2013 - BMMB 597D: Analyzing Next Generation Sequencing Data

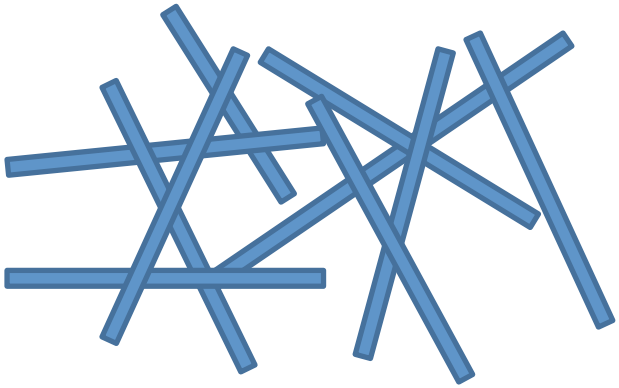
Week 7, Lecture 14

István Albert

Biochemistry and Molecular Biology
and Bioinformatics Consulting Center

Penn State

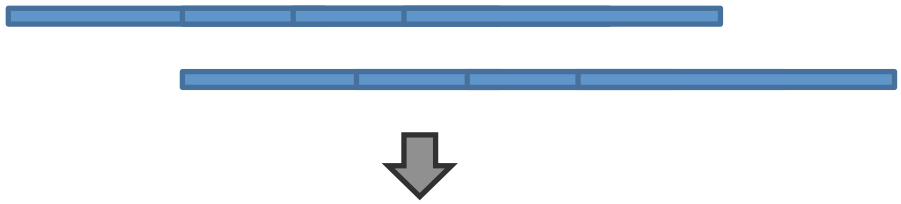
Processing sequencing reads



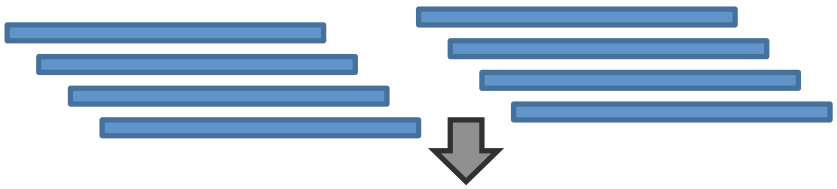
Sequenced DNA fragments
(DNA library)

Unknown Genome
De novo assembly

Know Genome
Read mapping (alignments)



Known genome (reference)



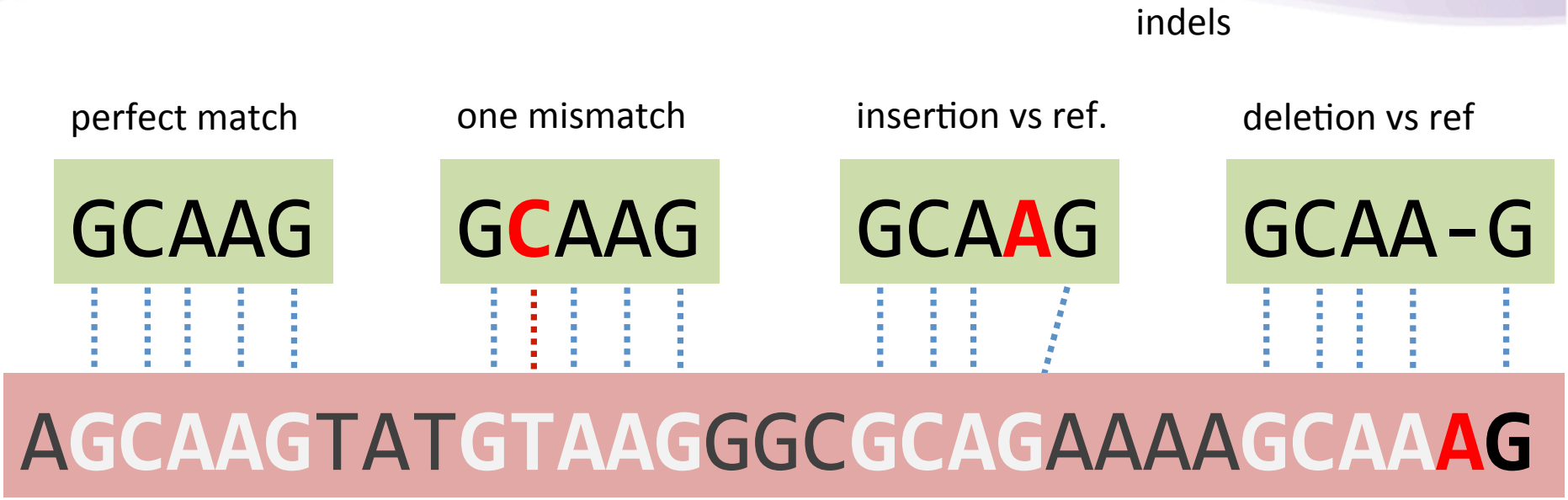
gene discovery (annotation)

chip-Seq, RNA-Seq, SNP calling etc.

Sequence alignments

- Arranging two or more sequences such as to maximize the length of the common regions between the two
- It is a very well developed field – the roots of the bioinformatics started with various alignment software
- We will only cover pair-wise alignments searching a database with a query
- High throughput sequencing poses special constraints: a **very large number** of very short reads - traditional methods were not feasible → **semi-global alignment**

Alignment concepts



NOTE: mismatches or indels can be longer than 1 base!

