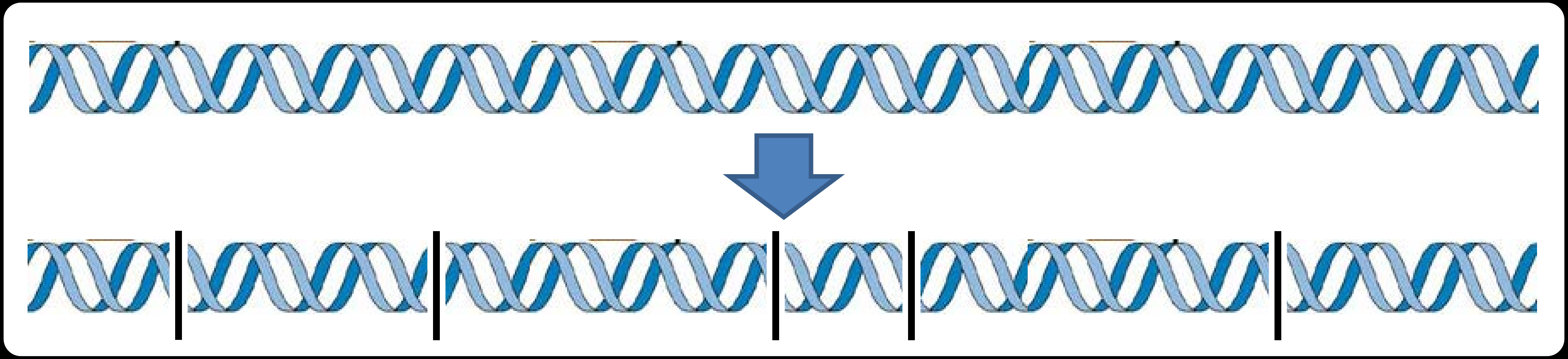
Genome Assembly

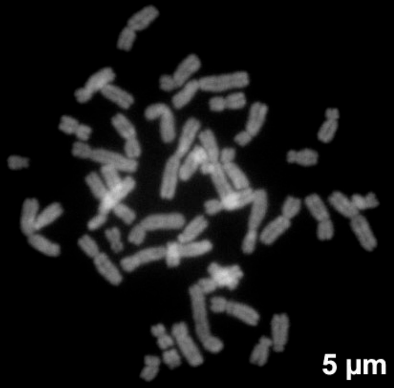
Genome Assembly: an open research field

- All modern sequencing technologies (described later) break up DNA into small segments of nucleotides



- While there are a number of reasons for breaking up the DNA, the biggest reason is that there is no sequential strategy that is sustainable or fast enough for long sequences
- Thus breaking up the DNA and sequencing the chunks in parallel is the only efficient approach

Terminology



- Bases:
  - A nucleotide is often called a base.
  - 1,000 nucleotides = kilobase (Kb)
  - 1,000,000 nucleotides = megabase (Mb)
  - 1,000,000,000 nucleotides = gigabase (Gb)
  - Human genome = just over 3 gigabases
- Reads:
  - Very short DNA sequences that you get out of the sequencing machine
  - Very short reads: 25-75 bases
  - Long reads: 400-500 bases
- Contigs:
  - Maximally assembled segments of reads
- Coverage:
  - The average number of times the same base appears in a read

A note on coverage

- The human genome was sequenced to 12x coverage
- Just because the average nucleotide appears 12 times does not mean that every nucleotide appears 12 times

- Many nucleotides, especially in repetitive regions of the genome are difficult to assemble accurately, so they are not well covered.

The Main Problem of Sequence Assembly

- Input:
  - – A very large number of reads
  - – Typically this is provided in a fasta file format
- Output:
  - – A smaller number of contigs
  - – Typically this is also provided as a fasta file
- Problem:
  - – Assemble the reads into the shortest possible sequence of nucleotides

A very basic picture of sequence assembly

ATATGGGCCACCAC   CACCACTGACGAC

ATATGGGCCACCACTGACGAC

🚫 ATATGGGCCACCACCACTGACGAC

- Reads are assembled by identifying reads with matching prefixes and suffixes
- The shortest possible assembly is always used because of Occam’s razor:
  - The simplest explanation is likely to be correct
- Having many overlapping reads can help fix ambiguities from repetition

Overlapping reads help, to an extent

```
    GGGCCACCACTGAC
        CACCACTGACGAC
ATATGGGCCACCAC
ATATGGGCCACCACTGACGAC
```

- Overlapping reads from lots of coverage can eliminate ambiguities in repetitive regions: as long as the repetitive regions are short
- Long repetitive regions that are as long or longer than the reads themselves cannot be resolved
- Repetitive regions thus prevent the formation of longer and longer contigs

