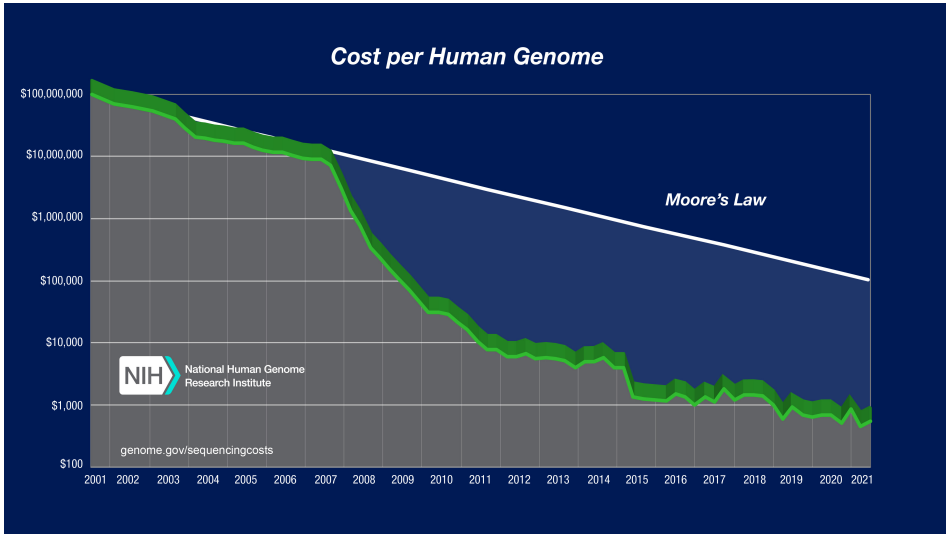
Mapping Sequencing Reads to a Reference Genome



High Throughput Sequencing

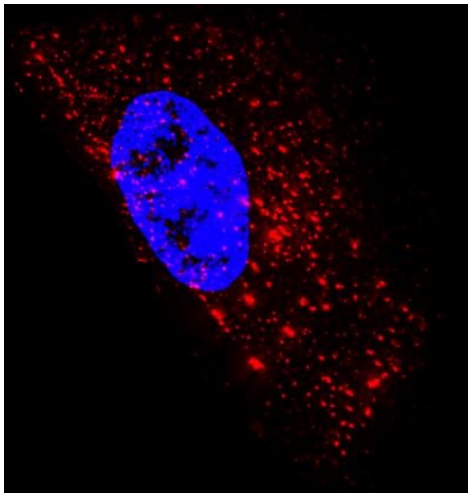
Example applications:

- Sequencing a genome (DNA)
- Sequencing a transcriptome and gene expression studies (RNA)
- ChIP (chromatin immunoprecipitation)

Example platforms:

- 454
- Illumina
- SOLiD

RNA

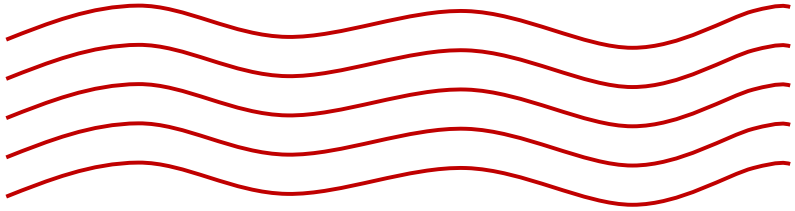


Sequencing Output

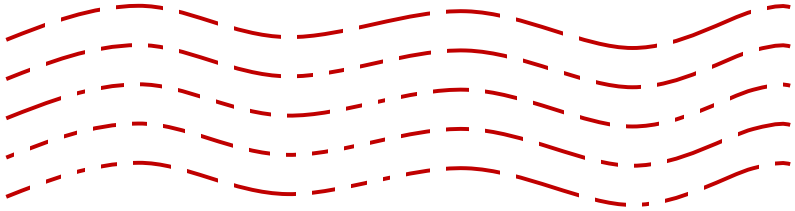
- Hundreds of millions of sequencing reads, each ~200 nts in length
- We need to map each read to the genome, i.e., determine the region of the genome each read corresponds to

```
- ACGTAGTCGTAGTCGCTTACGATTCGTATGCGTACGTGTAGTCTACGTGTA      - GTGCGTAGTCGTATATGCCTAGCATCGTTAGTCTAGCGTAGTCGTACGTAC
- CCGCGCGCTACTCTCGATGTACGTGTCAGTTACGATGTACTGTAGTCAGTGT      - ATCATCGGCGATAGTCTACGTAGTTATATCTACGCGGCGCCCATTCTGCAA
- TTTGATCGTAGTGTCACTGAGCAAACACCATTACTTACTATCTTGGACATC      - CGAGTAGGAGTCGTAGTCGTAGTCGATGCATCGAGTCTGCGATAGTCGTAG
- TGGGGCGATCGAGGATTCTAGTTATCGAGTGTCGGCGATTATGCGATCGAA      - GACAGTGCGGACGTATGTATAGCAGCAGTTACGTACGTAGTTGACGTAGTA
- GGCATCATACGTAGCGTATCGATTAGCAGTGCCGCGCTATACGTCGTCGAT      - CAACTTTCGGCATCTTCCGGTTCTCTCTCTCTAGATAGAGACTTACGATCG
- AGCGCGGGTCTGGACGACGGCTAGCGTCATCGTCGATCGATCGATCGTAAA      - TCCGCGCATCGTAGTCGCGCGCGACTATGCGCGACACACGTGTAGCATACC
- TCGGTCTCGATCGAGTGTCTGCGCGTCTCTGAGAGGCTAGGTAGAGAGCTG      - AGACTGAGTATATCGGCGCGTATGCGACTGTAGCTATATACGGCATGCGTC
- CTCCTCCGAGCGTAGTCGCGATACGTATCGGATCTGACGTATGCAGTCGAT      - GGATCGAGTCATCGTCGATCGAGTCGTAGCAGCAGTCGTAGCTAATAGTCG
- CCGTATGCGTAGTCGCGATCGATCGTGATTTGAGTCGCTAGATTACGTAGT      - GCGATCGTAGTCAGTTCGAGTCATGCGATCTGATGCGAGTACGTAGTCGTA
- TACGGCGACGTTACGCGTAGTCATGCATGCATGAGCCTAGTTGCGTATGCG      - TTATTATCGCGCAGTGTCGTACGTAGTCGTAGTCATCGATGCGTATATACT
- GAAGCGTACATGCTAGTCTACGTCTCGCAGTTCGTATGCGATCGTATGACG      - AGTACGTAGTATCTGAGCGTCTCTTCTACGGACGATCGATCGTACACGTGA
- ACGCACGATCGATCTATGCATCGATGCTAGTCGTATGCTAGTCTATGCGAT      - TTTATTACGACGATACGTGCGCATTGCGTATGCAGTGTCAGTCAGTCGTAG
- ACGTAGTCGTAGTCGCTTACGATTCGTATGCGTACGTGTAGTCTACGTGTA      - GTGCGTAGTCGTATATGCCTAGCATCGTTAGTCTAGCGTAGTCGTACGTAC
- CCGCGCGCTACTCTCGATGTACGTGTCAGTTACGATGTACTGTAGTCAGTGT      - ATCATCGGCGATAGTCTACGTAGTTATATCTACGCGGCGCCCATTCTGCAA
- TTTGATCGTAGTGTCACTGAGCAAACACCATTACTTACTATCTTGGACATC      - CGAGTAGGAGTCGTAGTCGTAGTCGATGCATCGAGTCTGCGATAGTCGTAG
- TGGGGCGATCGAGGATTCTAGTTATCGAGTGTCGGCGATTATGCGATCGAA      - GACAGTGCGGACGTATGTATAGCAGCAGTTACGTACGTAGTTGACGTAGTA
- GGCATCATACGTAGCGTATCGATTAGCAGTGCCGCGCTATACGTCGTCGAT      - CAACTTTCGGCATCTTCCGGTTCTCTCTCTCTAGATAGAGACTTACGATCG
- AGCGCGGGTCTGGACGACGGCTAGCGTCATCGTCGATCGATCGATCGTAAA      - TCCGCGCATCGTAGTCGCGCGCGACTATGCGCGACACACGTGTAGCATACC
- TCGGTCTCGATCGAGTGTCTGCGCGTCTCTGAGAGGCTAGGTAGAGAGCTG      - AGACTGAGTATATCGGCGCGTATGCGACTGTAGCTATATACGGCATGCGTC
- CTCCTCCGAGCGTAGTCGCGATACGTATCGGATCTGACGTATGCAGTCGAT      - GGATCGAGTCATCGTCGATCGAGTCGTAGCAGCAGTCGTAGCTAATAGTCG
- CCGTATGCGTAGTCGCGATCGATCGTGATTTGAGTCGCTAGATTACGTAGT      - GCGATCGTAGTCAGTTCGAGTCATGCGATCTGATGCGAGTACGTAGTCGTA
- TACGGCGACGTTACGCGTAGTCATGCATGCATGAGCCTAGTTGCGTATGCG      - TTATTATCGCGCAGTGTCGTACGTAGTCGTAGTCATCGATGCGTATATACT
- GAAGCGTACATGCTAGTCTACGTCTCGCAGTTCGTATGCGATCGTATGACG      - AGTACGTAGTATCTGAGCGTCTCTTCTACGGACGATCGATCGTACACGTGA
- ACGCACGATCGATCTATGCATCGATGCTAGTCGTATGCTAGTCTATGCGAT      - TTTATTACGACGATACGTGCGCATTGCGTATGCAGTGTCAGTCAGTCGTAG
```