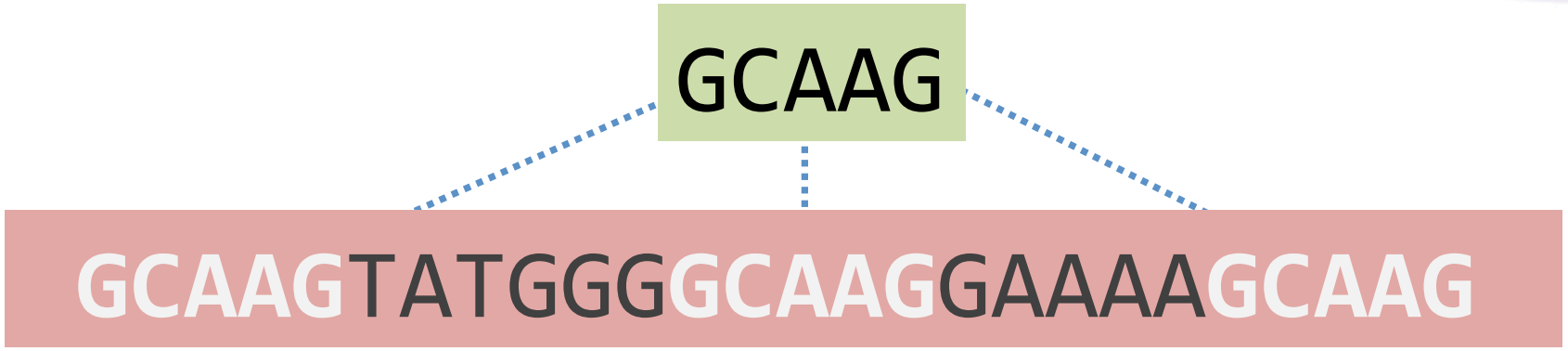
It gets complicated very quickly

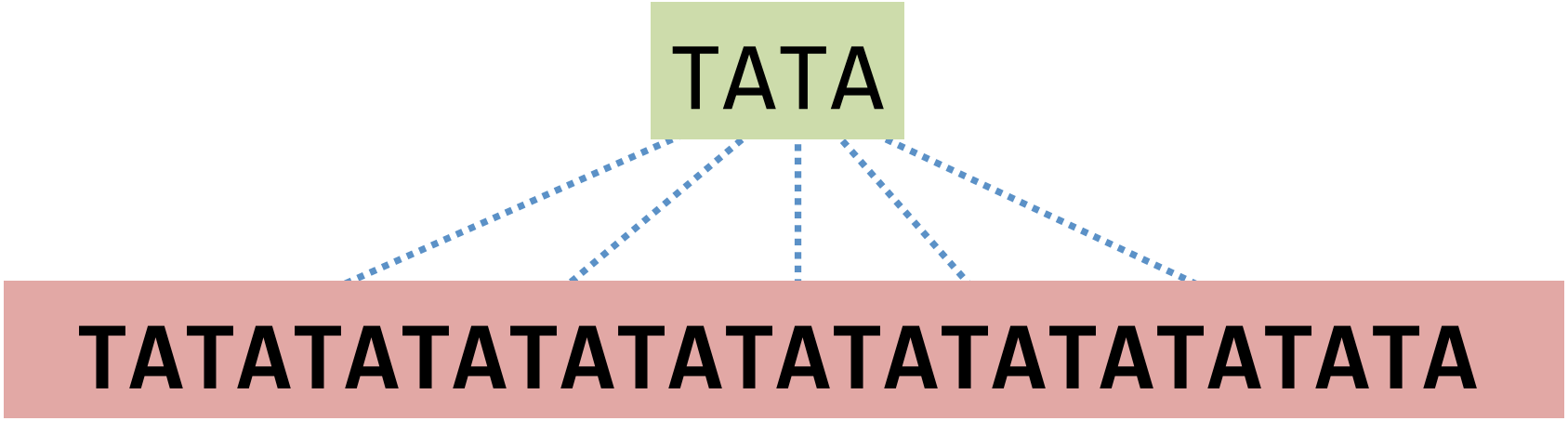
Alignment scoring depends on mismatch scoring (different across bases!), gap open, gap extension penalties

Use the terminology correctly

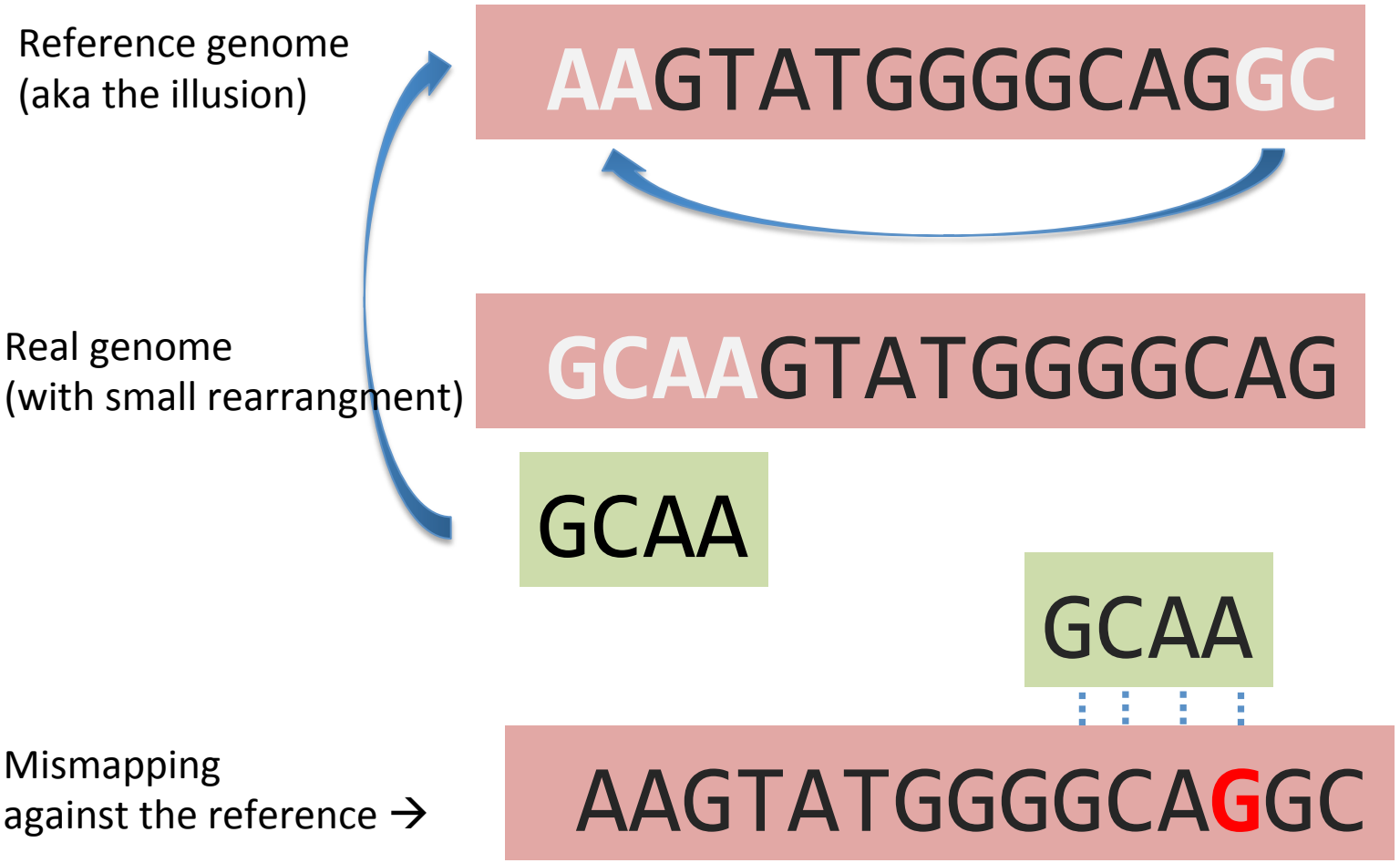
- A mismatch is not a SNP (single nucleotide polymorphisms)
- But SNPs will show up as mismatches
- indels are not DIPs (insertion/deletion polymorphisms)
- But DIPs will show up as indels