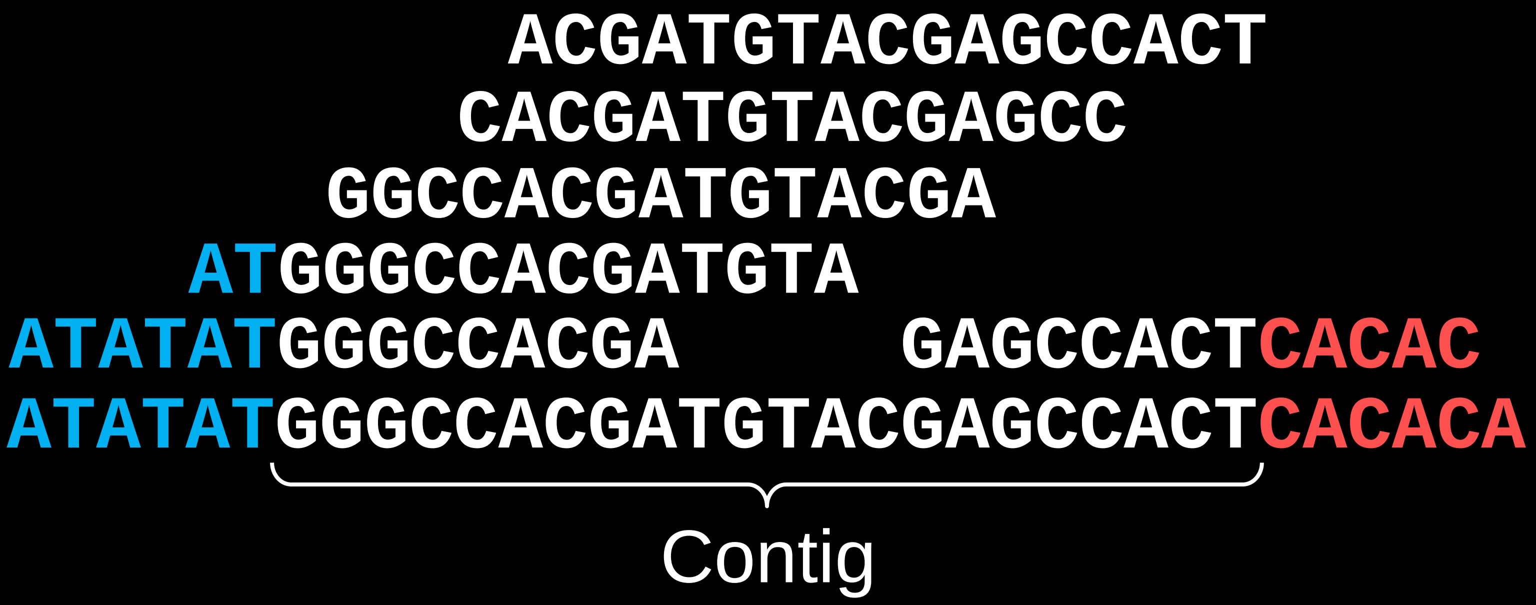
An example of serious repetition issues

```
          CACACACACACACACA
     GGCCACACACACACA
                  CACACACACACATGA
ATATGGGCCACCAC          CACCACTGACGAC

   ATATGGGCCACA.. ? ..CACATGACGAC
```

- When no read can touch both non-repetitive ends of a long repetitive region, then it is impossible to know how long the repetitive region is
- This is the border of a contig

What does a contig look like?

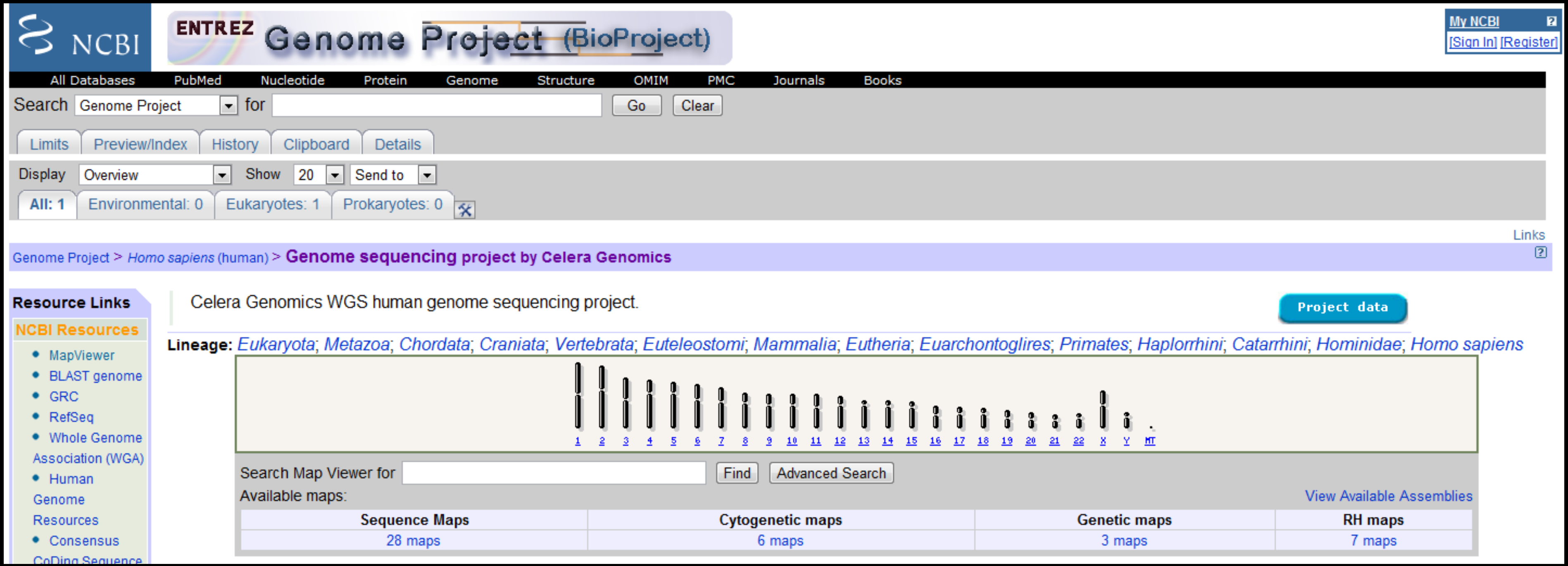


```
           ACGATGTACGAGCCACT
          CACGATGTACGAGCC
       GGCCACGATGTACGA
    ATGGGCCACGATGTA
ATATATGGGCCACGA     GAGCCACTCACAC
ATATATGGGCCACGATGTACGAGCCACTCACACA
```