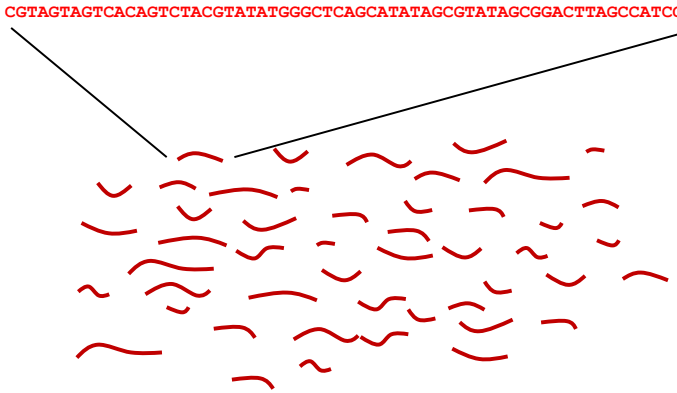

Nucleic Acid Sequencing



Nucleic Acid Sequencing



Nucleic Acid Sequencing



Nucleic Acid Sequencing

```
>CGTAGTAGTCACAGTCTACGTATATGGGCTCAGCATATAGCGTATAGCGGACTTAGCCATCG
>GCGTATAGTCTATATACGACTTATCGGCTCGGTCGCAGAGCAGATATATGCAGTTATATGCTAG
>CCTACGTTATATCGATACTACTAGCTCGTCATGAGCGAGTAGATAGTATGACGAGCGACGATC
...
>CGATATTAGCCTAGCATCATTACGGCGAGACTCTCGGCTCGCTATATAGCGCTATAGCGAT
>CGGCTATAGCGCATATGCTCAGTAGCTATTAGCAGTATTACGATTATAGTCTCGGCGCATTAC
>TTTCGCGGGATAAGTCTTCGTCTTATGCGACGATTATACGGCCGTATATTTGCATTTAGCATTT
>GGCGTATGGCGGATATCGGCGCGTCATAGCGCCGATTAGGCTACGCCGATGCATCG
>CGCGATCGCGCGCGATCGCGTCAGTCGCGCAGTAGCGCGGCATAGTCGTATCGGCGCCG
>TGACAGAAGCTATAAGAGTCAGTAGATCTGAGTATTAGCATTATCGGCGCGATGCGCGATAACG
>GCGTATAGTCTATATACGACTTATCGGCTCGGTCGCAGAGCAGATATATGCAGTTATATGCTA
>CGCGATCGCGCGCGATCGCGTCAGTCGCGCAGTAGCGCGGCATAGTCGTATCGGCGCCGATCGC
>ATAGCAGCACGTGATAGGATATGCTGCTCGTTCGACTATCATATCTGCGTCTGGTCTAGCA
>TGACAGAAGCTATAAGAGTCAGTAGATCTGAGTATTAGCATTATCGGCGCGATGCGCGA
>CCTACGTTATATCGATACTACTAGCTCGTCATGAGCGAGTAGATAGTATGACGAGCGACGATCC
>CGTAGTAGTCACAGTCTACGTATATGGGCTCAGCATATAGCGTATAGCGGACTTAGCCATCG
>TTTCGCGGGATAAGTCTTCGTCTTATGCGACGATTATACGGCCGTATATTTGCATTTAGCATTT
>GGCGTATGGCGGATATCGGCGCGTCATAGCGCCGATTAGGCTACGCCGATGCATCGTCGAGTA
```