Challenges: repetitive and low complexity regions





Challenge: genomic reorganization can easily lead to mis-mapping



Short Read Mappers (aligners)

- Use **heuristics** to quickly identify locations (hits) where the reads match
- Heuristics → not all hits will be found!
- Tradeoff: **resource usage** vs **speed** vs **accuracy** vs **usability**
- Each domain of application may have more appropriate tools

A few popular aligners

- BWA, bowtie, SOAP2, Shrimp, BFAST and many others

There is no **single best tool**, the issues to consider:

1. **documentation** → can we figure out how it works
2. **input features** → what type of input can it handle
3. **reporting features** → will it produce the type of output that we can use
4. **performance** → is it feasible to run on my resources

Aligner features

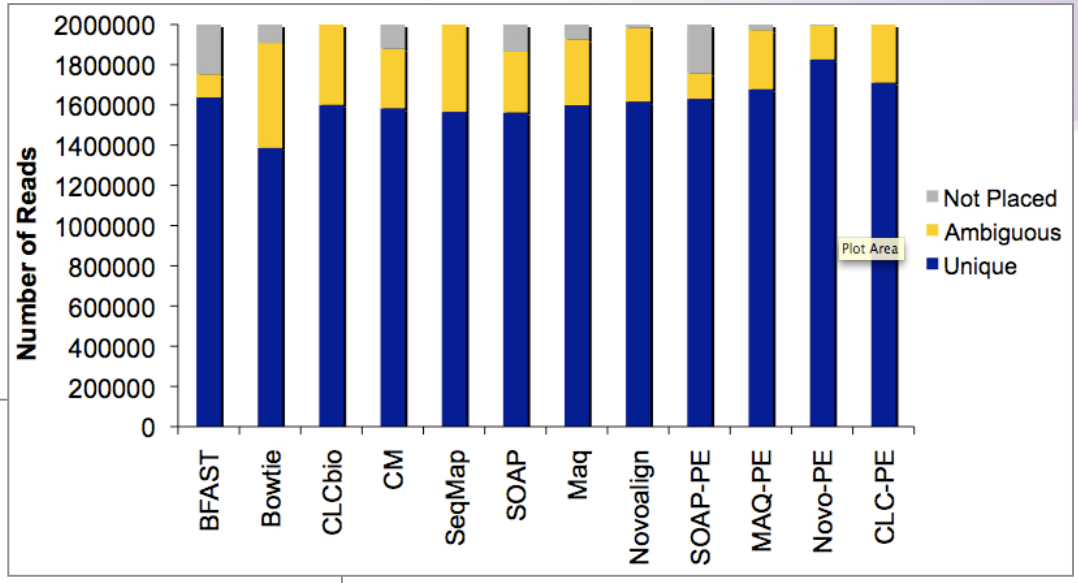
- Some cannot handle **indels** (insertion/deletions) → these tools are typically extremely fast will run on a laptop for even large genomes (**bowtie 1**) → application Chip-Seq

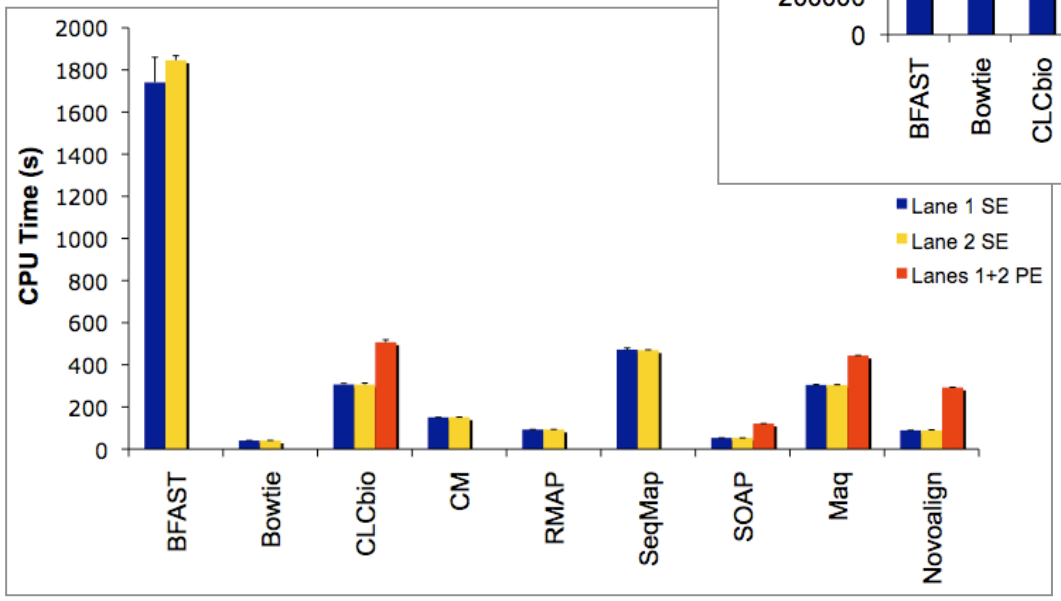
- May or may not use the quality score during the alignment (usually not, that would alter the alignment score)

- Differences in the they report alignments:

  - unique alignments only (note that this is an ill defined concept!)
  - best alignments above a cutoff
  - all possible alignments (can slow down the process greatly)

