Exam recap

Split into topic, book chapter, sample questions. Note that you still need to go over the lectures covering these topics when the lecture and book disagree the lecture wins! For any material that was covered already in midterms or homeworks, those questions are also fair-game for the final.

1. Basics - Chapter 1

- What is the central dogma of biology?
- What is the difference between transcription and translation?
- How do you reverse complement a DNA string?
- How do you translate a DNA string?

2. Inexact alignment & multiple alignment- Chapter 6

- Describe the subproblem, initial conditions, recurrence, and location of answer for global alignment between two sequences.
- Same as above for local alignment between two sequences
- Same as above for finding an overlap between two sequences
- Same as above for identifying if a sequence is a substring of another one
- Describe an algorithm for computing the multiple alignment of two strings. Please write out the pseudo-code and briefly outline the inputs/outputs of the functions you use.
- What are some differences between a "star" alignment algorithm, and a guide-tree alignment algorithm?

3. Alignment heuristics - Chapter 7

- The FASTA program uses the concept of "heavy" diagonals in order to speed up the alignment of sequences - please define what a "heavy" diagonal is.
- The BLAST program uses a relatively simple heuristic to quickly identify matches of a sequence within a database, however it has been very successful due to the clever use of statistics to further refine the results. Briefly describe the main idea behind this statistic approach.
- Assume you want to find an inexact match with at most 2 errors of a DNA sequence of length 20 within the human genome (3Gbp). Describe a strategy for efficiently finding such an alignment that is faster than simply running a dynamic programming algorithm.

4. Phylogenetic trees - Chapter 12 (excl. 12.4)

- In class we described two approaches for phylogenetic reconstruction - parsimony-based, and distance-based. These correspond to different goals for the structure of the resulting tree. Briefly outline what goal is achieved by parsimony methods. Can a distance method be used to construct a maximum parsimony tree?
- Describe the intuition behind the correction factors in the Neighbor-Joining algorithm.
- Exercise 5 in the book
- Given the distance matrix from Exercise 4, construct both the UPGMA and the Neighbor-Joining trees.

5. Motif finding - Chapter 9.1 - 9.2.1

- What is the difference between a position-weight-matrix (PWM) and a position-specific scoring matrix (PSSM)?
- In class we described a Gibbs sampler, an algorithm for finding motifs in a DNA sequence. Can the algorithm described in class identify all the motifs in a stretch of DNA?
- One step of Gibbs sampler algorithm involves identifying a section of a DNA sequence that best matches our current best guess at a motif, represented as a multiple alignment. What data-structure would you use to store this multiple alignment? Using this data-structure write the pseudo-code for identifying this best match between a sequence and the already computed multiple alignment.

6. Genome assembly - Chapter 4.5, Chapter 8.

- The Lander-Waterman model describes the expected number of contigs (N) in a genome project as a function of the genome length G, read length L, depth of coverage c, and the overlap between sequences o. Without remembering the exact formula, sketch the rough shape of the dependency between N and c, assuming G, L, and o are fixed.
- Assume you need to write a genome assembler. Briefly describe the algorithm you would use and outline it in pseudo-code. Assume you have available functions to determine whether pairs of sequences overlap, to merge multiple-alignments, and to compute the consensus sequence of a multiple alignment.
- Given the overlap matrix from page 207 in the book, describe how you would assemble these sequences and list the order in which the sequences will appear in the assembly.

7. Microarrays and data clustering - Chapter 10, Chapter 11.1-11.6

- Assume you have run a microarray experiment and received a file containing the raw intensity values at each position in the array. What is the first task you need to perform before you can identify a set of genes that are associated with the disease you are studying?
- Cluster the data in exercise 7 using nearest neighbor and farthest neighbor. What are the differences between the resulting clusters?
- What is the running time of the k-means clustering algorithm?
- k-means clustering requires a stopping condition - how would you decide when to stop the algorithm?

8. Proteomics - Chapter 11.7

- Why are proteomic approaches necessary even though technologies for analyzing RNA (microarrays, sequencing) are better developed?
- Briefly outline an algorithm for de novo identification of proteins from mass spectra (note - was discussed in class but not in book)

9. RNA folding - not in book

Due to low attendance in class this topic will be on the exam even though it's not in the book.

- Describe an extension to Nussinov's algorithm that allows you to specify a "per-stem" score, i.e. a fold that contains k adjacent matched bases (a stem of length k) will obtain a score f(k) instead of k (the score assuming each match results in a score of 1), where f is an arbitrary function.
- How would you modify Nussinov's algorithm to perform "local folding", i.e. identify a stretch of a DNA sequence that forms a good quality hair-pin, however the fold doesn't need to involve the whole sequence. This problem arises in the identification of transcription terminators in bacterial genomes - these are short segments of DNA that fold into a hairpin structure and that block the activity of the RNA polymerase.
- How would you modify Nussinov's algorithm to ensure that all loops are at least 10 nucleotides in length?