Mapping to Reference Genome

Reference Genome

```
CGTAGTAGTCACAGTCTACGTATATGGGCTCAGCATATAGCGTATAGCGGACTTAGCCATCGCGTG
TAGTCTAGTCAGTATAGCGATCAGTGACTATGCAGTCTACGTAGTCGTTATGACGTAGTCGATGTA
GCTAGTATCGTAGTGTACGGCATAAGTCGCGCATGGCTGCGTCGTCATCATATCGGCACGACGTCT
AGAGTAGTCACACTTGTGTGTATATAGCGCGCGGGAGGAGAGCTCTCTGAATAGCTGAGATGCGTA
AGTGCAGGAGAGATCTCTCTGAGAGAGTCTCGGGGATCTCTCTAGAAAGCTCTCGGAAGGATCTCG
AGAGGACTCTC6CGTAGAGAGCTTCAGAGAGACGATATATATGCGATTAGCTACGTATGC
```

Sequencing Read

```
TCATCATATCGGCACGACGTCTAGAGTAGTCACACTTGTGTGTATATAGCGCGCGGGAGGAGA
```


Burrows-Wheeler Transform (BWT)

```
ATCATTAAATCATG$          GTAA$CCTTTTAAAA

ATCATTAAATCATG$   0      $ATCATTAAATCATG   14
TCATTAAATCATG$A   1      AAATCATG$ATCATT    6
CATTAAATCATG$AT   2      AATCATG$ATCATTA    7
ATTAAATCATG$ATC   3      ATCATG$ATCATTAA    8
TTAAATCATG$ATCA   4      ATCATTAAATCATG$    0
TAAATCATG$ATCAT   5      ATG$ATCATTAAATC   11
AAATCATG$ATCATT   6      ATTAAATCATG$ATC    3
AATCATG$ATCATTA   7      CATG$ATCATTAAAT   10
ATCATG$ATCATTAA   8      CATTAAATCATG$AT    2
TCATG$ATCATTAAA   9      G$ATCATTAAATCAT   13
CATG$ATCATTAAAT  10      TAAATCATG$ATCAT    5
ATG$ATCATTAAATC  11      TCATG$ATCATTAAA    9
TG$ATCATTAAATCA  12      TCATTAAATCATG$A    1
G$ATCATTAAATCAT  13      TG$ATCATTAAATCA   12
$ATCATTAAATCATG  14      TTAAATCATG$ATCA    4
```


Efficient Substring Search

```
ATCATTAAATCATG$          GTAA$CCTTTTAAAA

TCA

                         $ATCATTAAATCATG   14
                         AAATCATG$ATCATT    6
                         AATCATG$ATCATTA    7
                         ATCATG$ATCATTAA    8
                         ATCATTAAATCATG$    0
                         ATG$ATCATTAAATC   11
                         ATTAAATCATG$ATC    3
                         CATG$ATCATTAAAT   10
                         CATTAAATCATG$AT    2
                         G$ATCATTAAATCAT   13
                         TAAATCATG$ATCAT    5
                         TCATG$ATCATTAAA    9
                         TCATTAAATCATG$A    1
                         TG$ATCATTAAATCA   12
                         TTAAATCATG$ATCA    4
```

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with A

[1,7)

Thus, the substring A

is in the reference sequence at indices

{6,7,8,0,11,3}

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with A

[1,7)