Contig

- Contigs are regions bordered by repetitive regions, where assembly properly indicates and verifies the DNA sequence
- The purpose of sequence assembly to identify the longest possible contigs

www.ncbi.nlm.nih.gov/genomeprj/1431



Search Genome Project for

Display Overview Show 20 Send to

Genome Project > *Homo sapiens* (human) > **Genome sequencing project by Celera Genomics**

Celera Genomics WGS human genome sequencing project.

Lineage: *Eukaryota*; *Metazoa*; *Chordata*; *Craniata*; *Vertebrata*; *Euteleostomi*; *Mammalia*; *Eutheria*; *Euarchontoglires*; *Primates*; *Haplorrhini*; *Catarrhini*; *Hominidae*; *Homo sapiens*

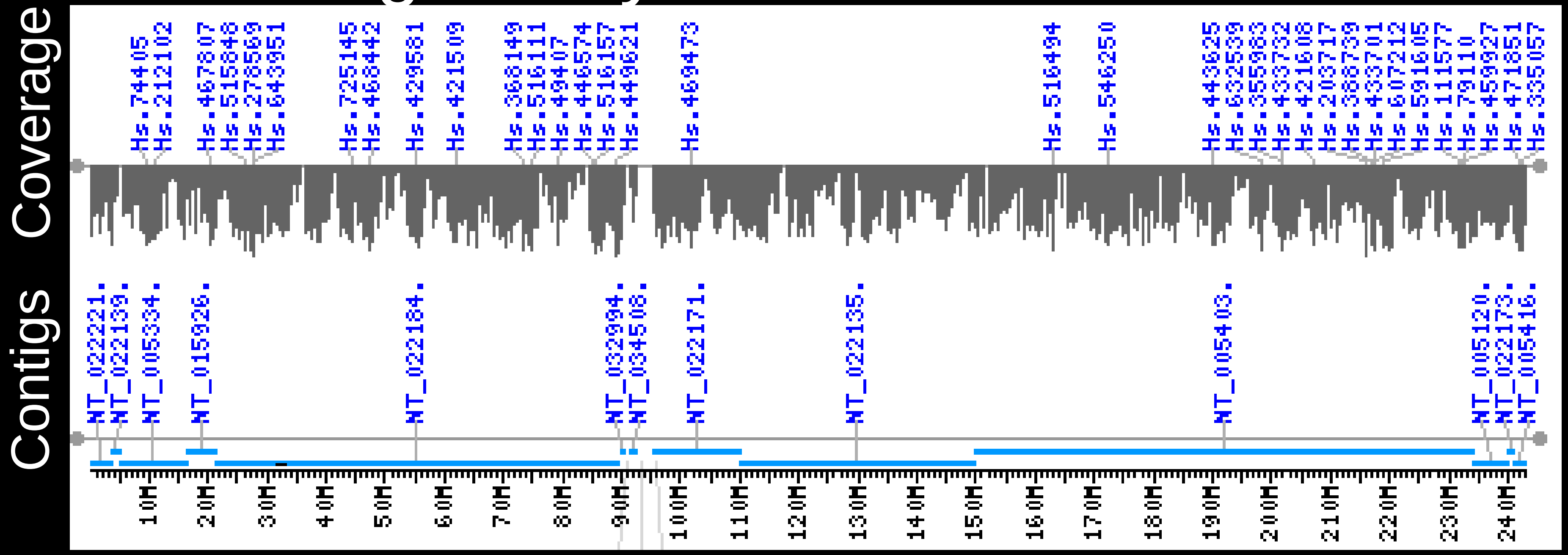
Search Map Viewer for

Available maps:

Sequence Maps	Cytogenetic maps	Genetic maps	RH maps
28 maps	6 maps	3 maps	7 maps

- The National Center for Biotechnology Information (NCBI) stores most public genomics data, such as the human genome