CPU time to align 1 million reads





from: http://www.massgenomics.org/short-read-aligners

Tool comparisons

The vast majority of tools comparisons lead to "red-herring" type fallacies

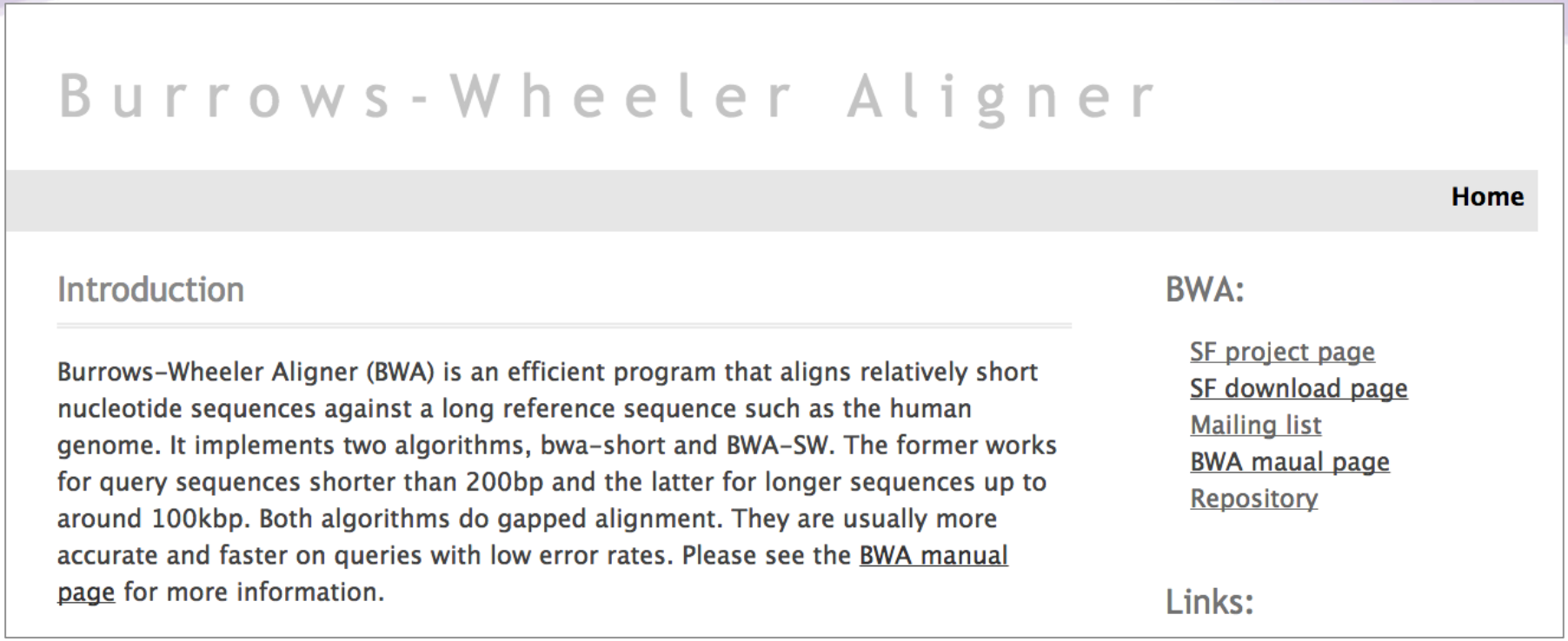
1. Tool A is faster than Tool B and we tacitly assume that the results are the same

2. Tool A is more 'accurate' than Tool B and we assume the resource utilization to be the same

Robust results should be tool independent!

- We need tools that are correct and finish running within our resources
- Results should be hold across methods
- There are exception to this – but those need to be explicitly documented – and you need to know why