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with A

[1,7)

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with **A** [1,7)

Range of rows starting with **CA** [7,9)

Thus, the substring **CA** is in the reference sequence at indices {10,2}

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with **A** [1,7)

Range of rows starting with **CA** [7,9)

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with **A** [1,7)

Range of rows starting with **CA** [7,9)

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with **A** [1,7)

Range of rows starting with **CA** [7,9)

Range of rows starting with **TCA** [11,13)

Thus, the substring **TCA** is in the reference sequence at indices {9,1}

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with A [1,7)

Range of rows starting with CA [7,9)

Range of rows starting with TCA [11,13)

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with A [1,7)

Range of rows starting with CA [7,9)

Range of rows starting with TCA [11,13)

Each step of the search *must be fast*!
O(1) time.

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Precompute Helper Information

GTAA$CCTTTTAAAA

Returns the number of nucleotide characters in the BWT that are lexographically less than c.

```
public int getNumberCharactersLessThan(char c);
```

```
getNumberCharactersLessThan('$');  returns 0
getNumberCharactersLessThan('C');  returns 7
getNumberCharactersLessThan('T');  returns 10
```

Returns the number of occurrences of nucleotide character c in the BWT up to but not including index i.

```
public int getNumberOccurrencesPriorToIndex(char c, int i);
```

```
getNumberOccurrencesPriorToIndex('A',5);   returns 2
getNumberOccurrencesPriorToIndex('A',13);  returns 4
getNumberOccurrencesPriorToIndex('T',8);   returns 2
```


Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

How can we compute this?

Range of rows starting with A

Range of rows starting with CA

Range of rows starting with TCA

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Efficient Substring Search

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

TCA

Range of rows starting with A

Range of rows starting with CA

Range of rows starting with TCA

How can we compute these?

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Another Example

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

ATT

Range of rows starting with T

Range of rows starting with TT

Range of rows starting with ATT

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

A Substring Not in the Reference

ATCATTAAATCATG$ GTAA$CCTTTTAAAA

GTC

Range of rows starting with C

Range of rows starting with TC

Range of rows starting with GTC

$ATCATTAAATCATG	14
AAATCATG$ATCATT	6
AATCATG$ATCATTA	7
ATCATG$ATCATTAA	8
ATCATTAAATCATG$	0
ATG$ATCATTAAATC	11
ATTAAATCATG$ATC	3
CATG$ATCATTAAAT	10
CATTAAATCATG$AT	2
G$ATCATTAAATCAT	13
TAAATCATG$ATCAT	5
TCATG$ATCATTAAA	9
TCATTAAATCATG$A	1
TG$ATCATTAAATCA	12
TTAAATCATG$ATCA	4

Algorithm: Mapping read to genome

TCA

Read is processed one character (NT) at a time, from right to left.

Base Case: Last character (NT) in read, ch

The range of rows where ch appears in the BWT is [start, end) where

```
start = getNumberCharactersLessThan(ch)

if (ch is not 'T')
      end = getNumberChactersLessThan(NT after
                                      ch alphabetically)
if (ch is 'T')
      end = length of BWT
```

Algorithm: Mapping read to genome

Read is processed one character (NT) at a time, from right to left.

Iterative Case: Character (NT) at index *i* in read, ch

The range of rows where the substring from *i* onward in the read appears in the BWT is [start, end) where

```
start = getNumberCharactersLessThan(ch) +
     getNumberOccurrencesPriorToIndex(ch,
          start for character at index i+1)

end = getNumberCharactersLessThan(ch) +
     getNumberOccurrencesPriorToIndex(ch,
          end for character at index i+1)
```


Efficient Substring Search with Errors

GCTAGAGTATATCGGCGCTATGCGTTGTCTCTCTAGAGAGACGATTCTAGTCTGCCTGTC

Suppose we tried to map a read of length 60 nts to a reference genome and found that the read did not map. Perhaps the read contains one or two errors from the sequencing process.

- Break the read up into three pieces and map each of the three pieces to the reference.
- If none of the three pieces map, then there are at least three errors in the read.
- If one or more of the pieces map, then we use that mapping to perform a *fast* alignment.