12x coverage fairly successful on humans



www.ncbi.nlm.nih.gov/mapview/maps.cgi?taxid=9606&chr=2

- Chromosome 2 is nearly 250 Mbases and sequenced within 13 contigs
- You can see that where coverage breaks, contigs also break.
- Sequencing technologies have no control over where the reads occur

Genomics is a constantly changing field

- This course will give you a taste of the research questions in major fields of genomics
- There are some practical limitations
  - Many aspects of genomics require specialized hardware – sequencers, supercomputers
- Genome Assembly of complex organisms requires computers with 30+ GB of RAM
- We can do scaled back versions of these things in the project
  - Assembling virus genomes

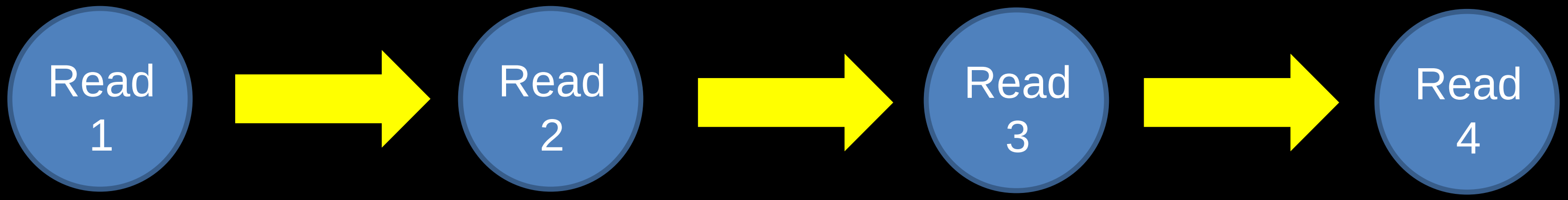
Simplistic Path assembly

- Pick one read A
- Look through all the other reads
  - Find a read B with a 15 base prefix that is identical to the 15 base suffix of A.
  - i.e. suffix(A, 15) == prefix(B,15)
- Look through all the other reads
  - Find a read C with a 15 base prefix that is identical to the 15 base suffix of B.
  - i.e. suffix(B,15) == prefix(C,15)
- Continue..

Why is simplistic path assembly bad?

- Because for each read, you have to look at every other read.
- 15000 bases covered 100 times by 30 base reads is 60,000 reads.
  - For each read, you must check every other read:
  - This is 3.6 billion checks.
  - If each check is 1 millisecond, we’re talking 1000 hours
  - 15k bases is a <u>tiny</u> genome!
- This is really slow. We can do way better.

Eulerian Path Assembly

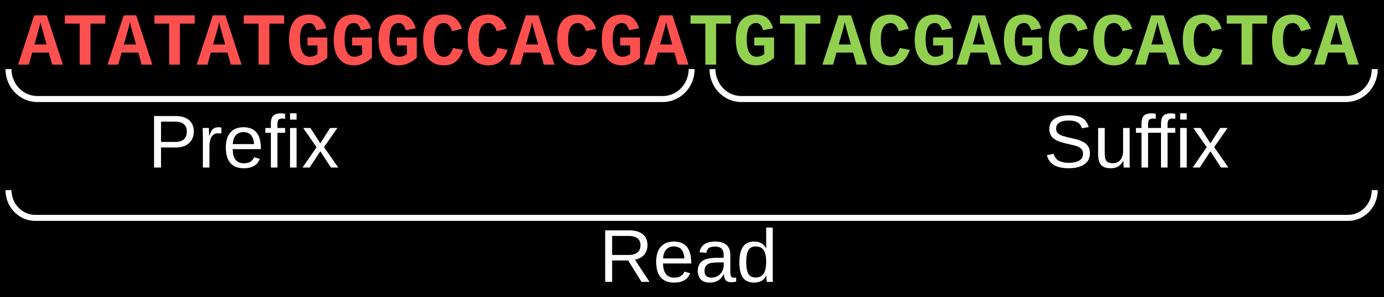


- The problem with simple assembly is that we are forced to start on one end and go through all the reads each time we want to extend the contig.

- What we really need is to go through all the reads once, and have it build the contig as we go.

- Eulerian Path Assembly can do this.
- First we need to learn a little graph theory.