BWA (Burrows-Wheeler Aligner) by Heng Li

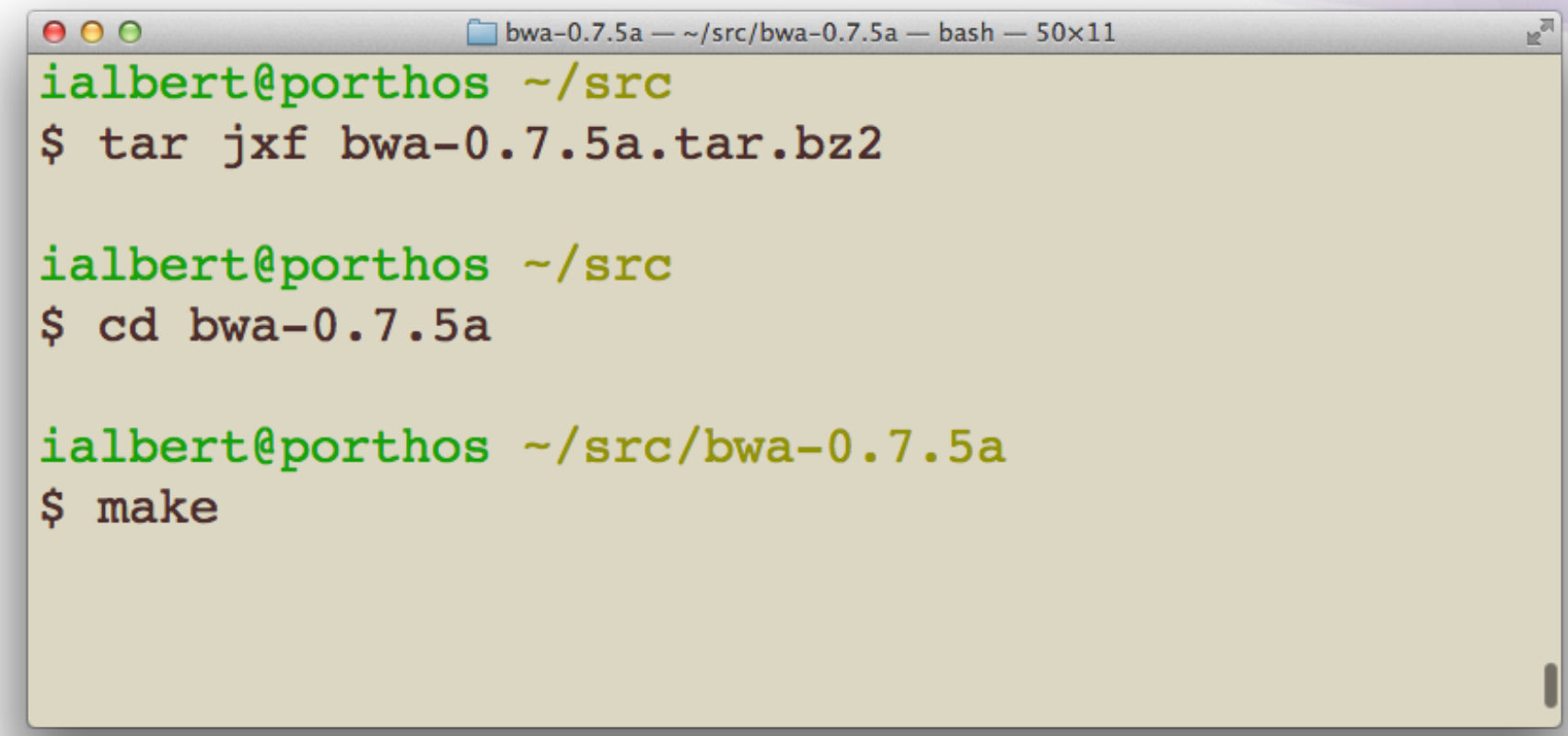


http://bio-bwa.sourceforge.net/

Download, unpack, compile with make and link to bin

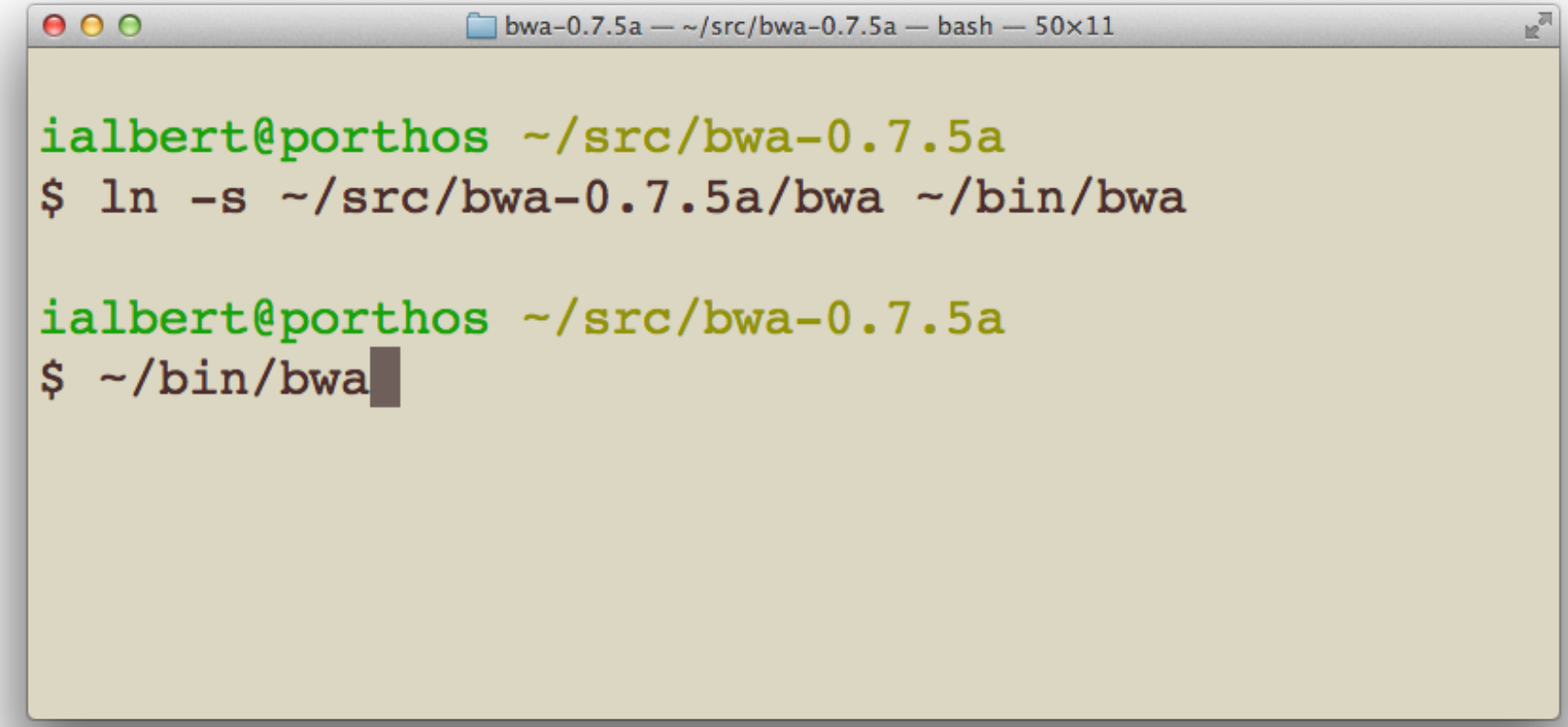
Read also the **bwa-mem** controversy – the **bwa-mem** paper rejection

Download – unpack and make



Uses **bz2** compression hence the **j** flag (the **z** command would expand a gzip file)

Link it the tool into your bin folder



Aligning reads with BWA (old-style)

For BWA is three step process (this is different for each aligner):

1. **Index the genome** – this only needs to be done once for each genome → **bwa index …**
2. **Create the alignment → bwa aln …**
3. **Format the alignment → bwa samse …**

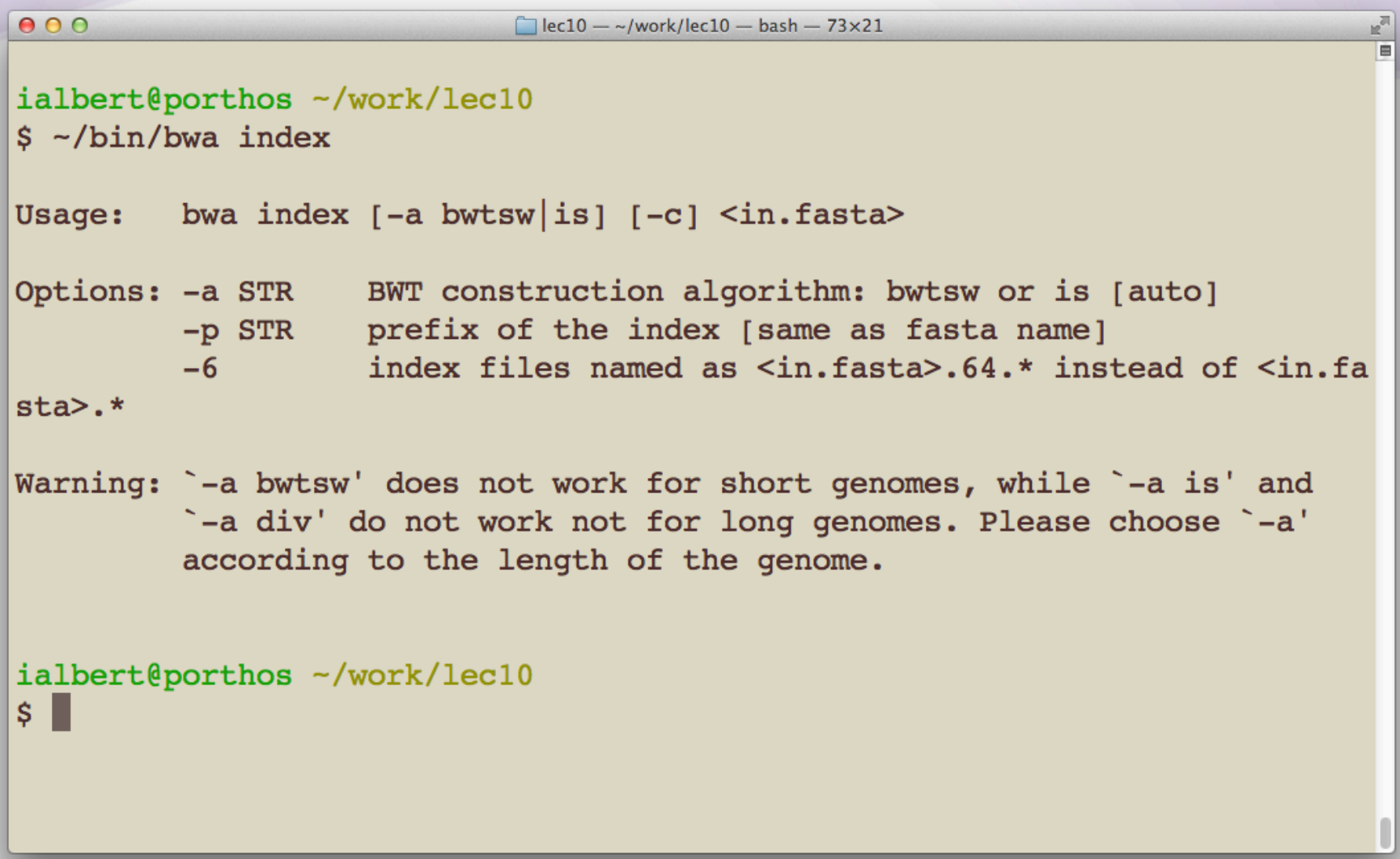
Aligning reads with BWA (new-style)

