DNA Sequencing and Sorting: Identifying Genetic Variations

Overview

Each of the cells in your body contains a copy of your genetic inheritance, your DNA which has been passed down to you, one half from your biological mother and one half from your biological father. This DNA determines physical features, like eye color and hair color, and can determine susceptibility to medical conditions like hypertension, heart disease, diabetes, and cancer.

In this unit you’ll learn about genome sequencing, which is being used to study DNA and identify human genetic variation. The unit introduces concepts in biology and genetics: DNA, chromosomes, and mutations. Mathematical and algorithmic concepts include binary numbers and sorting with Radix Sort.

Human chromosomes

DNA double helix

Neat Things You’ll Do.

1. See how scientists determine what’s different about someone’s genome through a simulation of DNA fragment mapping which identifies where experimentally obtained short DNA fragments fit in a reference human genome.

2. Mapping is made easier when scientists have a “dictionary” of the genome. You’ll learn how to build a dictionary using a method called Radix Sort, which can easily and quickly sort DNA sequences, and students too!
Unit Goals and Objectives

Goal: Increase understanding of DNA and the study of genetics
Objectives:
- Explain the role of DNA in determining a person’s physical and other characteristics (to include mitochondrial DNA and the X and Y chromosomes)
- Describe the relationship of the nucleotides and a DNA strand.
- Describe the science of genetics and the human genome.

Goal: Understand the technique of genome sequencing and its purpose.
Objectives:
- Explain why geneticists work with sequences of nucleotides and not the actual physical structure of DNA.
- Describe the purpose of comparing or mapping a “read” to the reference genome.
- Explain how a suffix array dictionary can assist in mapping.
- Describe the process and purpose of doing a radix sort.

Goal: Analyze the results of genome mapping.
Objectives:
- Identify and explain variations between reads and the reference genome.
- Understand the cause and effects of mutations.
Lesson 1  DNA – Getting Started

DNA (Deoxyribonucleic Acid) is a linear molecule made up of four types of subunits called nucleotides which are labeled A (Adenine), C (Cytosine), G (Guanine), and T (Thymine). The subunits are chemically linked together to form a long chain or DNA strand. Genetics is the study of inheritance of characteristics in organisms. DNA is the molecular carrier of inherited traits.

Scientists who study genetics and DNA usually ignore the physical structure of DNA and work primarily with the sequence of nucleotides. Variation in DNA is caused by mutation, which can be small or large changes in a nucleotide sequence. Everyone’s DNA undergoes mutation. Therefore, genome sequences from different individuals in the same species, while mostly identical, can vary in several ways. This lesson examines the structure of DNA and provides experience in comparing genome sequences directly.

DNA and Chromosomes

Two DNA strands form bonds between complementary nucleotides and these strands twist around each other to form a double helix, that is, double stranded DNA. The complementary nucleotides are the A, T pair and the C, G pair. Human DNA is packaged into 24 chromosomes, labeled Chromosome 1 to 22 and Chromosomes X and Y. A cell has two copies of the numbered chromosomes, called autosomes. In females, there are two copies of the X chromosome, and in males, one copy of the X chromosome and one copy of the Y chromosome. The mitochondria, the numerous energy generating organelles in cells, also each have a small piece of circular DNA called mitochondrial DNA. A genome refers to all the chromosomes in an individual.

Images of DNA

Below is notation for a fragment of double stranded DNA showing the nucleotides in the complementary or opposite strands. The designations 5′ and 3′ (pronounced “five prime” and “three prime”) indicate the direction of the DNA strands. As shown here, complementary strands are oppositely directed.

\[ 5' \text{ACTGCGGCTACTG} 3' \]
\[ 3' \text{TGACGCCGATGAC} 5' \]

Figure 1.1 shows a flattened diagram of a double stranded DNA fragment. The nucleotides within one strand are linked along a phosphate backbone. Complementary nucleotides between strands are hydrogen bonded.
A model of double stranded DNA is below in Figure 1.3. An animated version of this structure is at [http://upload.wikimedia.org/wikipedia/commons/1/16/DNA_orbit_animated.gif](http://upload.wikimedia.org/wikipedia/commons/1/16/DNA_orbit_animated.gif).

Figure 1.3 below shows a micrograph of the 24 human chromosomes, consisting of the numbered autosomes and the X and Y chromosomes. A normal cell contains two copies of each autosome. Females have two copies of the X. Males have one copy of the X and one of the Y. Colors are due to fluorescent staining of the chromosomes.
The lengths of the chromosomes as seen in the image above reflect their actual sizes as shown in Table 1.1 below.\[1][2]

<table>
<thead>
<tr>
<th>Chromosome</th>
<th>Length In Nucleotides</th>
<th>Chromosome</th>
<th>Length In Nucleotides</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>249,250,621</td>
<td>14</td>
<td>107,349,540</td>
</tr>
<tr>
<td>2</td>
<td>243,199,373</td>
<td>15</td>
<td>102,531,392</td>
</tr>
<tr>
<td>3</td>
<td>1980,22,430</td>
<td>16</td>
<td>90,354,753</td>
</tr>
<tr>
<td>4</td>
<td>191,154,276</td>
<td>17</td>
<td>81,195,210</td>
</tr>
<tr>
<td>5</td>
<td>180,915,260</td>
<td>18</td>
<td>78,077,248</td>
</tr>
<tr>
<td>6</td>
<td>171,115,067</td>
<td>19</td>
<td>59,128,983</td>
</tr>
<tr>
<td>7</td>
<td>159,138,663</td>
<td>20</td>
<td>63,025,520</td>
</tr>
<tr>
<td>8</td>
<td>146,364,022</td>
<td>21</td>
<td>48,129,895</td>
</tr>
<tr>
<td>9</td>
<td>141,213,431</td>
<td>22</td>
<td>51,304,566</td>
</tr>
<tr>
<td>10</td>
<td>135,534,747</td>
<td>X</td>
<td>155,270,560</td>
</tr>
<tr>
<td>11</td>
<td>135,006,516</td>
<td>Y</td>
<td>59,373,566</td>
</tr>
<tr>
<td>12</td>
<td>133,851,895</td>
<td>mitochondrial</td>
<td>16,569</td>
</tr>
<tr>
<td>13</td>
<td>115,169,878</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1.1: Lengths of Chromosomes

**DNA, Mutations, and Genetics**

**Genetics** is the study of inheritance of characteristics in organisms. DNA is the molecular carrier of inherited traits. Scientists who study genetics and DNA usually ignore the physical structure of DNA and work primarily with the **sequence of nucleotides** in a chromosome. Variations in the sequence in different individuals in a population can be used to trace how closely those individuals are related and how they have inherited their DNA from their parents.
Variation in DNA is caused by **mutation**, which can produce small or large changes in a nucleotide sequence. Everyone’s DNA undergoes mutation. In fact, since the human body contains something like 10,000,000,000,000 (10 trillion = $10^{13}$) cells and the DNA in a typical cell has a total length of approximately 6,000,000,000 (6 billion = $6 \times 10^9$) nucleotides, there are approximately $6 \times 10^{22}$ nucleotides in your body that could be mutated. To get an idea of how large that is, compare it to the number of atoms (think how small those are) in 12 grams of Carbon-12. That number, approximately $6.022 \times 10^{23}$, defines a **mole** and is also called **Avogadro's number**. Compare it also to the number of stars in the observable universe, which is estimated to be $5 \times 10^{22}$ (taken from Wikipedia: Large numbers). See Figure 1.4.