Terminology



ATATATGGGCCACGATGTACGAGCCACTCA

Prefix

Suffix

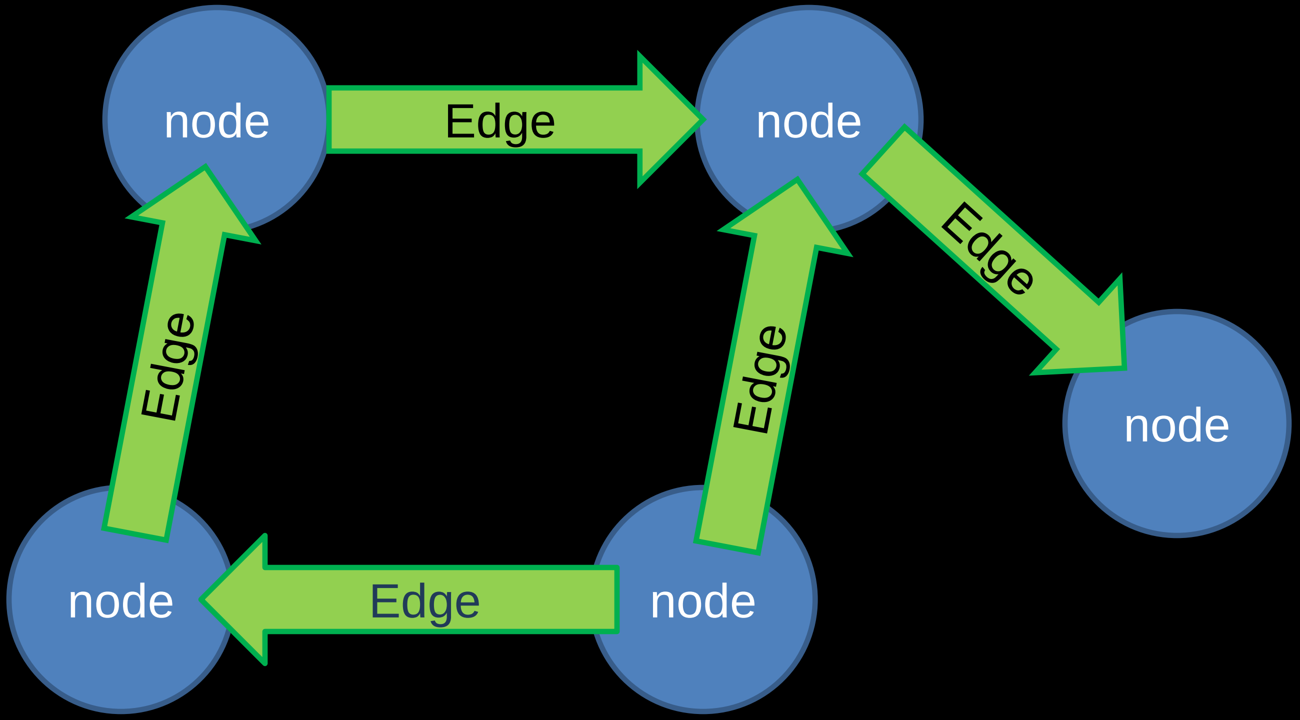
Read

```
…CACGATGTACGAGCCACTCA
           GAGCCACTCAATCTATTTGC…
```

Overlapping reads

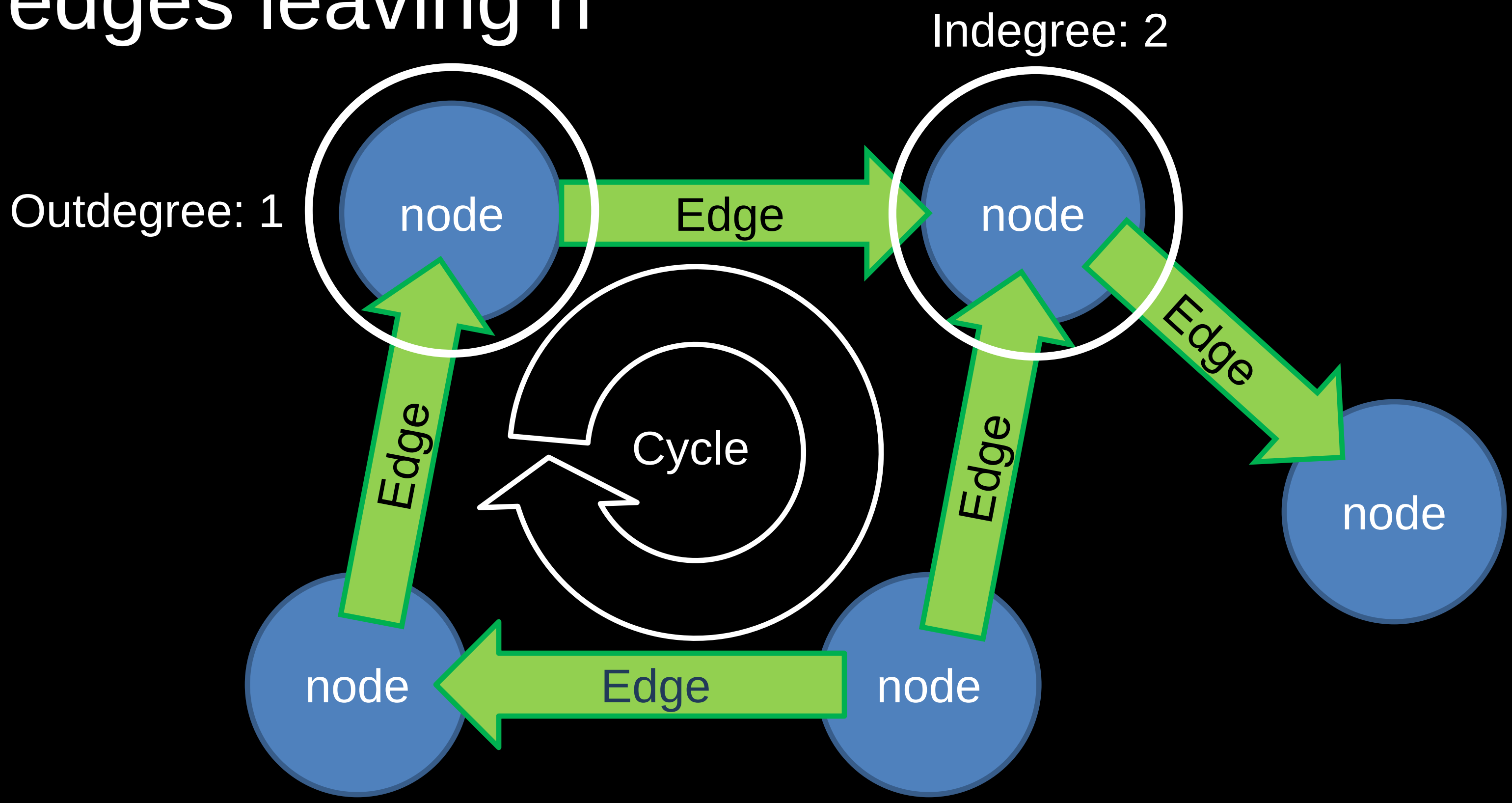
What is a graph?