For BWA is three step process (this is different for each aligner):

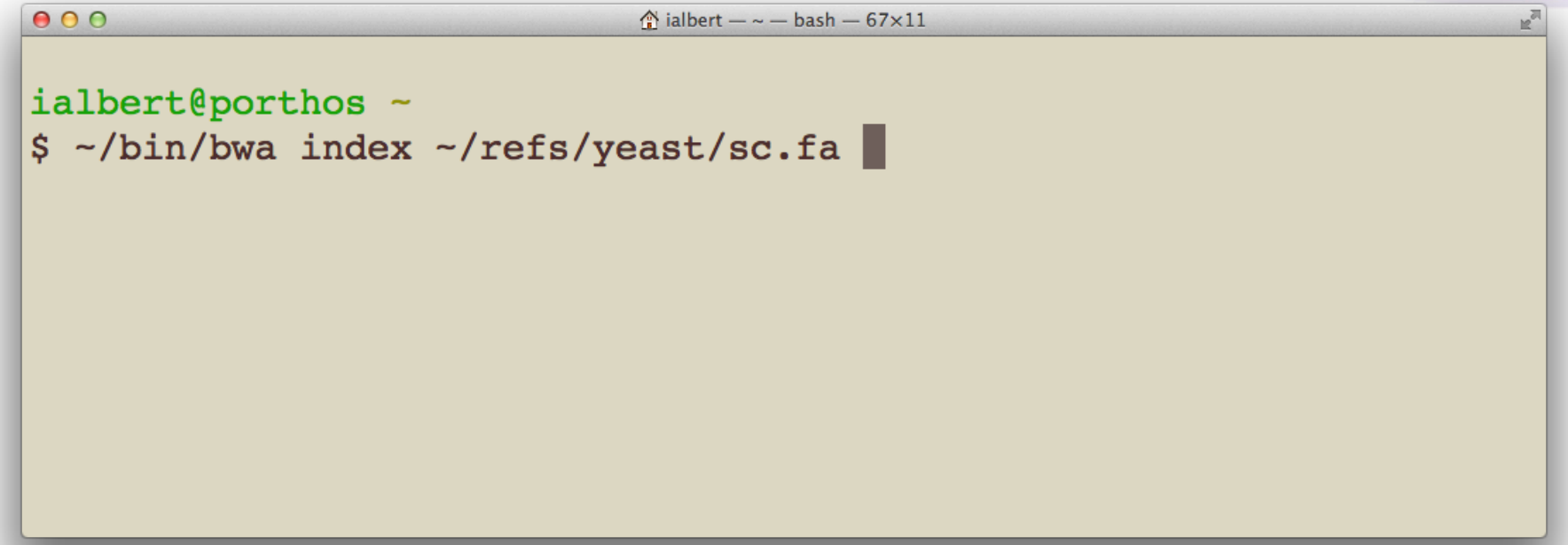
1. **Index the genome** – this only needs to be done once for each genome → **bwa index …**