![Figure 1.4: Comparison of Nucleotides to Atoms and Stars](image)

Given such a large number of nucleotides, even if the mutation rate is extremely small, this amounts to a potentially large number of changes. Fortunately, most of the mutations that occur in your body are lost because the mutations are repaired or the cells with the mutations die. But, some few mutations become permanent. Some of these we pass on to our children, some stay in the cells of our body. Cancers, for example, can arise from mutations. That's why it's important to protect yourself from mutagens like radiation, hazardous chemicals, pesticides, cigarette smoke, and UV sunlight, all of which can dramatically increase the mutation rate.

Below are two examples of the use of genomic sequence variation to draw conclusions about relations between populations.

**Mitochondrial DNA**, mentioned earlier, has a special property. The mitochondria are small organelles within a cell that reside in the cytoplasm. Only the mother, in her eggs, passes them on to children. The father's mitochondrial DNA does not pass to the next generation. That means that there is a line of inheritance in woman of mitochondrial DNA from their female ancestors. Study of genetic variation in mitochondrial DNA from many human populations throughout the world suggests that the mother of all humanity, called the **mitochondrial Eve**, lived in east Africa approximately 200 thousand years ago. This woman was not the only female ancestor of modern Homo Sapien, nor was she the first. Rather, she was the most recent common ancestor of all the tested populations in a direct female line of descent. While other women would have been alive during the time of the mitochondrial Eve, she was the only one whose line of inheritance was not broken at some point by having only male descendants or female descendants who didn’t reproduce.[3]
Below, Figure 1.5 shows the direct line of descent, through mothers, to you from one of your great-grandmothers. “You” are shown as female, but the line of descent is the same if you are male. The mitochondria in your cells are descended from those in your great-grandmother’s cells by the direct line of descent.

The **Y chromosome** also has a special property. Since only males carry a Y chromosome, it is only passed from father to son in each generation. That means there is a direct line of inheritance in men from their male ancestors. Study of genetic variation in the Y chromosome has suggested that a Black South African population, the **Lemba**, are closely related to the **Jewish Kohanim**, the historical Jewish priestly class. The Lemba, despite their distance in time and space from the ancient Middle Eastern Jewish tribes, maintain several Jewish traditions in their religious practices and have oral traditions which have them migrating out of Judea approximately 2500 years ago.[4]
Lesson 2  Comparing Genomes: The Human Genome Reference

In order to study DNA similarities and differences, like those used to determine the age and location of the mitochondrial Eve or the historical connections of the South African Lemba, scientists need a way to observe variation in genomes. Finding differences and similarities among genomes can be accomplished in several ways. One is to compare the genomes directly. Another, which we will discuss below, involves first comparing the genomes to a standard genome reference, and then examining only those parts that differ from the reference. For this approach to work, it is critical that there be a high quality reference already available, and fortunately, for humans, that is the case.

The Human Genome

In February 2001, the initial version of the human genome was published, an event that is considered a milestone in international scientific cooperation and achievement. In the years leading up to its publication, roughly $3 billion were spent to perform the necessary research and experiments. Since then, the reference has been continually refined and its accuracy improved. The current version is referred to as GRCh37/hg19 (Genome Reference Consortium, build 37; UCSC human genome version 19). The sequence is publicly available for computer download from a public repository at the University of California Santa Cruz (UCSC). A view of some of this data is shown in Figure 2.1 below.

Figure 2.1: Portion of UCSC Human Genome Data[5]
The reference human genome was produced from a mixture of DNA from different people and does not represent one specific individual. Below are covers of the scientific journals *Nature* and *Science* from the issues publishing the human genome in 2001.

**Figure 2.2:** The February 15, 2001 issue of Nature, was largely devoted to the human genome.\(^{[6]}\)
DNA Sequencing and Mapping

If the costs of collecting the data for a genome were as expensive today as it was in 2001, there would be no hope of doing genome sequence comparison to find genome variation. But, since then, costs have dropped dramatically as researchers have developed new technologies for collecting genome sequence data. Several experimental techniques are now available, and recently automated genome sequencing became the most popular and least expensive experimental method, using what are called high-throughput sequencing technologies. **Sequencing** determines the order of the nucleotides on each chromosome so that when we look at the representation of a chromosome sequence it appears as a long string of the letters A, C, G,
and T. In 2013 data from sequencing an entire human genome could be generated for under $10,000. These costs are expected to continue to drop steeply, and in your lifetime it is likely that part or all of your genome will be sequenced as a medical test. It may become as common as having a blood sample taken.

The laboratory based sequencing technologies do not produce an entire human genome sequence directly. Rather, they produce short fragment sequences called reads which, like the pieces of a jigsaw puzzle, have to be assembled in some way into a complete genome sequence. The reads are typically very short, ranging in size from 50 to 300 nucleotides long depending on the technology. Below is a sample of reads obtained from sequencing the genome of James Watson, one of the co-discoverers of the structure of DNA. Each read is preceded by a line starting with a ‘>’ character and the read’s label.

---

**Figure 2.4:** Sample “Reads” of James Watson

---

The most common way of putting the reads together is to use a computer to map them to the reference genome. Mapping is a computational method of finding the reference genome location that has the best match for each read. In some ways, mapping is like having a complete picture of the jigsaw puzzle under the pieces and searching the picture to see where each piece fits.

While the jigsaw puzzle analogy is useful, three big differences exist between mapping reads to a reference and assembling a puzzle. First, a jigsaw puzzle is two-dimensional, having length and width. A genome is one-dimensional, having only length. Second, jigsaw puzzle pieces exactly match the picture where they fit. However, while many reads will map exactly to the reference genome, some reads will not match the genome exactly. One reason is that laboratory experiments, which do not produce perfect data, often generate artifacts (experimental errors). A more important reason is the result of mutational differences between the reference and the
person whose genome is being sequenced. If these differences didn’t exist, we would all have identical DNA, like a clone army in the Star Wars movie "Attack of the Clones." (Clones are a group of organisms that all came from one ancestor and have identical DNA.) It is the complication caused by the fact that the reads do not all exactly match the reference that makes mapping a hard computational problem. Third, jigsaw puzzle pieces interlock, but do not overlap. Reads, on the other hand, frequently overlap. In fact, the more reads there are and the more overlapping that occurs, the better the results because it becomes easier to spot artifacts.

**ACTIVITY 2-1 Preparation For Mapping**

**Objective:** Complete a simple mapping and plan strategies for more complex mappings.

**Material:**
- DS-H1: Preparation For Mapping Activity Worksheet

1. Below is a section of the genome along with two reads.
   - genome: `CCATGGTTGAGACCATCGATAAGCTCGACGA`
   - read 1: `ACCATCCGAT`
   - read 2: `CGATAAGCTC`
   
   a. Map each read to a section of the genome.
   
   b. Do the reads overlap? What does this mean?
   
   c. Do the reads exactly match the reference? What does this mean?