- Graphs look like this:
  - Nodes represent things
  - Edges directionally connect two nodes
- Edges and nodes can stand for many things



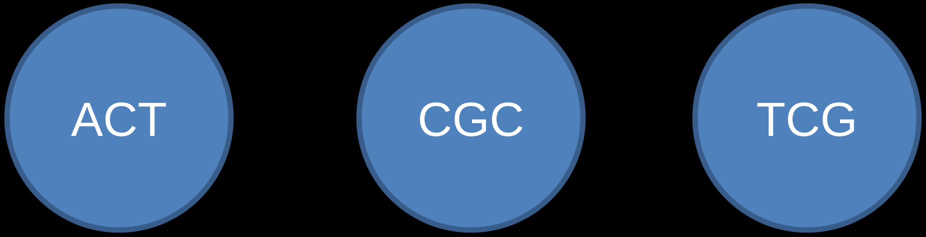
Some things to know about graphs

- The *indegree* of a node n is the number of edges going into n
- The outdegree of a node n is the number of edges leaving n

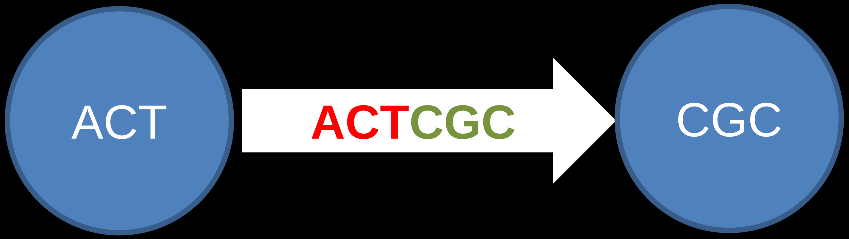


Using Graphs for Sequence Assembly

- Each node will stand for either a prefix or a suffix.
  - Prefixes and suffixes are the same length, so each node will be used to represent both.



