

*“I don’t know what I’m doing next...”*

#### Career Fair Description

## **Spring Job & Internship Fair**

Wednesday, March 8, 2023, from 12-3:00pm, Davis Center- 4th floor.

Our annual Spring Job & Internship Fair invites recruiters from regional and national organizations to campus to connect with you about job and internship opportunities! This is our largest career fair of the semester and includes organizations from all industries who want to recruit Catamounts.

#### **The fair is open to all majors and class years and has opportunities for everyone!**

- Explore what companies/organizations are hiring UVM students.
- Connect with recruiters to learn more about their company and what they offer.
- Discover what job/s and internship/s might be of interest to you now and in the future.
- Demonstrate the skills and abilities that make you a unique and successful candidate.

Prepare for Fair Day: Drop in at the Career Center Monday through Thursday between 10am-2pm to get feedback on your application materials or get questions about the career fair answered.

We look forward to seeing you this spring!

# Part-time summer opportunities



## Vermont Adaptive Ski and Sports

📍 Killington, Vermont, United States

🔗 <http://www.vermontadaptive.org>

☑ Hiring all majors

Vermont Adaptive Ski & Sports is a nationally recognized organization that empowers people of all abilities through inclusive sports and recreational programming regardless of ability to pay. In addition to sports, year-round programming options integrate environmental, holistic wellness, and competitive training philosophies for people of all ages with cognitive, developmental, physical and emotional disabilities.

### Job Titles

Summer Adaptive Sports Internship- Burlington, Full Time, Summer Adaptive Sports Internship-Burlington, Part-time, Summer Adaptive Sports Internship-Central Vermont, Full-time

### Job Type

Internship

### Employment Type

Full-Time, Part-Time

### Work Authorization

No US work authorization required

[Hide Details ^](#)

[Follow](#)



## Lake Champlain Maritime Museum

📍 4472 Basin Harbor Rd, Vergennes, Vermont 05491, ...

🔗 <http://www.lcmm.org>

Lake Champlain Maritime Museum is a year-round hub for Maritime Education. We use the active discovery and care of Lake Champlain's maritime heritage and environment as a launching pad to inspire life-long learning through hands-on, minds-on experiences. Interns at the Museum apply skills acquired through their university curriculum to work with archaeologists, curators, and educators. Together, we bring Lake Champlain's storied past to life through replica ships, active boat building, on-water ecology programs, nautical archaeology, collections and exhibits, and cultural heritage events to expand learning opportunities for people of all ages.

### Job Titles

Summer Camp Counselor

### Work Authorization

US work authorization is required, but the employer is willing to sponsor candidates

[Hide Details ^](#)

[Follow](#)

# *For those interested in conservation...*



## Vermont Department of Environmental Conservation

📍 National Life Drive, Davis 1, Montpelier, VT 05620, U...

🔗 <https://dec.vermont.gov/administration-innovation/e...>

✓ Hiring all majors

The Vermont Department of Environmental Conservation issues most of the state's environmental permits, sets forth regulations and assures compliance. Through its programs, the DEC manages water and air quality, regulates solid and hazardous wastes, administers a number of voluntary pollution and waste reduction programs. Among other responsibilities, positions in the Department collect data, conduct research, run volunteer programs, develop educational and outreach materials and programs, administer grants , work with conservation organizations and state and federal agencies to examine critical environmental issues. ECO AmeriCorps is a national service program administered by the DEC that places 24 full time members at non-profit and local and state government host sites through out the state of Vermont. Members serve an 11 month term (September through August).

### Job Titles

None

### Job Type

Job, Internship, Cooperative Education, Experiential Learning

### Employment Type

Full-Time, Part-Time

### Work Authorization

No US work authorization required

### School Years

Certificate Program, Second Year Community / Technical College, First Year Community / Technical College, Alumni, Masters of Business Administration, Accelerated Masters, Masters, Senior, Junior, Sophomore, Freshman

[Hide Details ^](#)

☆ Follow

# *For future health-care workers....*



## Southern Vermont AHEC

📍 368 River Street, Springfield, Vermont 05156, United...

🔗 <http://www.svtahec.org>

☑ Hiring all majors

Southern Vermont AHEC is a nonprofit health care workforce development organization. We place interns in southern Vermont with healthcare organizations including medical, behavioral, mental, prevention and public health. SVTAHEC also runs programs in Vermont to help students learn about careers in healthcare and to promote primary care and rural health.

