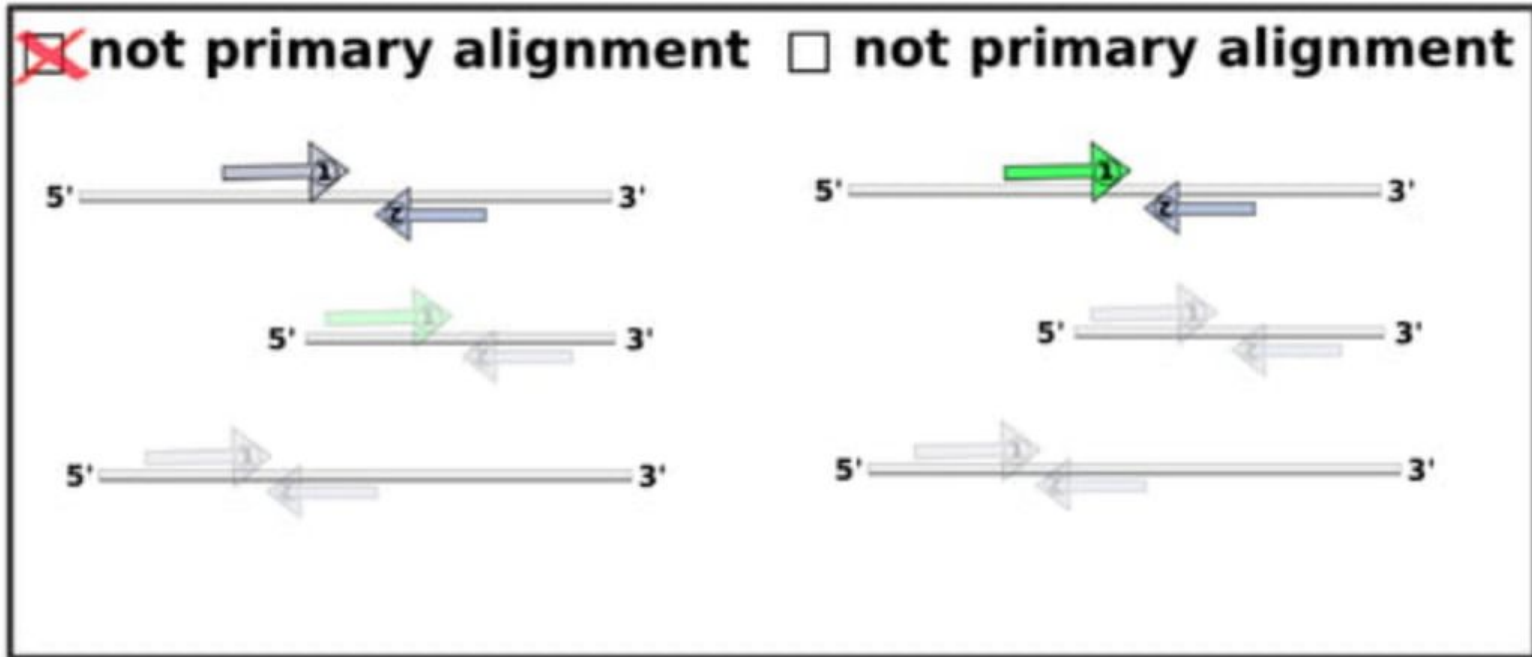
SAM Flag



SAM Flag

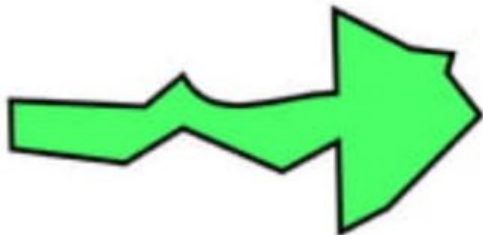


SAM Flag

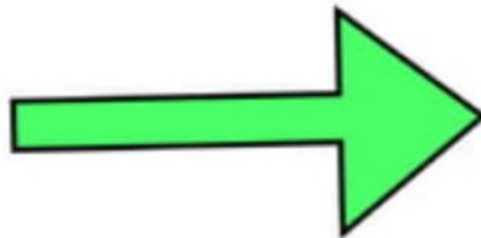


SAM Flag

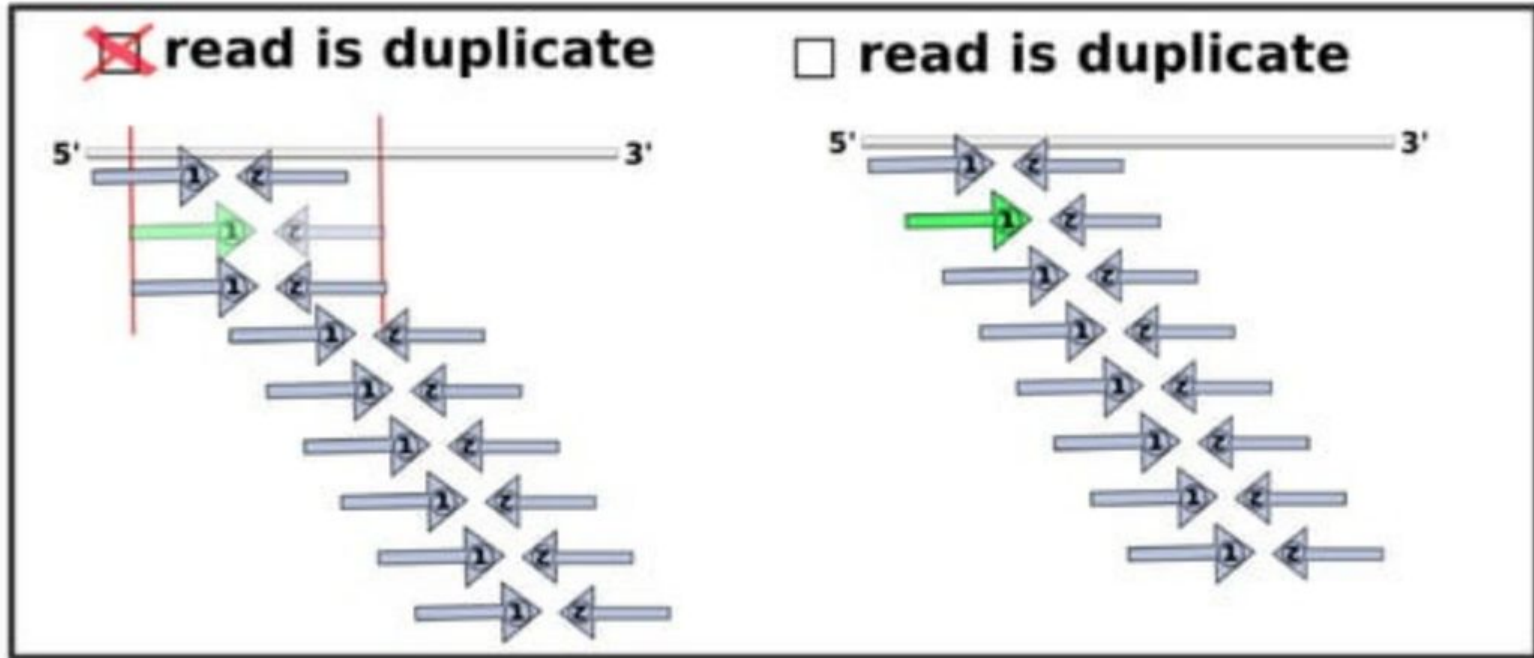
☒ read fails platform
quality checks



☐ read fails platform
quality checks



SAM Flag



ST-E00223:32:H5J57CCXX:4:1220:14651:8868 99 1 10086

base2	base10	base16	Meaning	Applies to:
000000000001	1	0x0001	The read originated from a paired sequencing molecule	Both
000000000010	2	0x0002	The read is mapped in a proper pair	Pairs only
000000000100	4	0x0004	The query sequence itself is unmapped	Both
000000001000	8	0x0008	The query's mate is unmapped	Pairs only
000000010000	16	0x0010	Strand of the query (0 for forward; 1 for reverse strand)	Both
000000100000	32	0x0020	Strand of the query's mate	Pairs only
000001000000	64	0x0040	The query is the first read in the pair	Pairs only
000010000000	128	0x0080	The read is the second read in the pair	Pairs only
001000000000	256	0x0100	The alignment is not primary	Both
010000000000	512	0x0200	The read fails platform/vendor quality checks	Both
100000000000	1024	0x0400	The read is either a PCR duplicate or an optical duplicate	Both