- Each edge will stand for a read, which connects a prefix to a suffix



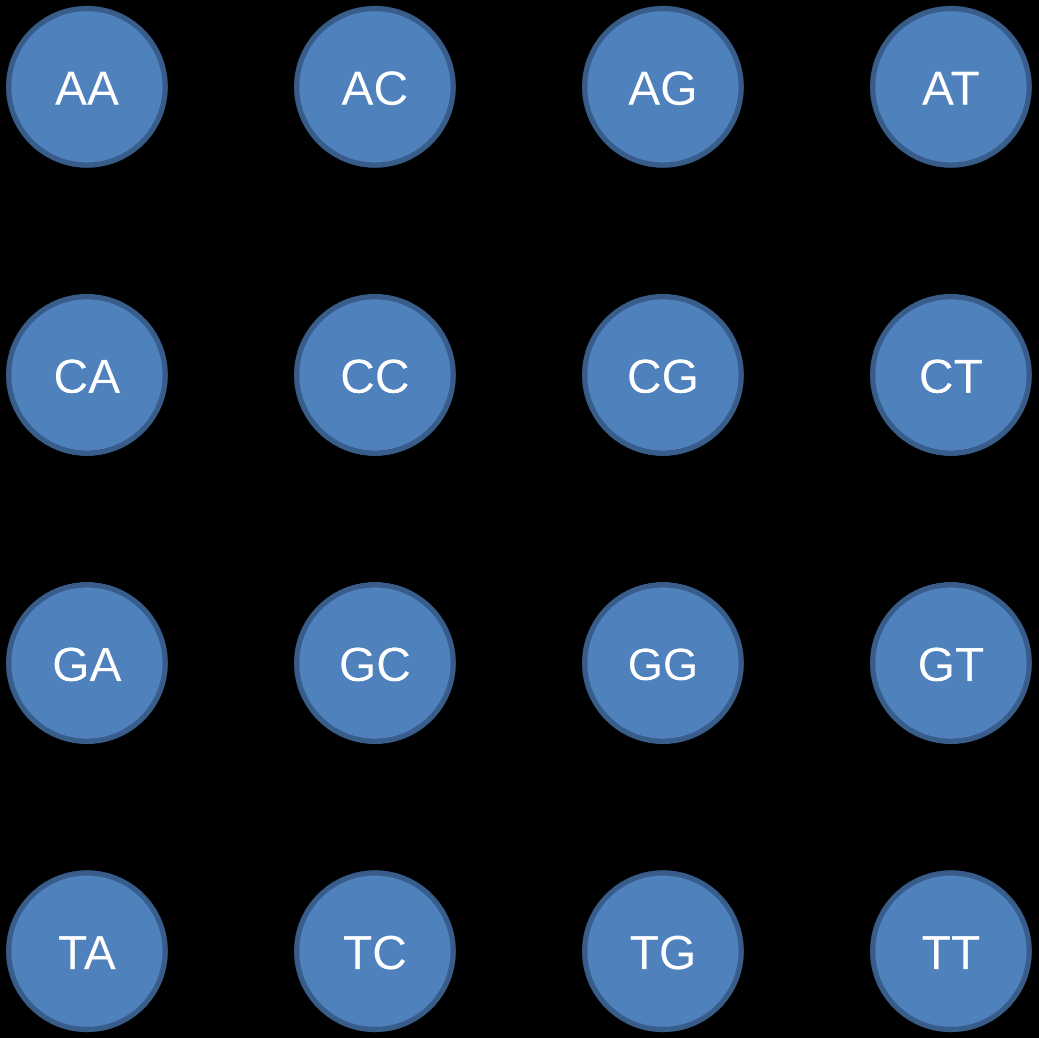
A really simple example

- For this example:
  - Reads are 3 bases
  - Prefixes are 2 bases, suffixes are 2 bases
  - (they overlap, and that's okay)
- Contig: {ATGGCGTGCA}
- Reads:

  {ATG, TGG, TGC, GTG, GGC, GCA, GCG, CGT}
- Prefixes and suffixes:

  {AA, AC, AG, AT, CA, CC, CG, CT, GA, GC, GG, GT, TA, TC, TG, TT}
- This is all 4x4 combinations of 2-letter prefixes and suffixes.