### Job Titles

Healthcare Intern: College Student Healthcare Internship Program, CSHIP

### Job Type

Internship

### Employment Type

Part-Time

### Work Authorization

US work authorization is required

### School Years

Certificate Program, Second Year Community / Technical College, First Year Community / Technical College, Alumni, Senior, Junior, Sophomore

[Hide Details ^](#)

☆ Follow

# Hey, Life Science Majors!!!

*Are you interested in a career in the biotechnology industry  
but don't know where to get experience?*



**Agilent's Sales Development Team is offering  
paid 12-month internships starting in  
Summer 2023!**

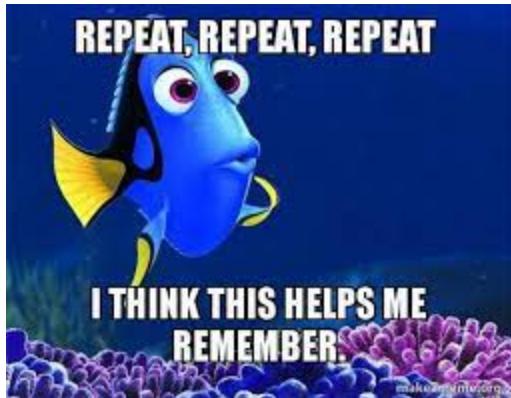
---

Visit the Agilent booth at the UVM Career Fair on  
**March 8, 2023**

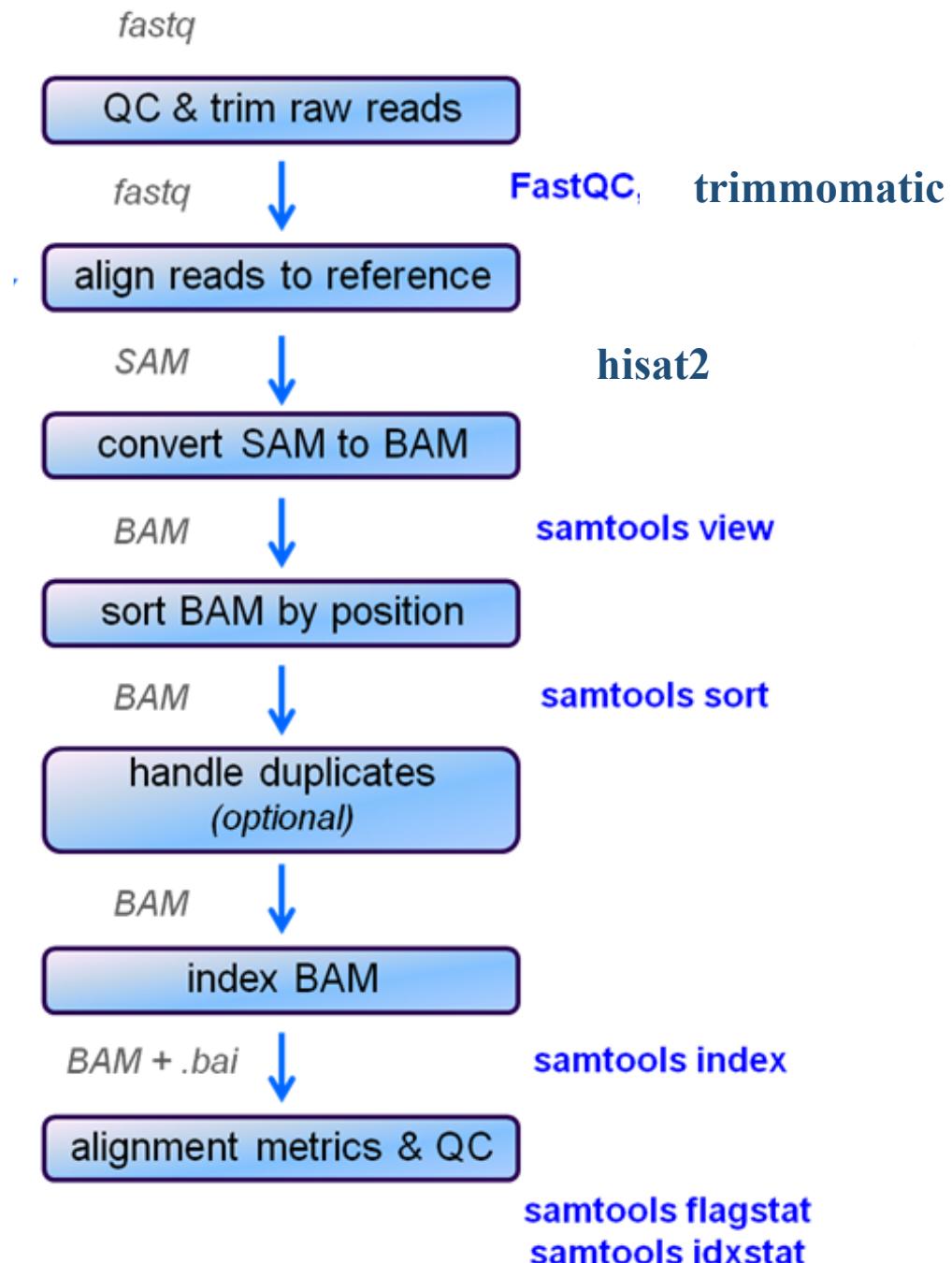
---

For more information about the program, email [Andrea.Lee@Agilent.com](mailto:Andrea.Lee@Agilent.com).  
Agilent is an equal employment opportunity and affirmative action employer.