2. In tomorrow’s class you will be given a number of reads, similar to those shown in part 1. Your task will be to again find the best matching location for each read in a reference genome. The best location may not match exactly. The reference genome (a fragment of the human genome) will be several hundred letters long.

Think about that task and work with your group to come up with one or more strategies for making your work easier. Write your strategies clearly, so that another group could follow your suggestions based only on what you have written.
Lesson 3  Mapping

In this lesson, you will map simulated subject reads to a real human genome reference sequence. The goal is to discover both the difficulty of mapping and the kinds of variations that can be detected. After you finish the first activity, you will learn about some of the different types of mutational variations and you will identify them in your “data.” In a second activity, you will explore a real method used to make mapping easier by producing a “dictionary” of the reference genome.

Consensus Sequence

A consensus sequence is the sequence with which most of the reads agree. In looking for agreement, you will look for matches to map the read to the sequence. Some reads will disagree due to experimental error. A read maps if it matches exactly or if it matches with one only difference.

ACTIVITY 3-1  Finding Consensus

Objective: To match the reads to the reference genome in order to reconstruct the reads into a single consensus sequence.

Materials:
- Handout DS-H2: Finding Consensus Activity Worksheet
- Handout DS-H3: Reference genome
- Handout DS-H4: Subject reads

1. Look for the best matching location for each read given to your group. Tape the read above or below its match on the reference, so that the read and reference are aligned. Reads that match exactly should be taped below the reference. Those that match inexacty (one error) should be taped above. Do not put reads on top of each other. Give any reads that could not be mapped (more than one error) to your teacher.

2. While you are working, think about strategies for finding the mapped position for each read more quickly. Write clear instructions for the strategies that seem to work best. Be prepared to share your strategies.

Genome Viewer

Below is an image of the Integrated Genome Viewer, a free software tool developed at the Broad Institute, a scientific research institute in Cambridge, Massachusetts, which is used to view reads mapped to a reference genome. The view is similar to what you have accomplished in this activity.
What might be helpful in your effort to map your reads? Perhaps you have started to notice that there are certain groups or runs of nucleotides that if found in your read make it easier to map. For example, you may have a TTTT in your read and you noticed where there was a TTTT in the reference sequence. A **suffix array** is a sorted list of suffixes of a sequence.

**ACTIVITY 3-2 Using a Suffix Array Dictionary**

**Objective:** Use a suffix array dictionary to assist in mapping reads to a reference sequence.

**Materials:**
- Handout DS-H6: Using a Suffix Array Dictionary Activity Worksheet
- Handout DS-H7: Suffix Array Dictionary

1. Review your suffix array dictionary and discuss in your group how the dictionary would have helped you map your reads in Activity 3-1.

2. Use your strategies and your suffix array dictionary to map additional reads assigned to your group.

3. As a class, determine if all posted reads are mapped correctly.
a. Look at the reads that were not mapped. Can your group now find a place for those reads?

b. For those reads above the sequence, circle in red the one error in each read.

**Coverage**

During the class activity you probably noticed that not every letter of the reference genome sequence had exactly the same number of reads aligned with it. However it is possible to describe the overall mapping with a single number, the coverage. Coverage is the average number of reads stacked over a reference genome letter. For small data sets you can determine coverage by literally counting the number of reads at each location and then dividing by the total number of positions in the reference sequence. Larger sets invite more thinking for less work!

**Questions for Discussion**

1. Look at the first 20 nucleotides in your reference sequence. What is the coverage for this subsequence. Use the direct “count and divide” approach to calculate the coverage.

2. Think about the division you had to do in calculating the class coverage number for the first 20 nucleotides. Now imagine determining coverage for the entire reference sequence (300 nucleotides).
   a. The denominator is easy. Why?
   b. You obtained the numerator by counting reads at each location, which requires looking at each location to get its count. But what does the total of all those counts really give you? Can you think of another, simpler, way to come up with that value for the entire reference sequence?

3. Use your work in part 2.b. to invent a formula that can be used to calculate coverage without having to count all the individual matches at each nucleotide location on the reference genome sequence.

4. Remember, the goal of mapping is to discover the correct sequence of nucleotides in the DNA of the actual subject (not the reference). How would you decide on the “correct” subject sequence in locations where the reads were not all the same?
Lesson 4  Genome Variations and Mutations

At the end of Lesson 3 and the two mapping activities you probably had some reads that just didn’t seem to fit. Given that this data came from an actual subject, perhaps we can find an explanation. First, let’s review coverage.

Coverage Review

As you discovered in Lesson 3, coverage is a measure of the average number of reads that are stacked above a randomly selected reference nucleotide by the mapping process. The formula for calculating coverage is $C \times G = R \times N$, where $C$ is coverage, $G$ is genome size, $R$ is read size, and $N$ is the number of reads.

Questions for Discussion

1. Suppose reads are 100 nucleotides long and the reference genome is 3 billion nucleotides long. How many reads does it take to have the same number of nucleotides as the reference genome?

2. Suppose the reads overlap in such a way that every reference nucleotide is covered by exactly four reads. This is a coverage of four. How many reads are there if each read is 60 nucleotides long?

3. Of course, in real data, the reads would not overlap so uniformly; some genome positions may have few or no reads covering them and other may have many. Consider the example of two reads covering part of a genome:
   
   genome: CCATGGTTGAGACCATCCGATAAGCTCGACGA  
   read 1: ACCATCCGAT  
   read 2: CGATAAGCTC

What is the coverage? We can answer this question by inspection or by our formulat.

4. Now consider a more realistic problem in which the reads aren’t all the same length. Suppose reads are 230 nucleotides long on average and that there are 750 million reads (this is NOT an unusually high number for this kind of data). What is the coverage?

Variations

Genome sequences from different individuals in the same species, while mostly identical, can vary in several ways. Some of the most common types of variation are listed below.

- **SNP** or Single Nucleotide Polymorphism. Polymorphism is the term given to the occurrence of a variation in the population at a specific genomic location. There may be two different sequences, or there may be many different sequences. For example, suppose two individuals have the following sequences at a particular location:

  Individual 1: TTGATCGT TTGACC ACT  
  Individual 2: TTGATCGT CTCGACC ACT  
  
  *
Note that the middle nucleotide in these two sequences is different (marked by *). It is T in Individual 1 and C in Individual 2. This is a SNP because it is a single nucleotide difference. Another way to express this difference is to say that the genome location has two alleles. An allele is a sequence that has an alternate form. We could say that this genomic location has two known alleles. It may have more. Currently, as many as 41,000,000 SNPs have been detected in the human genome. They are stored in a public database called dbSNP, which is U.S. Government funded through the National Institutes of Health.\textsuperscript{[8]}

- **Insertion or Deletion.** An insertion is a piece of sequence that has been added. A deletion is a piece of sequence that has been lost. Insertions and deletions are collectively called indels. Small indels, perhaps a few nucleotides long, are very common, but larger indels, say up to a million nucleotides, can also occur. Suppose two individuals have the following sequences at a particular location:

  Individual 1: CGGACTGAACAGGTC  
  Individual 2: CGGACAGGTC

In Individual 2, a small piece, TGAAC, is missing from the middle (between the *s). This is considered a deletion of TGAAC if most people in the population have the same allele as Individual 1. It is considered an insertion of TGAAC if most people have the same allele as Individual 2.