Step1: Prefixes and Suffixes are nodes



AA AC AG AT

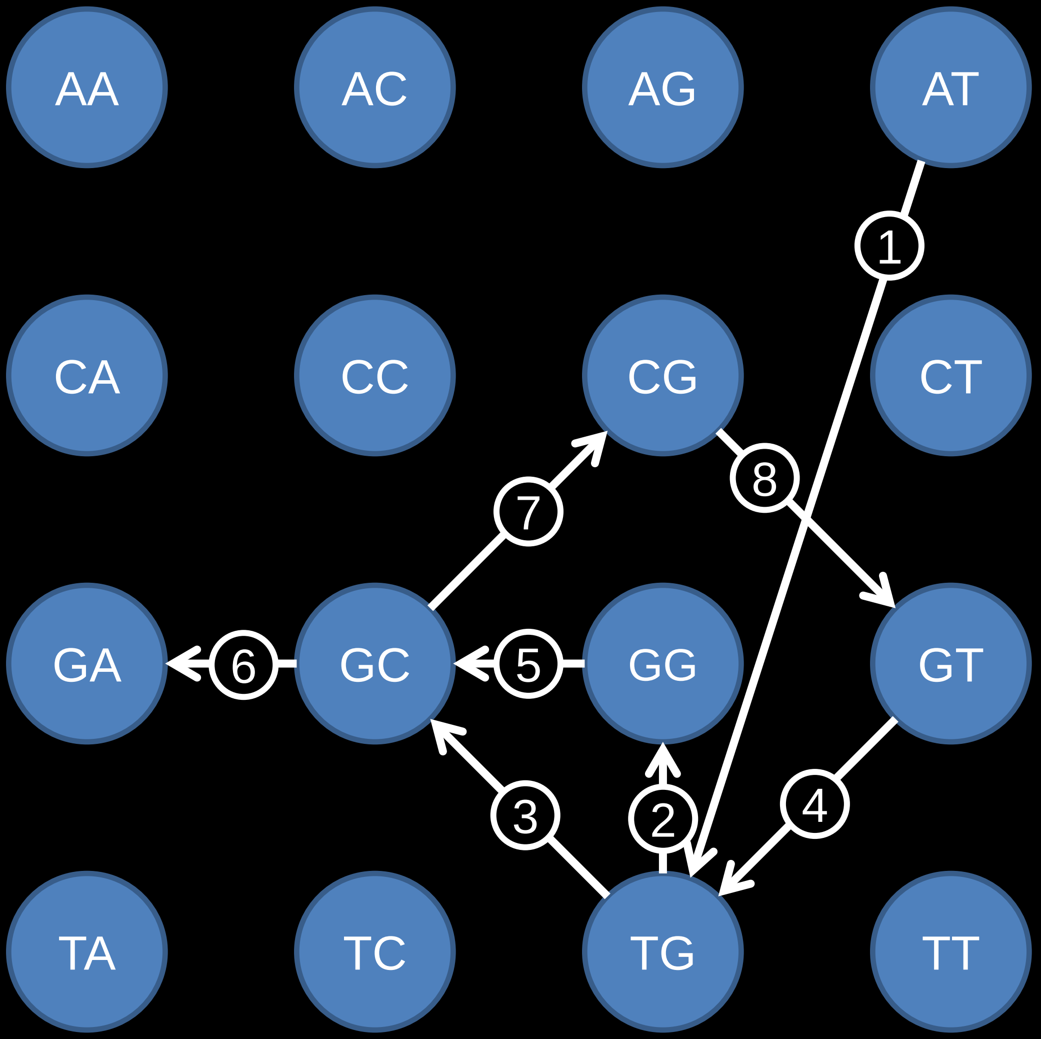
CA CC CG CT

GA GC GG GT

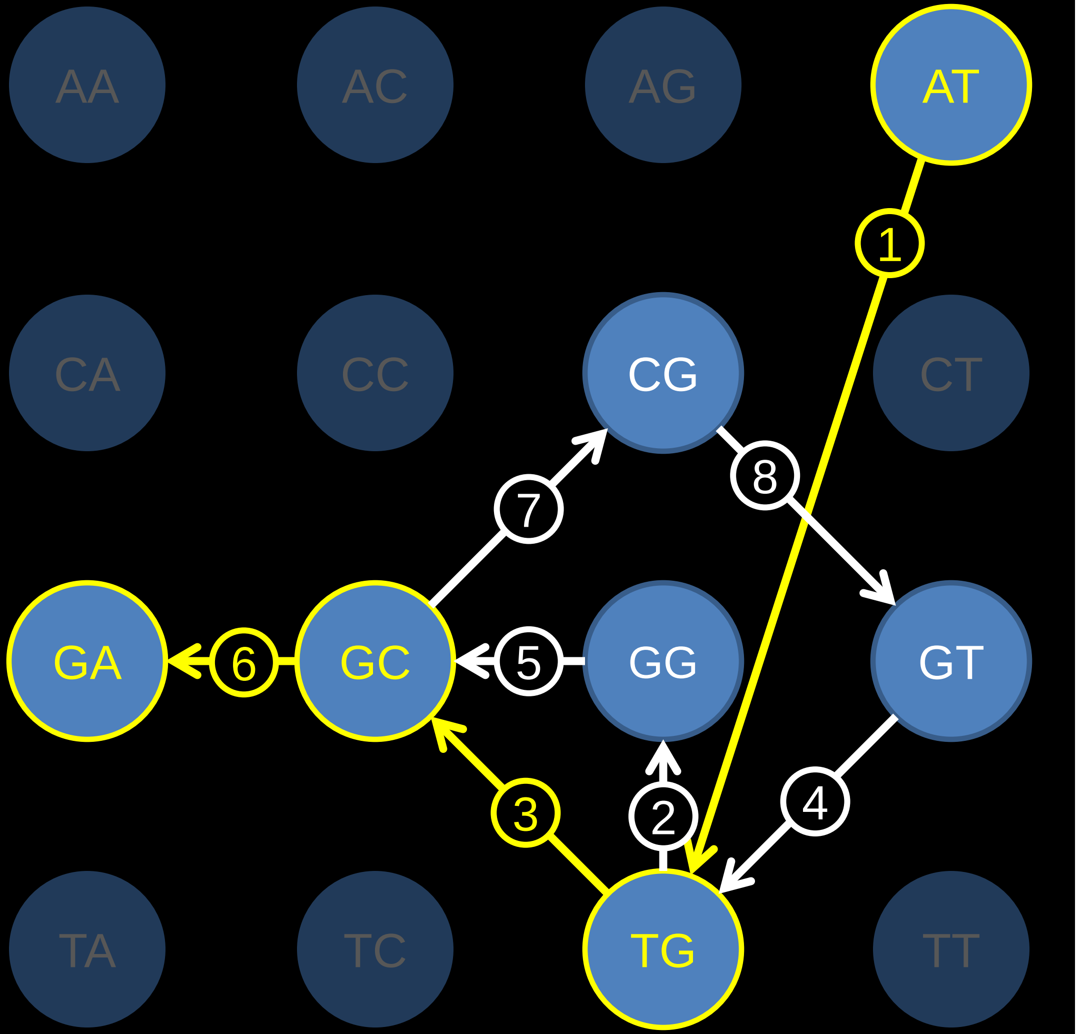
TA TC TG TT

Reads are Edges

1 ~~ATG~~
2 ~~TGG~~
3 ~~TGC~~
4 ~~GTG~~
5 ~~GGC~~
6 ~~GCA~~
7 ~~GCG~~
8 ~~CGT~~



How do we interpret this?



- Each node represents a prefix or suffix.
- Start with an edge with <u>outdegree > indegree</u>
- Assemble the contigs in any edge order:

  ① ③ ⑥
- You cannot reuse edges!
  - Each edge is a read
  - Frequently there will be identical reads: multiple edges between the same nodes.

Questions