2. **Create the alignment → bwa mem …**

There is built-in help



Index the genome

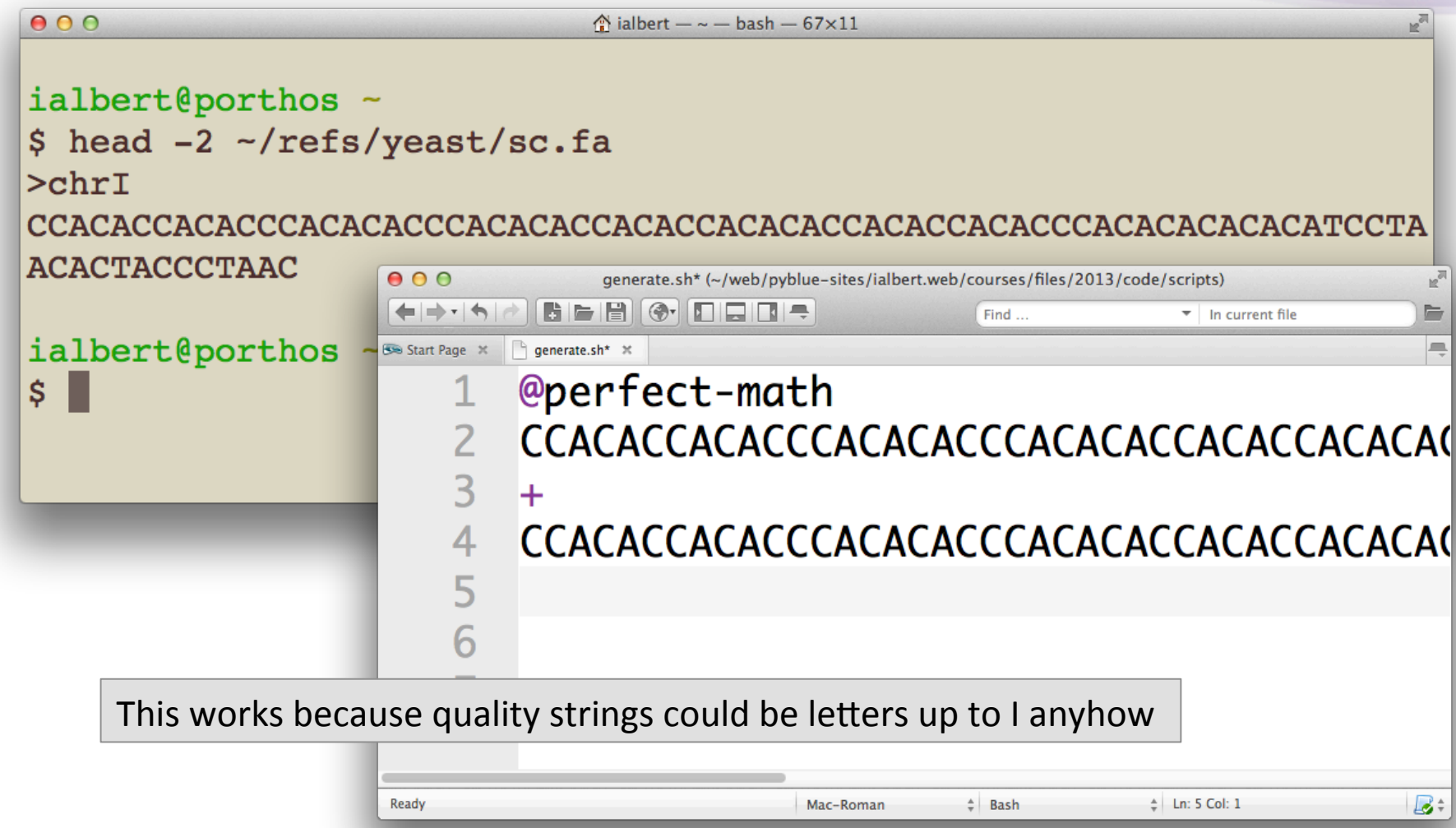


For long genomes (over 2GB) you need to index with the **–a bwtsw** option.

Periodically re-read the manual to remind yourself of what the tool can do.

Create a FASTQ query sequence

- Select some bases from the beginning of chromosome 1

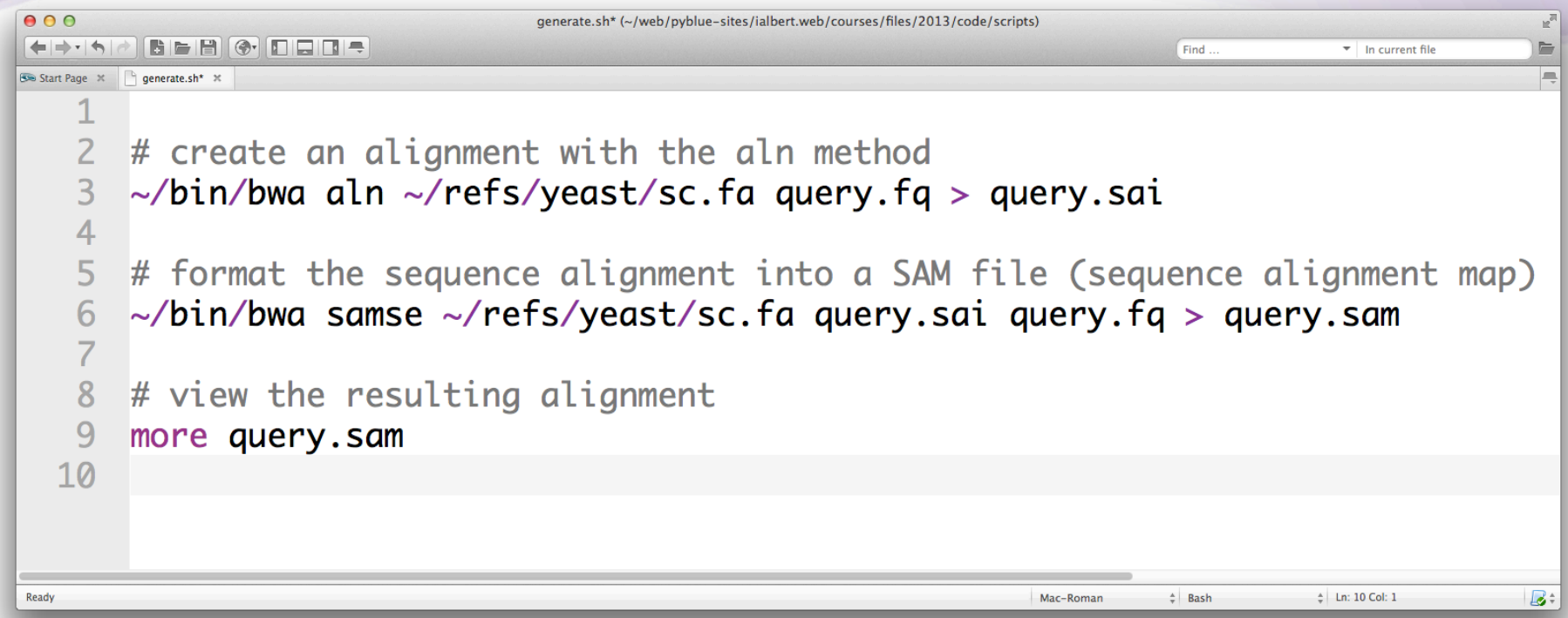


This works because quality strings could be letters up to I anyhow

BWA concepts

- the **index** or **prefix** (***.fasta**): the full path to the fasta reference sequence that has been indexed
- the **query (*.fq)**: the file containing the sequences that we want to look up in the reference
- the **alignment** (***.sai**): a binary intermediate file that contains the alignment information for each read
- the **sequence alignment map** (***.sam**): the result file that contains the alignment cross referenced with the query
- **bwa** can operate with different alignment methods! These may generate very different outputs!