- **Inversion.** An inversion involves removing a piece of DNA, flipping it over, and reinserting it. To show an example, both strands of DNA must be shown with the direction of the strands from 5' to 3' indicated. Suppose two individuals have the following sequences:

  Individual 1:  5' - AAACGGGTGTCAAGGC - 3'  
  3' - TTTGGCCCAAATTTCCG - 5'  

default

  Individual 2:  5' - AAACAAACACCCAAAGGC - 3'  
  3' - TTTGTCTTGGGTTTCCG - 5'

In Individual 1, the middle fragment marked by *s,

  5' - GGGTGTTT - 3'  
  3' - CCCACAAA - 5'

has been flipped end to end to become, in Individual 2,

  5' - AAACACCC - 3'  
  3' - TTTGTGTTG - 5'

Note that flipping before reinsertion, preserves the direction of the strands. A 48,000 nucleotide inversion of this type on the X chromosome, with no apparent effects, has been found to occur in up to one third of females in some human populations.\textsuperscript{[9]}
• **Repeat Expansion and Contraction.** Repeats are very common in DNA. One class is called **tandem repeats.** A tandem repeat contains the same sequence repeated several times in a row. Repeat expansions and contractions involve changing the number of copies of the repeated sequence and are similar to indels. Suppose two individuals have the following sequences at a particular location:

Individual 1: TT GTCAAAC GTCAAAC GTCAAAC AGCCG  
Individual 2: TT GTCAAAC GTCAAAC GTCAAAC GTCAAAC GTCAAAC AGCCG

The repeated sequence is GTCAAAC. It is repeated three times in Individual 1 and four times in Individual 2. It has either expanded in Individual 2 or contracted in Individual 1.

• **Segmental Duplication.** Sometimes, a large region of DNA, or a segment, becomes duplicated in an individual, but unlike tandem repeats, the second copy lies far away from the original copy or even on another chromosome. When the duplicate copies are on the same chromosome they are called **intrachromosomal,** when the copies are on different chromosomes, they are called **interchromosomal.**

**ACTIVITY 4-1 Variations**

**Objective:** Classify variations in reads as experimental errors or mutations.

**Materials:**  
DS-H8: Variations Activity Worksheet

Your teacher will assign your group a section of the reference sequence. Complete the following within your section and be prepared to present your results.

1. Identify the locations in your section of the class mapping where one or more reads differ from the reference by a single nucleotide. (In Activity 3-2 your circled these in red.) Single nucleotide differences could be due to experimental error or to mutations. Experimental errors are usually differences that appear in just one read. Mutations (SNPs) usually appear as a common difference in more than one read. Classify the locations displaying differences as likely experimental errors or as mutations by marking them on the actual reads.

2. Identify regions in your section of the reference sequence where no reads mapped. These are potential deletions. Underline these with a marker.

3. Look at the reads that could not be mapped. Reconsider whether these might map across the deletions in your section. Reads that map across a deletion have their left part matching the left side of a deletion and their right part matching the right side of a deletion. This will leave a part in the middle that was underlined in the previous step, which is the true deletion.

4. Be prepared to discuss the following questions about DNA sequencing with your class.
   a. Why might some regions be not well covered or not covered at all?
   b. Why might some locations have two different sets of reads, one set that agrees on one nucleotide and one set that agrees on another nucleotide? In other words, the difference is not due to experimental error.
c. Why might some regions contain many more reads than the average?

d. Where do reads come from that don't match the genome?

e. Explain why longer reads provide better information for sequencing than short reads do.

Questions for Discussion

5. Identify and explain locations on the reference sequence where there is an experimental error, a mutation or a deletion.

6. Discuss questions 4.a. – 4.e. from Activity 4-1.

Extension

Recall that the human genome is approximately 6 billion nucleotides long. Suppose you wished to locate “matches” within this genome for a completely randomly generated sequence of length \( n \). How large must \( n \) be before it becomes unlikely to find any matches?

This addresses a possible claim that “any sequence you write using A, T, C, G has to match the reference genome somewhere.” Is this true?
Lesson 5  Binary Number Prep For Sorting

As you've seen from the mapping activity, finding the correct mapping location for a read can be difficult and frustrating. You may have tried to simplify the process by reasoning something like this. "Since the read I am working with starts with "AGG," I'll scan along the genome looking for AGG and then see if the rest of the read fits there." This is the basis of one algorithmic approach to mapping. Using this approach, it would be even better if you had an "alphabetical list" of the genome that would keep all the parts that start with AGG together, so that during your scanning, you wouldn't have to look at the parts that don't start with AGG. In the second half of the Mapping Activity, you were given a sorted list of subsequences from the genome sequence in order to try this. The dictionary list made it easier to find the mapping location of each read. In this lesson we will discuss how a sequence can be efficiently “sorted.”

From A Genome To A Sorted Genome

To make the idea of sorting a genome more concrete, suppose we have a very short genome of length 30 nucleotides like this:

\[
\text{T TAGGCTAGCTAGCCCTAGGGCAATAGCT}
\]

We will sort the “suffixes” of the genome. A suffix is the ending part of a word. In this case, the word is our short genome and the 30 suffixes are the following "endings." The number before each suffix is the position in the genome of the first letter of the suffix. Note that the first suffix is the entire genome and each subsequent suffix is one letter shorter:

1  T TAGGCTAGCTAGCCCTAGGGCAATAGCT
2  TAGGCTAGCTAGCCCTAGGGCAATAGCT
3  AGGCTAGCTAGCCCTAGGGCAATAGCT
4  GGGCTAGCTAGCCCTAGGGCAATAGCT
5  CTAGCTAGCCCTAGGGCAATAGCT
6  TAGCTAGCCCTAGGGCAATAGCT
7  TAGCTAGCCCTAGGGCAATAGCT
8  AGGCTAGCCCTAGGGCAATAGCT
9  GCTAGGGCTAGGGCAATAGCT
10 TTAGGCTAGGGCAATAGCT
11 TAGGCTAGGGCAATAGCT
12 AGGGCTAGGGCAATAGCT
13 GGGCTAGGGCAATAGCT
14 CAGGGCTAGGGCAATAGCT
15 AGGGCTAGGGCAATAGCT
16 TTAGGGCAATAGCT
17 AAGGGCAATAGCT
18 GGGCAATAGCT
21 GGCAATAGCT
22 GCAATAGCT
23 CAATAGCT
24 AATAGCT
25 ATAGCT
26 TAGCT
27 AGCT
28 GCT
29 CT
30 T

Now, let’s sort the suffixes alphabetically. Since the suffixes are different lengths, we’ll use the rule that shorter words come before longer words if they match up to the end of the shorter word (see suffixes 27 and 8 below). We get the following list of suffixes (again, the number before the suffix is the position of the first letter of the suffix in the original genome). This is called a suffix array:

24 AATAGCT
13 AGCCCTAGGGCAATAGCT
27 AGCT
8 AGCTTAGCCCTAGGGCAATAGCT
3 AGGCTAGCTTAGCCCTAGGGCAATAGCT
19 AGGGCAATAGCT
25 ATAGCT
23 CAATAGCT
15 CCCTAGGGCAATAGCT
16 CCTAGGGCAATAGCT
29 CT
6 CTAGCTTAGCCCTAGGGCAATAGCT
17 CTAGGGCAATAGCT
10 CTTAGCCCTAGGGCAATAGCT
22 GCAATAGCT
14 GCCCTAGGGCAATAGCT
28 GCT
5 GCTAGCTTAGCCCTAGGGCAATAGCT
9 GCTTAGCCCTAGGGCAATAGCT
21 GGCAATAGCT
4 GGCTAGCTTAGCCCTAGGGCAATAGCT
20 GGGCAATAGCT
30 T
12 TAGCCCTAGGGCAATAGCT
26 TAGCT
7 TAGCTTAGCCCTAGGGCAATAGCT
2 TAGGCTAGCTTAGCCCTAGGGCAATAGCT
18 TAGGGCAATAGCT
11 TTAGCCCTAGGGCAATAGCT
1 TTAGGCTAGCTTAGCCCTAGGGCAATAGCT

The sorted list of suffixes above is much easier to scan to find a matching position for a read. Suppose our read starts "TAG". We only have to look in the sorted list for the suffixes that start with TAG, those suffixes start at positions 12, 26, 7, 2, and 18. We can then compare our read to just those suffixes.

Questions For Discussion

1. How does this suffix array differ from the dictionary used in the last lesson? Is this method of sorting more helpful to identify mappings?

2. How should we deal with a read where the first letter is perhaps incorrect due to an error or SNP?

We easily sorted our very short 30-nucleotide example “genome” by hand, but real genomes are much longer and we need a computational method to produce the sorted suffix array. We will study a computer algorithm called Radix Sort, which can quickly and efficiently sort millions of numbers or millions of short sequences of letters. Its simplicity allows it to be demonstrated by hand in just a few simple steps. For our demonstration, however, we need to be able to represent the letters of a genome by binary numbers.

Binary Numbers

The numbers we are most familiar with are called decimal or base 10 numbers. That is because each position in the number represents a power of 10. For example, the number 5678 can be written as

\[ 5000 + 600 + 70 + 8 \]

or as

\[ 5 \times 10^3 + 6 \times 10^2 + 7 \times 10^1 + 8 \times 10^0. \]

In a binary number, or base 2 number each position represents a power of 2. Let's recall the powers of 2 from 0 to 10 and some others. Complete the following list.

\[
2^0 = 1 \quad 2^1 = 2 \quad 2^2 = 4 \quad 2^3 = 8 \quad 2^4 = 16 \quad 2^5 = 32 \\
2^6 = 64 \quad 2^7 = 128 \quad 2^8 = 256 \quad 2^9 = 512 \quad 2^{10} = 1024 \\
2^{11} = 2048 \quad 2^{12} = 4096 \quad 2^{13} = 8192 \quad 2^{14} = 16384 \quad 2^{15} = 32768 \\
2^{16} = 65536 \quad 2^{17} = 131072 \quad 2^{18} = 262144 \quad 2^{19} = 524288 \quad 2^{20} = 1048576
\]

(a kilo) = $2^{10}$

(a mega) = $2^{20}$

(a giga) = $2^{30}$

(a terra) = $2^{40}$

(tera) = $2^{50}$

(tera) = $2^{60}$

(tera) = $2^{70}$

(tera) = $2^{80}$
Computer scientists often “learn to count” by powers of two. That’s because they are so common when dealing with computers and computer algorithms. In computers, powers of two are used for all calculations. In a decimal number, only the digits 0 - 9 are allowed, but in binary numbers, only the digits 1 and 0 are allowed. They are called bits and represent "on" and "off" in a memory location. A byte is the same as 8 bits and is considered a basic "unit" of computer memory.

Some powers of 2 have become recognized in our language. The number $2^{10}$ is called a kilo because it is close to one thousand. The number $2^{20}$ is called a mega because it is close to one million. In 2013, computer files for music and short videos often have megabyte sizes. A 5 megabyte file contains $5 \times 2^{20} \times 8$ bits. A megabyte is abbreviated MB. The number $2^{30}$ is called a giga because it is close to one billion. Laptop memory (2013) is often 4 gigabytes which is $4 \times 2^{30} \times 8$ bits. A gigabyte is abbreviated GB. The number $2^{40}$ is called a tera because it is close to 1 trillion. Large computer hard disks have memory size (2013) of around 3 terabytes, which is $3 \times 2^{40} \times 8$ bits. A terabyte is abbreviated TB. The last number shown, $2^{64}$ is one more than the largest number which can be stored by a “64 bit” computer in its internal memory.

### Powers of Two And The Chess Board

A story is told about an ancient Indian king and a poor mathematician. The king, tired with the common board games of his time, asks the mathematician to invent a new one. After weeks of thinking, the mathematician comes back with a board with 64 squares and two sets of pieces, each headed by a king. The mathematician calls the game Chess. The Indian king is so pleased with the game, that he tells the mathematician to name his price, offering gold and silver, diamonds and rubies. The shrewd mathematician says to the king, "Your highness, I am but a humble man, I would rather have food for my family than the riches you describe. If you will just give me one grain of rice for the first square on the board, two for the second, four for the third, eight for the fourth, continuing doubling on each square until the 64th square, that will be enough." The king, in surprise at the modest request, urges the mathematician to take the treasure, but the mathematician holds firm, and so the king sends for his agricultural ministers to pay the sum. After a week of calculating, the ministers return to the king with very bad news. The amount of rice for the 34th square would be more than the kingdom produces in a year and the amount for the 64th square would be larger than the largest mountain in India. At first the king is very angry and thinks of chopping off the mathematician's head, but then he recalls how much he enjoys chess, and being perhaps wiser than the mathematician thought, comes up with an idea. He calls the mathematician to his court and tells him that the mathematician and his family must carry away the payment themselves, the payment for the first square on the first day, the payment for the second square on the second day, until the 64th day. But, if on any day he fails to carry away that day's complete payment, then the remaining days are forfeit. The mathematician's smile quickly turns to a frown, but he accedes to the king's terms. By the 28th day, even with all his family and friends and their carts and wagons helping, the mathematician is unable to carry off all the rice and has to forfeit the rest. And while they all eat well for a year, the mathematician never becomes rich.
Questions For Discussion

To get an idea of the size of the numbers in this story, assume a grain of rice weighs 20 milligrams (.020 grams) and that a metric ton weighs 1 million grams. A metric ton is ~2204 pounds, while an English ton is 2000 pounds.

3. How many metric tons of rice were paid on the 28th day?

4. How many metric tons of rice are required for the 34th square of the board?

ACTIVITY 5-1 Writing Binary Numbers

Objective: Convert decimal numbers to binary numbers and binary numbers to decimal numbers.

Materials:
Handout DS-H9: Writing Binary Numbers Activity Worksheet

In a binary number, each position represents a power of 2 and only the digits 0 and 1 are allowed. For example, the binary number

1101001

can be written as

1 * 2^6 + 1 * 2^5 + 0 * 2^4 + 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0.