00001100011

$$2^6 + 2^5 + 2^1 + 2^0 = 64 + 32 + 2 + 1 = 99$$

Decoding SAM flags

<https://broadinstitute.github.io/picard/explain-flags.html>


Concise Idiosyncratic Gapped Alignment Report (CIGAR)

Encoding the details of the alignment

Operation	Meaning
M	Match*
D	Deletion w.r.t. reference
I	Insertion w.r.t. reference
N	Split or spliced alignment
S	Soft-clipping
H	Hard-clipping
P	Padding

Reference:
Experimental:

ACCTGTC--TACCTTACG
ACCT-TCCATACTTTATC



4M 1D 2M 2I 7M 2S

CIGAR string:

4M1D2M2I7M2S



LENGTH/OPERATION

CIGAR Extended

Operation	Meaning
=	Exact match
X	Mismatch
D	Deletion w.r.t. reference
I	Insertion w.r.t. reference
N	Split or spliced alignment
S	Soft-clipping
H	Hard-clipping
P	Padding

Reference:

Experimental:

ACCTGTC--TACCTTACG

ACCT-TCCATACTTTATC



4= 1D 2= 2I 3= 1X 3= 2S

CIGAR string:

4=1D2=2I3=1X3=2S

SAM to BAM

Do it once

Create BWT of reference genome.

```
$ bwa index grch38.fa
```



Output is in
SAM format

Align paired-end FASTQ to BWT index.

```
$ bwa mem -t 16 grch38.fa 1.fq 2.fq > sample.sam
```



Output is in BAM format.

Unsorted!
random genomic order as
reads are randomly placed
in FASTQ by sequencer.

Convert SAM to BAM

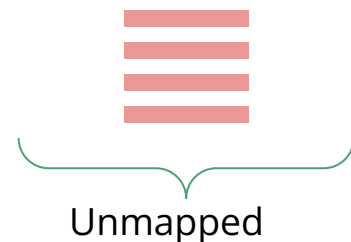
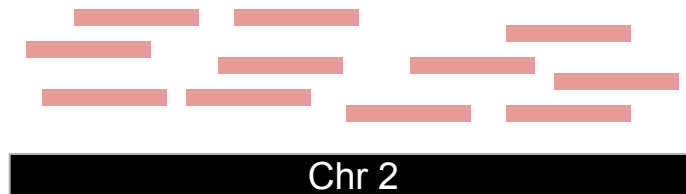
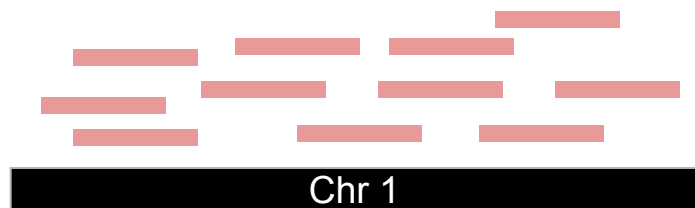
```
$ samtools view -b sample.sam > sample.bam
```

SAM - unmapped reads

- A read appears even if it is unmapped!
- Unmapped reads have:
 - flag 4
 - MAPQ 0
 - Missing info for other fields (* or 0)