Run and format with BWA **aln**

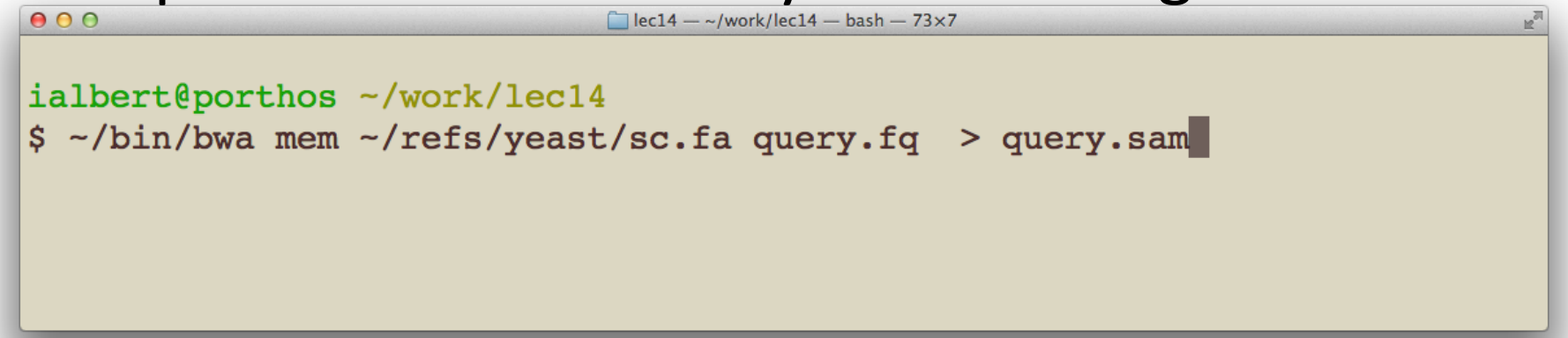


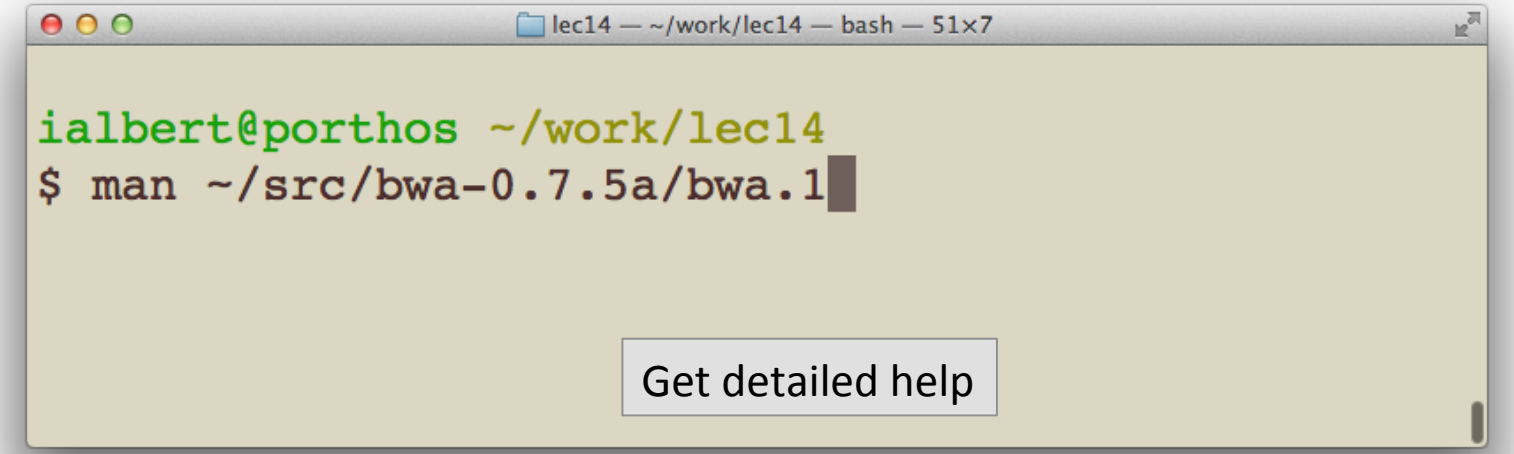
A multistep process that is best used via a shell script.

The file **query.sam** is the final output of the alignment

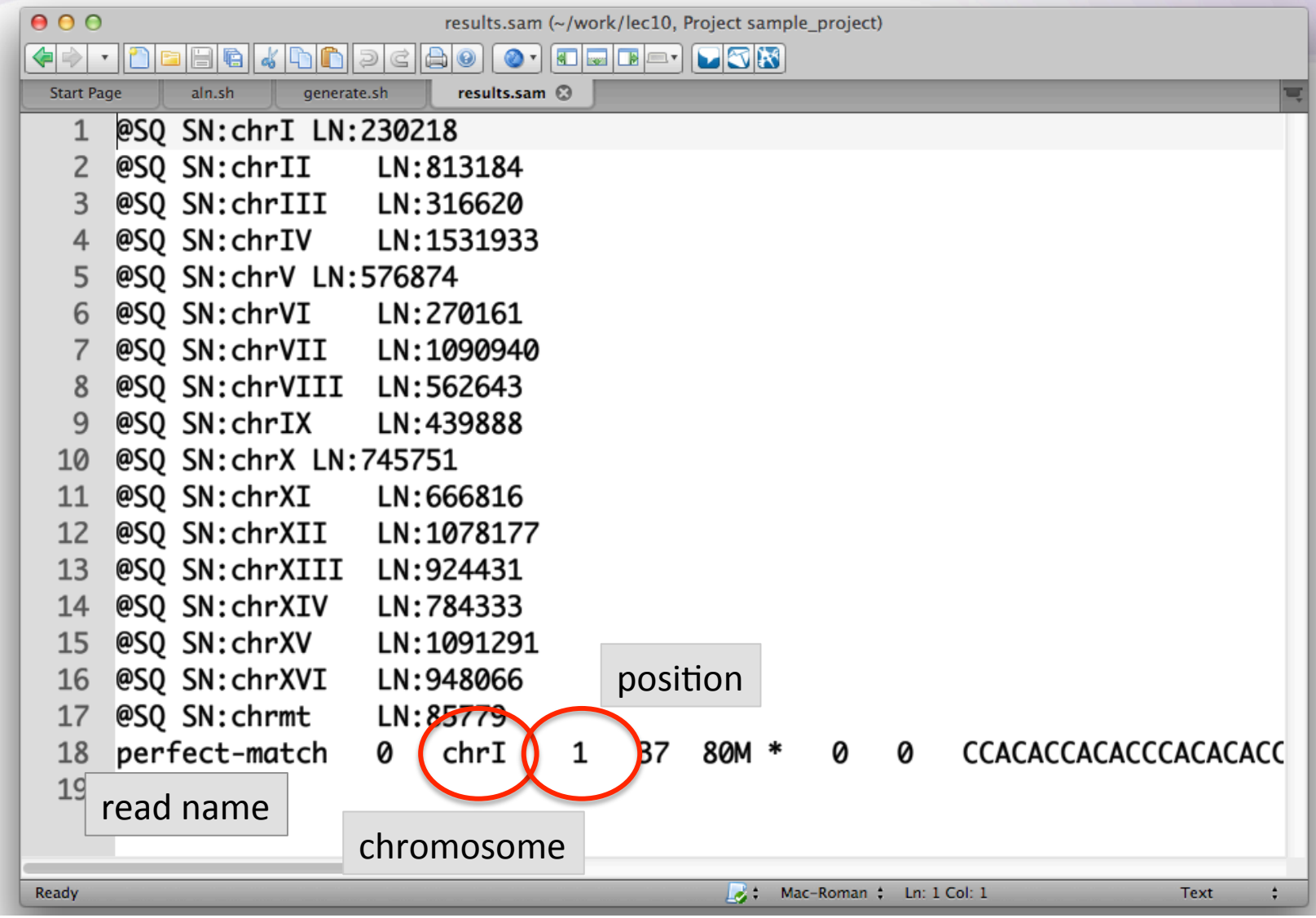
Use the bwa mem method

- This is a different alignment strategy that operates via a radically different algorithm





The result of the alignments



Homework 14

Create a FASTQ file using the first 60 or so bases from chromosome 1 of the yeast genome.

Align your query file with both BWA and BLAST and look at the output of both processes.

Both are sequence aligners, what is the main difference that you observe?

(remember to turn your fastq file into fasta when blasting)