A binary number can be converted to a decimal number by simply multiplying and adding the terms. Since the terms with a coefficient of zero (like 0 * 2^4) add nothing, they can be ignored. The decimal number equivalent of the binary number above is

1 * 2^6 + 1 * 2^5 + 1 * 2^3 + 1 * 2^0 =
1 * 64 + 1 * 32 + 1 * 8 + 1 * 1 =
64 + 32 + 8 + 1 =
105.

As you can see, it is just necessary to add the powers of 2 with coefficients of 1.

1. Convert these binary numbers to decimal:
   a. 1001101
   b. 1000000
   c. 11111

2. How would you convert a decimal number to a binary number? Discuss this in your group and come up with a strategy. Write the steps of your algorithm.
To convert a decimal number to a binary number, we need to find the powers of 2 that sum to the number. Two methods are described below. Does either match your group’s idea?

Method 1: One way to convert is by repeatedly subtracting powers of 2 from the decimal number. For example let’s convert 386 to binary. Start by subtracting the largest power of 2 that fits in 386, which is 256, and then continue to subtract each smaller power of 2 that fits in the previous result:

\[
\begin{align*}
386 - 256 &= 130 \Rightarrow 1 \times 256 \\
130 - 128 &= 2 \Rightarrow 1 \times 128 \\
64 &\text{ doesn’t fit in 2 } \Rightarrow 0 \times 64 \\
32 &\text{ doesn’t fit in 2 } \Rightarrow 0 \times 32 \\
16 &\text{ doesn’t fit in 2 } \Rightarrow 0 \times 16 \\
8 &\text{ doesn’t fit in 2 } \Rightarrow 0 \times 8 \\
4 &\text{ doesn’t fit in 2 } \Rightarrow 0 \times 4 \\
2 - 2 &= 0 \Rightarrow 1 \times 2 \\
1 &\text{ doesn’t fit in 0 } \Rightarrow 0 \times 1
\end{align*}
\]

So, \(386 = 1 \times 256 + 1 \times 128 + 1 \times 2 = 110000010\).

3. Try converting these decimal numbers to binary using method 1.
   a. 138
   b. 255
   c. 3821

Method 2: Repeatedly divide by 2, record the remainder, and keep the quotient for the next division. When dividing by 2, there are only two possible remainders, 0 if the number is even, and 1 if the number is odd. For example, we can convert 386 again as follows:

\[
\begin{align*}
386 \div 2 &= 193 \text{ with remainder 0 (386 is even)} \\
193 \div 2 &= 96 \text{ with remainder 1 (193 is odd)} \\
96 \div 2 &= 48 \text{ with remainder 0 (96 is even)} \\
48 \div 2 &= 24 \text{ with remainder 0 (48 is even)} \\
24 \div 2 &= 12 \text{ with remainder 0 (24 is even)} \\
12 \div 2 &= 6 \text{ with remainder 0 (12 is even)} \\
6 \div 2 &= 3 \text{ with remainder 0 (6 is even)} \\
3 \div 2 &= 1 \text{ with remainder 1 (3 is odd)} \\
1 \div 2 &= 0 \text{ with remainder 1 (1 is odd)}
\end{align*}
\]

Now, the binary number can be read off directly from the remainders. Note that the high end of the number is the last division: \(110000010\)
4. Try converting the following decimal numbers to binary numbers using method 2.
   a. 3402
   b. 255
   c. 81118

**Binary Codes**

When using binary numbers to represent letters, we make a code for each letter. For example, if we have the 4 DNA letters A, C, G, and T, then we need 4 codes. For our codes, we will use the binary numbers corresponding to the four decimal numbers 0, 1, 2, and 3. All the codes need to be the same length, so we will make each code the length of the longest number by adding zeros on the left if necessary. The largest decimal number always has the largest binary number. In our case, 3, which is binary 11 has a length of two, so all the codes will have a length of two. It is important to assign the codes so that when the codes are sorted in numerical order, the letters are in alphabetical order.

<table>
<thead>
<tr>
<th>DNA Letter</th>
<th>Decimal Number</th>
<th>Binary Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>00</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>01</td>
</tr>
<tr>
<td>G</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>T</td>
<td>3</td>
<td>11</td>
</tr>
</tbody>
</table>

**Figure 5.1:** DNA letter Decimal Number Binary Code

**Practice**

1. Suppose we were encoding protein sequences. Proteins are made of 20 amino acids, so we would need 20 codes. What decimal numbers would we use? What is the code for the 20th amino acid? What would be the length of the codes? What is the code for the first amino acid?

2. Computers use binary encoding to represent the letters and symbols on a keyboard. When you type on a laptop or tablet or computer keyboard, or text on a cellphone, a binary code is sent to the computer processor for each letter that you type. Originally, there were 128 codes and they were called ASCII characters (pronounced "ask-ee"). They consisted of 52 English letters (both upper and lowercase), the digits from 0 to 9, special characters like &, ?, !, and non-printable characters like "space", "tab", and "delete". Since there are 128 codes, the numbers 0 to 127 are used. 127 is binary is 1111111 which has length seven, so all the codes have length seven. A table of the ASCII codes is shown below.
In the table, the code for each character can be determined by writing the codes for the column, followed by the codes for the row. For example, the code for the capital letter ‘R’ combines 101 (column) and 0010 (row to obtain 1010010 which equals 82 in decimal.

a. Find the binary code for lower case “e” using ASCII.
b. Find the binary number for the equal sing “=” by converting 61 to a binary number. Check using the column-row method.

3. Today, binary encodings are needed for all the world’s alphabets.
   a. What is the standard for all codes now?

   b. How many characters can Unicode code? What is the length?
Lesson 6  Radix Sort

A **radix** is the number of digits used in a numbering system. For the decimal system, the radix is 10. For the binary system, the radix is 2. In Radix Sort, the number of subgroups that the sequences are divided into, in each round of sorting, equals the radix. In our demonstration, we will repeatedly divide the sequences into two groups.

Normally, to sort words alphabetically by hand, we start at the first letter (leftmost in the word). The counter-intuitive aspect of Radix Sort is that we sort starting with the last letter (rightmost in the word).

**ACTIVITY 6-1  Sorting Students**

**Objective:** To understand the process of radix sorting.

**Materials:**
Handout DS-H10: Sorting Students Activity Worksheet

1. Your DNA sequence.
   a. Write down a five-letter DNA Sequence of your choice (using A, C, G, T).
   b. Convert your sequence to binary code.

   **Sequence:** ____ ____ ____ ____ ____

   **Binary Code:** __ __ __ __ __ __ __ __ __ __  

2. What process would you use to have the entire class line up in alphabetical sorted order? Discuss in your group.

3. Your teacher will use the following procedure to “sort” the class. Be prepared to discuss why this sorting technique works.

   a. All students in the class line up shoulder to shoulder (side by side). Line up in a location with enough space to allows students to step forward and walk to rejoin the line in a new position.

   b. Look at the digit being sorted (for the 1st round, look at your 10th position).
      i. **ZEROS Forward.** Look at your digit. If your digit is 0 step forward from the line. If your digit is 1, stay in place.
      ii. **ZEROS Right/ONES Left.** All students holding a 0 move to the right past all students holding a 1, while students holding 1 shift left.
      iii. **ZEROS Back.** Students holding a zero take a step backwards to rejoin the line in the space left by the students with a 1.
      iv. **CHECK.** Starting with the rightmost student, each student calls off the digit they just sorted. First students will have 0 and once a student reports he or she has a 1, all the rest of the students should have a 1. After each even numbered sort (sort # 2, 4, 6, 8, 10), starting with the rightmost student, each student calls off the
actual letters sorted so far from their sequence. After two rounds, the first student should have digits 00 and will call off “A.” The As will be followed by the Cs and then the Gs and then the Ts. After four rounds the first student might have 0000 and call of “AA.”

v. **REPEAT.** Follow the process in b. i-iv for each subsequent round until the first digit in your sequence number is sorted.