# Learning the SAM/BAM format



# Alignment Workflow



# hisat2\_align.sh

*This will not be  
“ready-to-go” for  
everyone, you may  
need to add upon it  
for your dataset*

*Basic Template*

```
#!/bin/bash
#SBATCH --partition=bluemoon
#SBATCH --nodes=1
#SBATCH --ntasks=2
#SBATCH --mem=10G
#SBATCH --time=3:00:00
#SBATCH --job-name=align_CD8
# %x=job-name %j=jobid
#SBATCH --output=%x_%j.out

for i in *fastq.gz; do
SAMPLE=$(echo ${i} | sed "s/.fastq.gz//")
echo ${SAMPLE}.fastq.gz

DBDIR=/gpfs1/cl/mmg232/course_materials/hisat2_index
GENOME="GRCh39"
p=2

module load hisat2-2.1.0-gcc-7.3.0-knvgwpc
module load samtools-1.10-gcc-7.3.0-pdbkohx

#align to GRCh39
hisat2 \
-p ${p} \
-x ${DBDIR}/${GENOME} \
-U ${SAMPLE}.fastq.gz \
-S ${SAMPLE}.sam >& ${SAMPLE}.log

#create bam file
samtools view ${SAMPLE}.sam \
--threads 2 \
-b \
-o ${SAMPLE}.bam \
```

1

2

3

4



# Read the methods

---

- Did the authors add special arguments during alignment?
- You need to understand why this was done.
- If you do not understand – email me and we can chat

# Part 1: Provide the job submission parameters

Our script will be written in sections:

1. Will provide the job submission parameters

```
```
#!/bin/bash
#SBATCH --partition=bluemoon
#SBATCH --nodes=1
#SBATCH --ntasks=2
#SBATCH --mem=40G
#SBATCH --time=24:00:00
#SBATCH --job-name=align_CD8
# %x=job-name %j=jobid
#SBATCH --output=%x_%j.out
```
```

[https://prodriuez19.github.io/Intro-to-rnaseq/lessons/05\\_Mapping\\_with\\_HISAT2.html](https://prodriuez19.github.io/Intro-to-rnaseq/lessons/05_Mapping_with_HISAT2.html)

# Part 2: To keep the naming convention for each file output by this script

```
```bash
for i in reads/*.fastq      Beginning of for loop
do
    SAMPLE=$(echo ${i} | sed "s/.fastq//")
    echo ${SAMPLE}.fastq
```

```

After running the hisat2\_align.sh script with the FASTQ provided, the outputs will look like this:

```
```
align_CD8_7422840.out  SRR13423162.log          SRR13423165.fastq.gz
hisat2_align.sh         SRR13423162_sorted.bam    SRR13423165.log
hisat2.log               SRR13423162_sorted.bam.bai SRR13423165_sorted.bam
SRR13423162.bam         SRR13423162.txt          SRR13423165_sorted.bam.bai
SRR13423162.fastq.gz    SRR13423165.bam          SRR13423165.txt
```
```

```

# Part 3: Load the modules required to run the commands

```
```
module load hisat2-2.1.0-gcc-7.3.0-knvgwpc
module load samtools-1.10-gcc-7.3.0-pdbkohx
```
```

# Part 4: The *actual* commands to be executed

```
#align to GRCm39  
hisat2 \  
-p ${p} \  
-x ${DBDIR}/${GENOME} \  
-U ${SAMPLE}.fastq.gz \  
-S ${SAMPLE}.sam &> ${SAMPLE}.log
```

Converts FASTQ to SAM

```
#create bam file  
samtools view ${SAMPLE}.sam \  
--threads 2 \  
-b \  
-o ${SAMPLE}.bam \  
  
#remove sam files once bam file is created  
rm ${SAMPLE}.sam
```

```
#output stats  
samtools flagstat ${SAMPLE}.bam > ${SAMPLE}.txt
```

```
# sort the bam file based on coordinates  
samtools sort ${SAMPLE}.bam -o ${SAMPLE}_sorted.bam
```

```
# index bam file  
samtools index ${SAMPLE}_sorted.bam
```

```
done &> hisat2.log  
```
```

SAM to BAM  
+ *more stuff*

# SAMtools usage

- <http://www.htslib.org/doc/samtools.html>

samtools view [*options*] *in.sam|in.bam|in.cram* [*region...*]

samtools [sort](#) [-l *level*] [-u] [-m *maxMem*] [-o *out.bam*] [-O *format*] [-M] [-K *kmerLen*] [-n] [-t *tag*] [-T *tmpprefix*] [-@ *threads*] [*in.sam|in.bam|in.cram*]

samtools index [-bc] [-m *INT*] *aln.sam|aln.bam|aln.cram* [*out.index*]

# Read alignments files: the SAM format

- ‘Sequence Alignment/Map’ format -  
<http://samtools.sourceforge.net/SAMv1.pdf>
- SAM/BAM files can be manipulated with SAMtools
- SAM files are tab-delimited files, human-readable
- The SAM file contains two sections:
  1. Header section:
    - Metadata about the genome, the samples, the pipeline
    - Header lines start with @
  2. Alignments (or ‘records’) section

# To view a SAM file:

```
module load samtools-1.10-gcc-7.3.0-pdbkohx
```

```
samtools view -h SRR13423162_sorted.bam | less -S
```