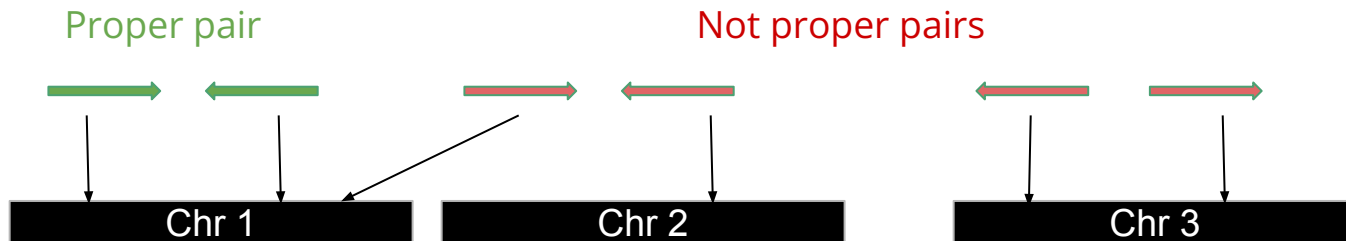
What could
cause a read to
be unmapped?

```
SRR1569760.7733303      77      *      0      0      *      *      0      0      AAAAAAGAGTTTCATTACGGGAAATAAGTTTTCCCTTTTTC
CGATATTATAGCGAAAAGGTTATCTCTCTTATTGTGCATTGTTGTGCCACTACCATA  @@@DDDD:DDHHBGHGIBDFGIHGBGEHIGBDFEFGGCHGHGGEEH>FGIIIIIG@EEE>CCC>>ACCCDC
CCCCC@D3@>@A@4>8<?@ACCCCCCCC  AS:i:0  XS:i:0
```



SAM - paired-end data

- Both reads of a pair appear (as separate records)
- Records contain information about the paired read
- Several flags relate to paired information
- Especially flag 2 - “Read mapped in proper pair”



BAM & CRAM - binary SAM

- Compressed - smaller size
- Faster to read by a computer
- Impossible to read by humans - must use some conversion tool
- Required by some bioinformatic tools
- CRAM might contain only varitional changes from reference

Working with SAM files

- SAM files are text files
- You can view them - use `less`
- You can use Linux commands to manipulate the file:

E.g.: get first 5 alignments:

```
grep -v '^@' aln.sam | head -5
```

- Or you can use a dedicated command line tool - **samtools**

Samtools allows you to...

- View SAM and BAM files
- Select records that satisfy some criteria
- Convert between formats
- Manipulate SAM/BAM files
 - Sorting
 - Indexing
- Extract statistics

Running samtools

```
$ samtools
Program: samtools (Tools for alignments in the SAM format)
Version: 1.9 (using htslib 1.9)
Usage:  samtools <command> [options]
```

- samtools features many commands
- Each command has its own function and options
- Today we'll look at three commands:
 - samtools view
 - samtools sort
 - samtools index

samtools view - viewing files

View sam/bam

```
$ samtools view aln.bam | less
```

View sam/bam including header

```
$ samtools view -h aln.bam | less
```

View just header of sam/bam

```
$ samtools view -H aln.bam | less
```

Samtools sort

Sort a bam file by location

```
$ samtools sort aln.bam > aln.sort.bam
```

Sort a bam file by read name

```
$ samtools sort -n aln.bam > aln.sort_name.bam
```

Samtools index

- Useful for quick handling of a bam file
- Takes a while for large bam files
- Required by some software tools
- Only works on **sorted BAM** files
- Creates a new .bai file

```
$ samtools index aln.sort.bam
```

samtools view - filter records

Only print records with MAPQ ≥ 20

```
$ samtools view -q 20 aln.bam | less
```

Only print records with third bit enabled (unmapped)

```
$ samtools view -f 4 aln.bam | less
```

Only print records with third bit disabled (mapped)

```
$ samtools view -F 4 aln.bam | less
```

samtools view - filter records

Only print records mapped to chromosome 3 (indexed files only)

```
$ samtools view aln.bam chr03 | less
```

Only print records mapped to chromosome 3 positions 1000-2000 (indexed files only)

```
$ samtools view aln.bam chr03:1000-2000 | less
```

Random access requires sorted and indexed bam

Samtools view - converting formats

Convert sam to bam

```
$ samtools view -bh aln.sam > aln.bam
```

Convert bam to sam

```
$ samtools view -h aln.bam > aln.sam
```

Combine with filtration

```
$ samtools view -bh -q 30 -f 2 aln.sam > aln.HQ.mapped.bam
```

Integrative Genomics Viewer (IGV)

Visualization tool for exploring and analyzing genomic data