4. Was your class able to follow the procedure to sort the DNA sequences? Why did this process work? What was the key?

**The Radix Sort Deck**

A Radix Sort Deck contains cards that are labeled with a DNA sequence and the binary number associated with that sequence. As you know, there are four possible letters (A, C, G, T) and four binary codes (A = 00, C = 01, G = 10, T = 11). Each binary code is represented by two holes in the card on top and two holes on the bottom. On the top, a zero has a small hole and a one has a long cutout hole to the edge of the card. On the bottom, the types of holes for each binary digit are reversed.

Each Radix Sort deck contains 28 cards and a Cover card. Each regular card is labeled with a DNA sequence of 7 letters. Above each letter is a two-digit binary number or code for the letter. The holes (and the binary digits) along the top are numbered from 1 on the left to 14 on the right.

**ACTIVITY 6-2    Sort Using The Radix Sort Deck**

**Objective:** Explore radix sorting using a sort deck.

**Materials:**
- Handout DS-H11: Sort Using the Radix Sort Deck Activity Worksheet
- Radix Sort Card Deck
- Paperclips

1. Look at the top card on your deck of cards.
   a. How long is the DNA sequence?
   b. How many digits are in the binary number that represents your DNA sequence?
   c. If there is a card in the deck for all the possibilities of 7-letter DNA sequences, how many cards are in the deck (do not count yet)? After you answer, then count to see if you were correct.

2. Shuffle the cards making sure to keep them face up with the DNA sequences upright and all in the same orientation.

3. Let’s sort the cards.
   a. **PAPERCLIP the holes.** Starting with the final digit (in the first round, the hole numbered 14, pass one paperclip through the hole on top and one paperclip through the hole on the bottom. Once the clips are all the way through the deck, carefully pull them
apart. Some cards will stay with the top paperclip and the remainder will stay with the bottom paperclip (see figure). The cards on the top paperclip all have a "0" at position 14 in the binary code. The cards on the bottom paperclip all have a "1" at position 14.

b. **ZERO in front.** Carefully remove the paperclips and stack the cards from the top paperclip (those with a "0" at position 14) in front of the cards from the bottom paperclip (with a “1” at position 14). It is very important that the cards not be allowed to get out of order when removing the paperclips.

c. **CHECK.** Look at your cards after the first round and make sure all cards with a 0 in position 14 are in front of all those with a 1 in position 14. After each even numbered round, check that the final letters are in alphabetical order.

d. **REPEAT.** Move to the next hole and repeat steps a – c for each subsequent round until the first digit in your sequence number is sorted. If you can’t remember which hole you just sorted, look at the top of the deck. One of the holes will have all the slots in the back. That’s the hole you just sorted.

4. Did your radix sort work and how would you know?

5. Was your group able to follow the procedure to sort the DNA sequences? Why did this process work? What was the key?
Exploration

1. Determine what happens when sorting starting at digit 1. After sorting on digit 1, all the sequences with a “0” at digit 1 are followed by all sequences with a “1” at digit 1. How are these sequences grouped by first letter?

2. Clearly the letters are not sorted yet because each code has two digits. So, let’s repeat the sorting steps for digit 2. Again, how are the sequences grouped by first letter?

3. Something is wrong. The sequences are divided into groups, but the order is wrong. Let’s look at the codes. Write down the first (leftmost) letters and the codes from digits 1 and 2, in order as they appear in the deck. Also write the decimal number for each code:

<table>
<thead>
<tr>
<th>Letter</th>
<th>Letter Code</th>
<th>Decimal number for Code</th>
<th>Decimal number for reverse Code (#4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digit 1</td>
<td>Digit 2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. Notice that the decimal numbers are almost, but not quite, in order. Look closely at the codes you have written. They’re actually sorted, but you have to read them backwards. Try writing the decimal numbers for the codes if you read the codes backwards. For example, find the row with the letter G. It’s code is 1 0 which is 2 in decimal. But, if we reverse the code, then it is 0 1 which is 1 in decimal. In the fifth column write the decimal number for the reverse codes.

5. This gives a clue to how to sort correctly. Let’s sort hole 2 before hole 1. Reshuffle your sequences, and sort by digit 2 first, then digit 1. Look at the sequences and describe how the first letters are grouped.

6. What we’ve done so far is sort on one letter. How can this be extended to the first two letters? Based on the last experiment, two suggestions are to
   - Sort on 2, then 1, then 4, then 3.
   - Sort on 4, then 3, then 2, then 1.
Try both methods and write down how the first two letters are grouped when you are done. Remember to reshuffle your sequences before you start.
Practice

1. The Radix Sort algorithm appears to be very fast. We could sort all 16,384 possible DNA sequences of length 7 ($4^7 = 16,384$) in just 14 steps. The time on a computer is actually proportional to the product of: 1) the number of sequences, 2) their length, and 3) the length of the binary codes for the DNA alphabet. Our Radix Sort demonstration seems to have a time cost proportional only to the length of the sequences (7 letters) and the number of binary digits in the letter codes (2), that is, it only takes us 14 steps. The number of sequences is missing. Describe what you think are the "hidden" time costs for our Radix Sort demonstration.

2. Suppose you want the sequences to appear in "reverse" sorted order, that is, starting with the cards that begin TT… and ending with the cards that begin AA…. This only requires one modification to the sorting steps above. Explain the modification.

3. A DNA molecule is formed by two strands. Each strand is composed of nucleotides linked end-to-end. A strand "points" from the 5' carbon in one nucleotide to the 3' carbon in the next nucleotide. Sequences are usually written 5' to 3'. For example

$$5' \text{ AGTACCT 3'}$$

The sequence in one strand in a DNA molecule is the "reverse complement" of the sequence in the other strand. A reverse complement sequence is formed by 1) reversing the sequence and 2) writing the complement for each letter. The complements are A/T and C/G. For example, the reverse complement of the sequence above is obtained by first reversing the sequence to

$$\text{TCCATGA}$$

and then writing the complement of each letter. A replaces T, T replaces A, C replaces G, and G replaces C.

$$\text{AGGTACT}$$

The reverse complement also runs from 5’ to 3’, so it would be written like this:

$$5' \text{ AGGTACT 3'}$$

When together in the same DNA molecule, the strands point in opposite directions, so the two strands together would look like this:

$$5' \text{ AGTACCT 3'}$$
$$3' \text{ TCATGGGA 5'}$$

Suppose you want to sort the sequences in order by their "reverse complement" by modifying the sorting steps above. Explain the modifications.
4. Suppose we sorted on letters directly instead of using binary codes to represent the letters. In that case, the radix would be four (four different letters) and the sorting would first produce four group of sequences, one group for A, one for C, one for G, and one for T. Then the groups would be combined. In what order should they be combined? Do we still need to start at the rightmost letter when sorting multi-letter sequences?