# SAM header descriptions

| Tag              | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>@HD</code> | The header line. The first line if present.                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>VN*</code> | Format version. Accepted format: <code>/^ [0-9]+\. [0-9]+\$/</code> .                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>SO</code>  | Sorting order of alignments. Valid values: <code>unknown</code> (default), <code>unsorted</code> , <code>queryname</code> and <code>coordinate</code> . For coordinate sort, the major sort key is the RNAME field, with order defined by the order of <code>@SQ</code> lines in the header. The minor sort key is the POS field. For alignments with equal RNAME and POS, order is arbitrary. All alignments with '*' in RNAME field follow alignments with some other value but otherwise are in arbitrary order. |
| <code>GO</code>  | Grouping of alignments, indicating that similar alignment records are grouped together but the file is not necessarily sorted overall. Valid values: <code>none</code> (default), <code>query</code> (alignments are grouped by QNAME), and <code>reference</code> (alignments are grouped by RNAME/POS).                                                                                                                                                                                                           |
| <code>@SQ</code> | Reference sequence dictionary. The order of <code>@SQ</code> lines defines the alignment sorting order.                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>SN*</code> | Reference sequence name. The SN tags and all individual AN names in all <code>@SQ</code> lines must be distinct. The value of this field is used in the alignment records in RNAME and RNEXT fields. Regular expression: <code>[!-~]+-&lt;&gt;-[!-~]*</code>                                                                                                                                                                                                                                                        |
| <code>LN*</code> | Reference sequence length. Range: <code>[1,2<sup>31</sup>-1]</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <code>AH</code>  | Indicates that this sequence is an alternate locus. <sup>4</sup> The value is the locus in the primary assembly for which this sequence is an alternative, in the format ' <code>chr:start-end</code> ', ' <code>chr</code> ' (if known), or '*' (if unknown), where ' <code>chr</code> ' is a sequence in the primary assembly. Must not be present on sequences in the primary assembly.                                                                                                                          |
| <code>AN</code>  | Alternative reference sequence names. A comma-separated list of alternative names that tools may use when referring to this reference sequence. <sup>5</sup> These alternative names are not used elsewhere within the SAM file; in particular, they must not appear in alignment records' RNAME or RNEXT fields. Regular expression: <code>name(,name)*</code> where <code>name</code> is <code>[0-9A-Za-z] [0-9A-Za-z**. @_-]*</code>                                                                             |
| <code>AS</code>  | Genome assembly identifier.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <code>M5</code>  | MD5 checksum of the sequence. See Section 1.3.1                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <code>SP</code>  | Species.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>UR</code>  | URI of the sequence. This value may start with one of the standard protocols, e.g http: or ftp:. If it does not start with one of these protocols, it is assumed to be a file-system path.                                                                                                                                                                                                                                                                                                                          |
| <code>@RG</code> | Read group. Unordered multiple <code>@RG</code> lines are allowed.                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>ID*</code> | Read group identifier. Each <code>@RG</code> line must have a unique ID. The value of ID is used in the RG tags of alignment records. Must be unique among all read groups in header section. Read group IDs may be modified when merging SAM files in order to handle collisions.                                                                                                                                                                                                                                  |
| <code>CN</code>  | Name of sequencing center producing the read.                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>DS</code>  | Description.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <code>DT</code>  | Date the run was produced (ISO8601 date or date/time).                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>FO</code>  | Flow order. The array of nucleotide bases that correspond to the nucleotides used for each flow of each read. Multi-base flows are encoded in IUPAC format, and non-nucleotide flows by various other characters. Format: <code>/\*  [ACMGRSVTWYHKDBN]+/</code>                                                                                                                                                                                                                                                     |

# SAM alignment section

cf. FASTQ format

| Read Name              | FLAG      | Chrom | AlnStart | CIGAR    |   |     | Sequence | BaseQuals                 |
|------------------------|-----------|-------|----------|----------|---|-----|----------|---------------------------|
| 6_1303_10584_85775 99  | groupVIII | 311   | 3        | 63M3I34M | = | 780 | 572      | GGGTATTGGGC @CFFFFFH      |
| 6_1111_20943_90813 163 | groupVIII | 315   | 40       | 100M     | = | 809 | 594      | TAATGAAGCCAT @BDDFDDA+<A< |
| 6_2111_2016_88235 355  | groupVIII | 315   | 3        | 100M     | = | 856 | 573      | TAATGAAGCCAT @?DADDBD>D>B |
| 6_1104_8139_99999 163  | groupVIII | 316   | 14       | 100M     | = | 818 | 602      | AATGAAGCCATT @@FFFFFGHGHH |
| 6_1304_4167_91751 163  | groupVIII | 322   | 5        | 52M3I29M | = | 812 | 573      | GCCATTTTAC <<BDBDEHHDF    |
| 6_2301_14383_16382 163 | groupVIII | 323   | 40       | 51M3I46M | = | 809 | 589      | CCATTTTACT CCFFFFFFHHHH   |

