COSC 494/594 – Homework #1: Sequence Manipulation and Probabilistic Modeling
(Reformatted using ChatGPT and updated from 2023 instance)
Due Date: August 28, 2025

Reading:

- Chapter 1 (Introduction)
- Sections 3.1–3.4 and 3.6 (Sequence models and basic probability)

Let your instructor know if you use any third-party libraries or online tools beyond standard programming packages.

---

1. Academic Integrity Statement (2 points)

Review the Honor Code in the syllabus and the supplemental document. In your `report.txt`, explicitly state:

"I have read and agree to the terms of the COSC 494/594 Honor Code."

---

2. Retrieve a Genomic Sequence (1 point)

Download the full genome of **bacteriophage lambda** from NCBI (accession **NC_001416.1**).

- Save the file as `lambda.fasta` and include it in your submission.

---

3. Reverse Complement Generator (5 points)

Write a program (in any programming language) to compute the **reverse complement** of the sequence in `lambda.fasta`.

- The output should be saved as `lambda.rev.fasta` and should include the FASTA header: `>reversed`
- In `report.txt`, include instructions on how to run your program (e.g., command-line usage).
- Include your source code file(s) in the submission.

---

4. Nucleotide and Dinucleotide Frequencies (5 points)

Create a program to compute and report:

- The frequency of each nucleotide (A, C, G, T)

- The frequency of each dinucleotide (e.g., AA, AC, AG, ..., TT)

Output Requirements:

- Include a summary table in `report.txt`
- Save the source code and include brief instructions for use

5. Additional Sequences (1 point)

Download the following FASTA sequences from NCBI:

- Human mitochondrial genome (**NC_012920**) → Save as `human_mito.fasta`
- Neanderthal HVR I region (**AF254446**) → Save as `neander_sample.fasta`
  Include both files in your submission.

6. Sequence Probability Modeling (8 points)

Using the **human mitochondrial genome** (`human_mito.fasta`):

- **Train a multinomial model**: Estimate individual nucleotide probabilities
- **Train a third-order Markov model**: Estimate conditional probabilities of each nucleotide given the previous three
- Compute the **log-probability** of the **Neanderthal sequence** under both models

Output Requirements:

- Show both computed log-probabilities in `report.txt`
- Discuss any surprising results or assumptions (e.g., unknown bases)
- Submit all source code and instructions for use

7. Markov-Based Random Sequence Generator (8 points)

Use the third-order Markov model trained in Task 6 to generate a **synthetic DNA sequence of 20,000 bases**.

- Save the output in FASTA format as `markov_simulated.fasta`
- Include source code and run instructions in your submission
- In `report.txt`, include a short commentary on how realistic the sequence appears (e.g., dinucleotide frequencies)

HINT: Code from Task #4 can be reused to sanity-check this prior to submission.

✅ Submission Checklist:

```
python
CopyEdit
submission/
├── lambda.fasta
├── lambda.rev.fasta
├── human_mito.fasta
├── neander_sample.fasta
├── markov_simulated.fasta
├── reverse_complement.py
├── frequency_analysis.py
├── model_probability.py
├── markov_generator.py
└── report.txt
```

🧮 Grading Rubric Summary:

Task	Description	Points
1	Honor code agreement	2
2	Download bacteriophage genome	1
3	Reverse complement code + output	5
4	Frequency analysis code + output	5
5	Download human/Neanderthal sequences	1
6	Probability model code + log-probability calculations	8
7	Markov model simulation + discussion	8
	Total	**30**