5. This question refers to the mapping activity. Suppose you have the suffix array of the genome, but one of your reads comes from a reverse complement strand. How can you find where it maps using the suffix array?

**Why Radix Sort Works**

Radix sort relies on three properties to work. After sorting on a column:

**Property 1.** All the letters in that column are sorted.

**Property 2.** The words with the same letter in that column end up in the same order they had before sorting on that column.

**Property 3.** Words starting with that column are in alphabetically sorted order.

Let's see these properties at work by referring to the Figures below. We’ll assume that letters instead of binary codes were used in radix sort. That means, each column is sorted into four groups, one for each letter and all the A’s go in front, then the C’s then the G’s, then the T’s.

![Radix Sorting Steps](image)

**Figure 6.1:** Radix Sorting Steps

**Step 1** - Radix sort starts in the last column (blue arrow and yellow letters). When that column is sorted, properties 1 and 2 are true. Now, look at the rows with final letter A. They are in the same
order as in the unsorted rows (red arrows on the left). We can tell this because the arrows do not cross. It is also true of the words with final letter C, with final letter G, and with final letter T. This shows that property 2 is true.

**Step 2** - The next step in radix sort is to sort the middle column, which makes Property 1 true. For each middle column letter, all the rows that end "A_" come before all the rows that end "C_", which come before all the rows that end "G_", which come before all the rows that end "T_". Therefore, Property 3 is at least partly true, but we need to check the rows that end “C_”. Are they also sorted amongst themselves?

Look at the words with middle letter C. They are in the same order as they were in Step 1. Again, the red arrows to the left do not cross (Property 2 is true). This means that all the rows that end "C_" are also sorted amongst themselves because the last letters were sorted in Step 1. For example the row "TCA" comes before the row "TCC" because final A comes before final C in Step 1. Also, “ACG” comes next because final G follows final C in Step 1. This is also true of all the rows that end “A_”, “G_”, and “T_”. This satisfies the remainder of Property 3.

**Step 3** - Radix sort ends after the first column is sorted so Property 1 is true. As discussed previously, this guarantees that all rows that end "A_" come before all rows that end "C_", etc., satisfying part of property 3. Look, for example, at the rows which start “G_”. They are in the same order as they were in Step 2, the red arrows to the left do not cross (Property 2 is true). As described previously, this guarantees that all the words that end "G_" are sorted amongst themselves, satisfying the remainder of Property 3. Now, the words are completely sorted because three letters make up an entire word.

**Sorting History**

Below is a 1950s advertisement from the journal Radiology, which shows the use of a type of radix sort for sorting radiology patient records. It’s called the “Keysort Radiology System.” Note the punched holes in the edges of the patient record card and the long handled needle for sliding through the holes.

![Figure 6.2: Radiology Sorting Advertisement](Radiology, Vol 60, No 5, May 1953)
Glossary

Allele - one of two or more alternate forms of a genomic sequence. Often, one allele contains a mutation relative to the other allele.

Artifacts – in genome mapping refers to experimental errors.

ASCII - American Standard Code for Information Interchange - the original computer codes for letters, numbers, and symbols in the English language.

Autosomes – any of the numbered chromosomes (and not the lettered X or Y sex chromosomes).

Binary number – a number represented in the number system that has 2 as its base and only uses the digits 0 and 1.

Bit – the smallest unit of data or information in a computer. It has the single binary value of either 0 or 1.

Byte – a basic unit of computer memory made up of 8 bits.

Chromosomes – a structure within the cell that carries the genetic materials.

Clones – a group of organisms that all came from one ancestor and have identical DNA.

Complementary nucleotides – nucleotides that form bonds and DNA strands that twist around each other to form a double helix.

Consensus Sequence – the sequence with which most of the reads agree.

Coverage - average number of reads mapped to each reference genome location.

Deletion – a piece of sequence that has been lost.

DNA (Deoxyribonucleic Acid) – a linear molecule made up of four types of subunits called nucleotides labeled A (Adenine), C (Cytosine), G (Guanine), and T (Thymine).

DNA Strand – either of the two chains that makes up the double helix of DNA.

Double helix – a pair of parallel helices intertwined as in the structure of a DNA molecule.

genetics – the study of inheritance of characteristics in organisms.

genoce – a full set of chromosomes; the inheritable traits of an organism.

genome sequencing – the process of sequencing the complete DNA of an organism’s genome.
**DNA Sequencing and Sorting**

**giga** – denoting a factor of $10^9$ or a billion; in terms of bytes is the number $2^{30}$.

**Indel** - insertion or deletion. A difference at a particular location in two sequences, consisting of the absence of a particular subsequence in one sequence that is present in the other sequence.

**Insertion** – a piece of sequence that has been added.

**Interchromosomal** – duplicate copies of a region of DNA between difference chromosomes.

**Intrachromosomal** - duplicate copies of a region of DNA that are on the same chromosome.

**Inversion** - the replacement of a subsequence of nucleotides by its reverse complement (where A and T are complements and C and G are complements).

**Kilo** – denoting a factor of $10^3$ or one thousand; in terms of bytes is the number $2^{10}$.

**Mapping** - the process of matching reads to their corresponding locations in a reference genome.

**Mega** – denoting a factor of $10^6$ or one million; in terms of bytes is the number $2^{20}$.

**Mitochondria** – the numerous energy generating organelles in cells.

**Mitochondrial DNA** – the small piece of circular DNA found in mitochondria.

**Mitochondrial Eve** – the mother of all humanity within the context of a line of inheritance in women.

**Mutation** - changes in the nucleotide sequence, includes SNPs, indels, and inversions.

**Nucleotide** – the basic structural unit of nucleic acids such as DNA.

**Polymorphism** – the occurrence of a variation in the population at a specific genomic location.

**Radix** - the number of digits used in a numbering system. For the decimal system, the radix is 10. For the binary system, the radix is 2. In Radix Sort, the number of subgroups that the sequences are divided into equals the radix.

**Radix Sort** - the process of sorting a list of numbers by first sorting on one specific digit (keeping all numbers is their same relative order within the groups formed by the “sort” digit), then sorting the resulting list by another digit, and so on.

**Read** - a short fragment of a DNA sequence produced by a sequencing experiment.

**Reference genome** – a digital nucleic acid sequence database assembled as a representative example of a species’ set of genes.
**Segmental duplication** – a duplication of a large region or a segment of DNA that lies far away from the original copy or even on a different chromosome.

**Sequence** - a string consisting of the four basic nucleotides (A, C, G, and T) that appear in a particular strand of DNA.

**Sequence of nucleotides** – the succession or sequence of order of the nucleotides within a DNA molecule.

**Sequencing** – in terms of DNA, the determining of the order of the nucleotides on each chromosome.

**SNP - Single Nucleotide Polymorphism** - a difference at a particular location in two sequences, consisting of one nucleotide’s being present in one sequence while a different nucleotide is present in the other sequence.

**Suffix array** – a sorted list of suffixes of a sequence.

**Tandem repeat** – the same sequence of nucleotides repeated several times in a row.

**References**