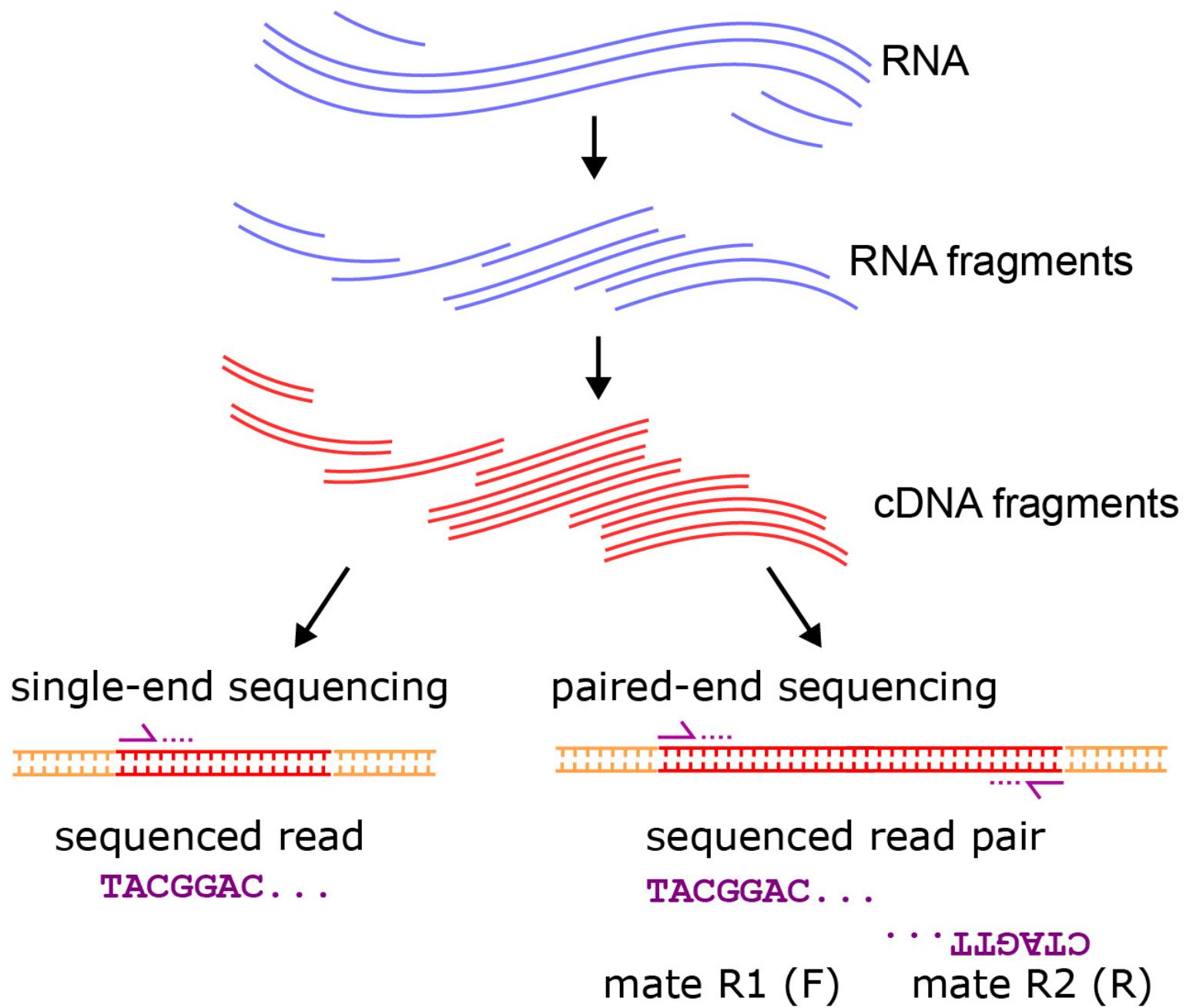
| Col | Field | Type   | Regexp/Range                             | Brief description                     |
|-----|-------|--------|------------------------------------------|---------------------------------------|
| 1   | QNAME | String | [!-?A-~]{1,255}                          | Query template NAME                   |
| 2   | FLAG  | Int    | [0,2 <sup>16</sup> -1]                   | bitwise FLAG                          |
| 3   | RNAME | String | \*  [!-()+-<>-~] [!-~]*                  | Reference sequence NAME               |
| 4   | POS   | Int    | [0,2 <sup>31</sup> -1]                   | 1-based leftmost mapping POSition     |
| 5   | MAPQ  | Int    | [0,2 <sup>8</sup> -1]                    | MAPping Quality                       |
| 6   | CIGAR | String | \*  ([0-9]+[MIDNSHPX=])+                 | CIGAR string                          |
| 7   | RNEXT | String | \* =  [!-()+-<>-~] [!-~]*                | Ref. name of the mate/next read       |
| 8   | PNEXT | Int    | [0,2 <sup>29</sup> -1]                   | Position of the mate/next read        |
| 9   | TLEN  | Int    | [-2 <sup>29</sup> +1,2 <sup>29</sup> -1] | observed Template LENGTH              |
| 10  | SEQ   | String | \*  [A-Za-z.=.]+                         | segment SEQuence                      |
| 11  | QUAL  | String | [!-~]+                                   | ASCII of Phred-scaled base QUALity+33 |

# Column 2: Bitwise Flag

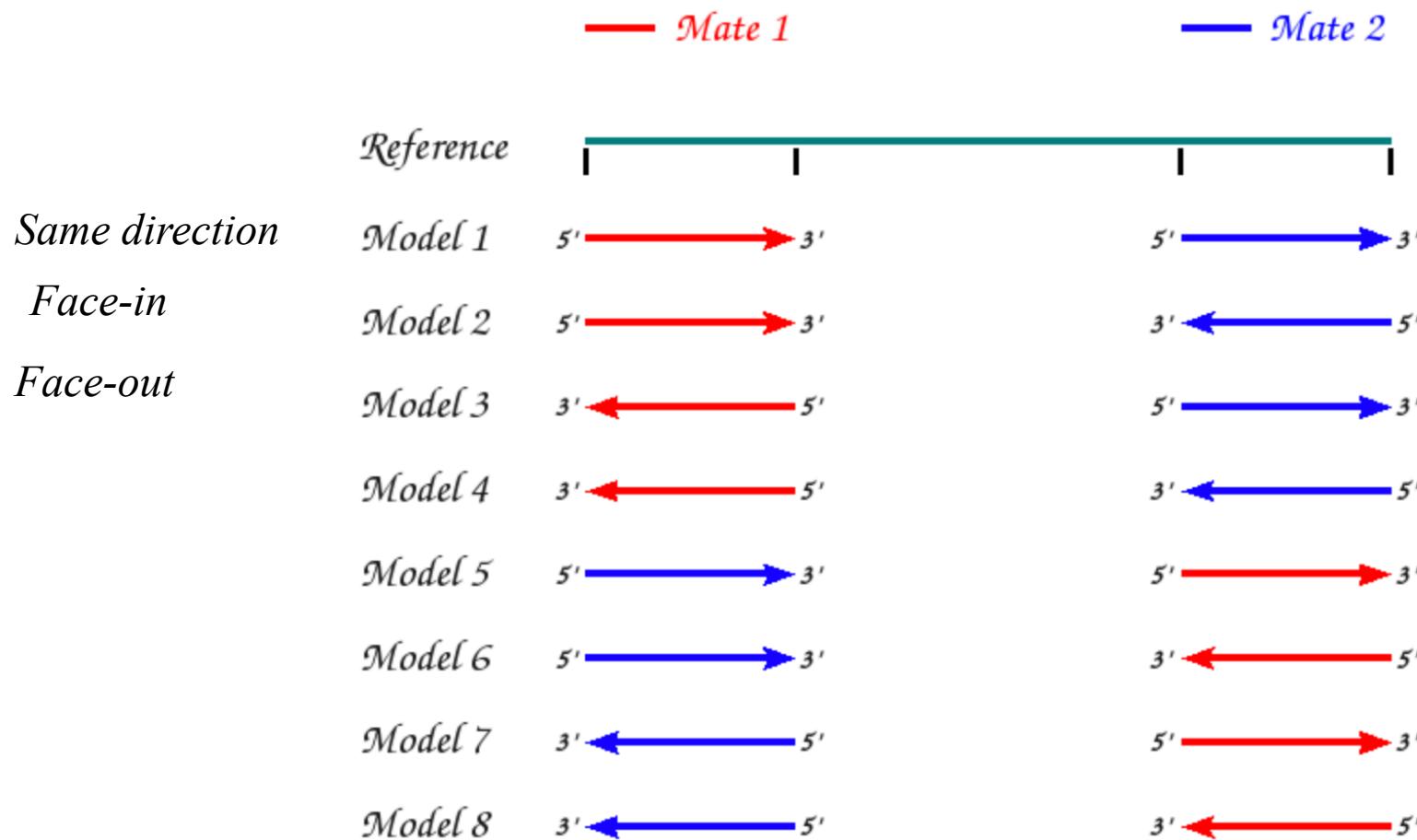
- Is a lookup code to explain certain features about the particular read
- It tells you whether the read aligned, is marked as a PCR duplicate, if its mate aligned, etc.

| Bit  | Description                                                             |
|------|-------------------------------------------------------------------------|
| 1    | 0x1 template having multiple segments in sequencing                     |
| 2    | 0x2 each segment properly aligned according to the aligner              |
| 4    | 0x4 segment unmapped                                                    |
| 8    | 0x8 next segment in the template unmapped                               |
| 16   | 0x10 SEQ being reverse complemented                                     |
| 32   | 0x20 SEQ of the next segment in the template being reverse complemented |
| 64   | 0x40 the first segment in the template                                  |
| 128  | 0x80 the last segment in the template                                   |
| 256  | 0x100 secondary alignment                                               |
| 512  | 0x200 not passing filters, such as platform/vendor quality controls     |
| 1024 | 0x400 PCR or optical duplicate                                          |
| 2048 | 0x800 supplementary alignment                                           |

*A combination of the flags, results in one integer, which makes it difficult to interpret*



# “Proper” mate-pairing



A combination of the flags, results in one integer, which makes it difficult to interpret

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

# Column 6: ‘CIGAR strings’

|                        |           |     |    |          |   |     |     |                           |
|------------------------|-----------|-----|----|----------|---|-----|-----|---------------------------|
| 6_1303_10584_85775 99  | groupVIII | 311 | 3  | 63M3I34M | = | 780 | 572 | GGGTATTGGGC @CFFFFFH      |
| 6_1111_20943_90813 163 | groupVIII | 315 | 40 | 100M     | = | 809 | 594 | TAATGAAGCCAT @BDDFDDA+<A< |
| 6_2111_2016_88235 355  | groupVIII | 315 | 3  | 100M     | = | 856 | 573 | TAATGAAGCCAT @?DADDBD>D>B |
| 6_1104_8139_99999 163  | groupVIII | 316 | 14 | 100M     | = | 818 | 602 | AATGAAGCCATT @@FFFFFGHGHH |
| 6_1304_4167_91751 163  | groupVIII | 322 | 5  | 52M3I29M | = | 812 | 573 | GCCATTTTAC <<bdbdehhdf    |
| 6_2301_14383_16382 163 | groupVIII | 323 | 40 | 51M3I46M | = | 809 | 589 | CCATTTTACT CCFFFFFFHHH    |

| Op | BAM | Description                                           |
|----|-----|-------------------------------------------------------|
| M  | 0   | alignment match (can be a sequence match or mismatch) |
| I  | 1   | insertion to the reference                            |
| D  | 2   | deletion from the reference                           |
| N  | 3   | skipped region from the reference                     |
| S  | 4   | soft clipping (clipped sequences present in SEQ)      |
| H  | 5   | hard clipping (clipped sequences NOT present in SEQ)  |
| P  | 6   | padding (silent deletion from padded reference)       |
| =  | 7   | sequence match                                        |
| X  | 8   | sequence mismatch                                     |

100M — 100 matching nucleotides (i.e. no gaps)

63M-3I-34M — 63 matching nucleotides

3 nucleotides not in the reference (3bp insertion)

34 matching nucleotides

## Aligned Read

TGCAGGGATGGATGTGTCCTCCTCAGCTGCTTATTTAACCTCCACTGCACAAACATGTTTGTTATATTCTTCGCTGTAGTCGTGAAGC

TGCAGGGACTGCAGGGATGGATGTGTCCTCCTCAGCTGCTTATTTAACCTCCAC---ACAACATGTTTGTTATATTCTTCGCTGTAGTCGTGAAGCAGAGTATGATACTG

## Reference

# Column 6: ‘CIGAR strings’

| Op | BAM | Description                                           |
|----|-----|-------------------------------------------------------|
| M  | 0   | alignment match (can be a sequence match or mismatch) |
| I  | 1   | insertion to the reference                            |
| D  | 2   | deletion from the reference                           |
| N  | 3   | skipped region from the reference                     |
| S  | 4   | soft clipping (clipped sequences present in SEQ)      |
| H  | 5   | hard clipping (clipped sequences NOT present in SEQ)  |
| P  | 6   | padding (silent deletion from padded reference)       |
| =  | 7   | sequence match                                        |
| X  | 8   | sequence mismatch                                     |

- Example: intron = 81 bases

ERR022486.8388510 81 22 32099 255 **58M81N18M** = 27484 -4772  
CCTTGGTCTTGCCGAAGTAGATCTCATTGAGAGTGGAGCGGATCTGTTCTCCATTCCCTCCA  
CCAGGCGTCCGAT :9=<==;<><=><?>>?<?==>>?>><?>>??<AA?  
@AFADD;GDGAG@GGCBE@GG?GG>GGGG?GGGGGGGG NM:i:0 XS:A:- NH:i:1

**Aligned Read** 58M 81N 18M *Spliced*

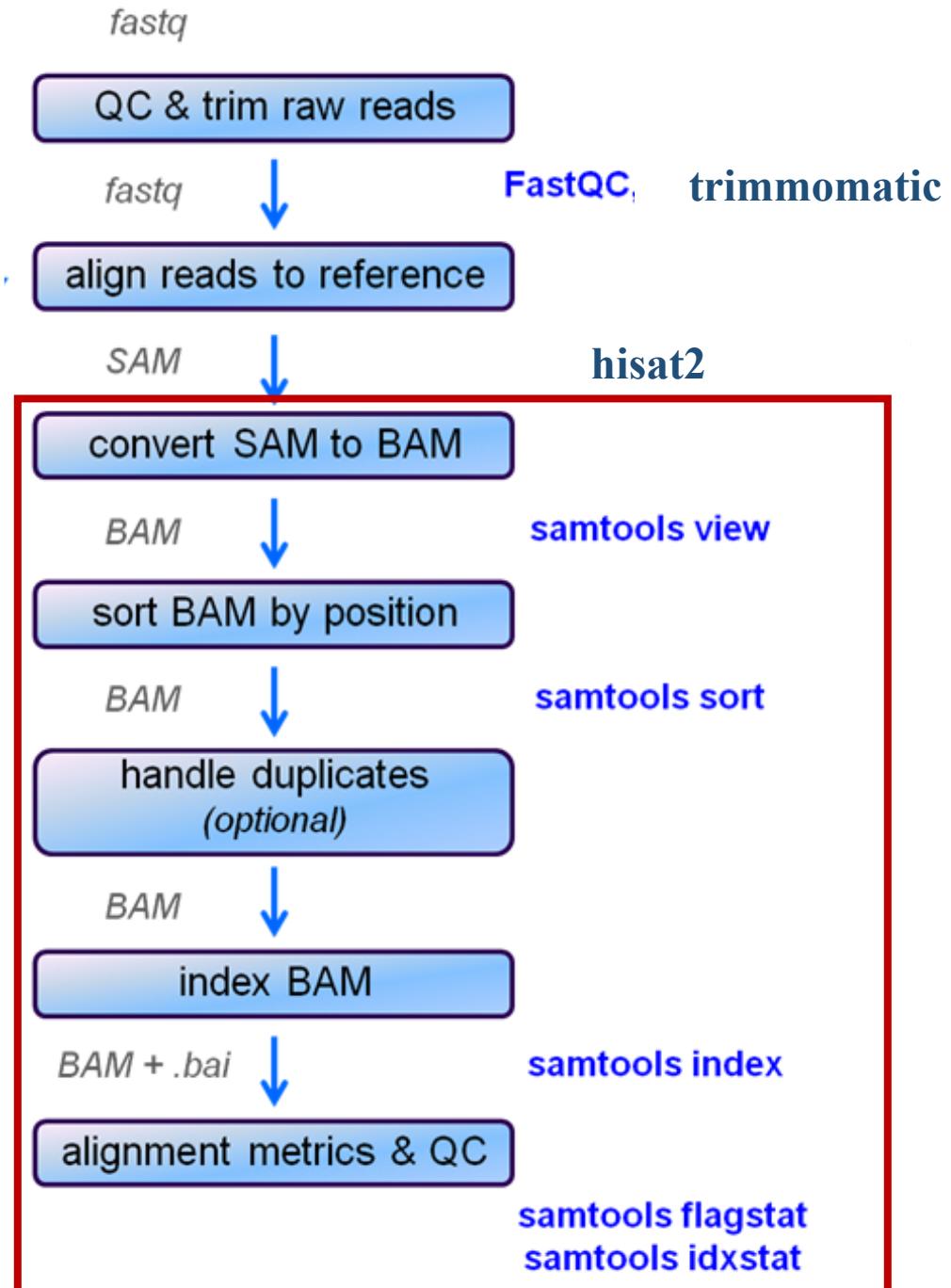
TGCAGGGACTGCAGGATGGATGTGTTCCCTCAGCTGCTTA-----TATATTCTTCGCTGTAGTCTGTAAGC

TGCAGGGACTGCAGGATGGATGTGTTCCCTCAGCTGCTTA**TTTTA**CTCCACACAACATGTTTG**GT**TATATTCTTCGCTGTAGTCTGTAAGCAGAGTATGATAAC

**Reference**

We never want to  
keep a SAM file -  
we immediately  
convert it to a BAM  
file

## Alignment Workflow



# BAM file

- BAM (Binary Alignment/Map) format:
  - ❖ Compressed binary representation of SAM
  - ❖ Greatly reduces storage space requirements to about ~27% of original SAM
  - ❖ Not human-readable

# Common order of operations

1. SAM files are converted to BAM files (*samtools view*)
2. BAM files are sorted by reference coordinates (*samtools sort*)
3. SORTED BAM files are indexed (*samtools index*)

# samtools view

```
 samtools view -b input.sam > input.bam
```

- Input is usually a SAM file, but can also use a BAM
- Common uses: extracting a subset of data into a new file, converting between SAM/BAM files, or just viewing raw files

# samtools sort

```
 samtools sort sample.bam -o sample.sorted.bam
```

- Reads need to be ordered in “genomic order” – not the order in which they were sequenced

# samtools index

```
 samtools index sorted.bam
```

- Creates index file that allows for fast look-up
- Generates \*.bam.bai file

# Summary

## hisat2\_align.sh

*This will not be  
“ready-to-go” for  
everyone, you may  
need to add upon it  
for your dataset*

***Basic Template***

```
#!/bin/bash
#SBATCH --partition=bluemoon
#SBATCH --nodes=1
#SBATCH --ntasks=2
#SBATCH --mem=10G
#SBATCH --time=3:00:00
#SBATCH --job-name=align_CD8
# %x=job-name %j=jobid
#SBATCH --output=%x_%j.out

for i in *fastq.gz; do
SAMPLE=$(echo ${i} | sed "s/.fastq.gz//")
echo ${SAMPLE}.fastq.gz

DBDIR=/gpfs1/cl/mmg232/course_materials/hisat2_index
GENOME="GRCh39"
p=2

module load hisat2-2.1.0-gcc-7.3.0-knvgwpc
module load samtools-1.10-gcc-7.3.0-pdbkohx

#align to GRCh39
hisat2 \
-p ${p} \
-x ${DBDIR}/${GENOME} \
-U ${SAMPLE}.fastq.gz \
-S ${SAMPLE}.sam >& ${SAMPLE}.log

#create bam file
samtools view ${SAMPLE}.sam \
--threads 2 \
-b \
-o ${SAMPLE}.bam \
```

1

2

3

4