Final Exam

Name: ______________________ Signature: ______________________

ID #: __________________

INSTRUCTIONS:

• This is a closed book, closed mouth exam.
• You may use either pen or pencil.
• Check to see that you have 6 exam pages plus this cover.
• Use only the space allotted. Do not write on the back of the page.
• Each fill-in is worth two points.
• Good luck!!

<table>
<thead>
<tr>
<th>Problem</th>
<th>Score</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bio 1</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>Bio 2</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>Bio Total</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>CS 1</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>CS 2</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>CS 3</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>CS Total</td>
<td>55</td>
<td>55</td>
</tr>
</tbody>
</table>

(up to 20 points) Are you certifiably a biologist but not a computer scientist? If so, explain your training in both biology and computer science/engineering to see if you should get algorithm handicap points.
Biology Questions

1. The four letters of the DNA alphabet are ______________

2. A gene is a DNA sequence that ______________

3. The genome of a bacteria is typically about ______________ base pairs long.

4. The human genome is about ______________ bases long.

5. The homology between genes in different organisms is a natural consequence of ______________

6. PCR stands for ______________ and is used to ______________

7. Gel electrophoresis separates DNA fragments by ______________

8. High levels of coverage is needed in a sequencing project to avoid the problem of ______________

9. List three problems that make fragment assembly difficult:
   1. ______________
   2. ______________
   3. ______________

10. A chimera is a ______________

11. One reason why gaps occur in homologous sequences is ______________

12. ______________ and ______________ are the names of heuristic algorithms for sequence alignment.
13. Smith-Waterman differs from edit distance computation because it computes

14. (5 points) What is an advantage of a curated sequence database over an uncurated database?

What is an advantage of an uncurated sequence database over a curated one?

15. (6 points) You are given two sequence alignments. One is the alignment of two DNA sequences with an identity of 36% (i.e. 36% of the bases match in the alignment). The other is the alignment of two amino acid sequences, with an identity of 22%. Which of the two alignments represents greater biological similarity between sequences? Explain why.
Computer Science Questions

1. Describe an $O(mn)$ algorithm for exact string matching, i.e. which tests whether string $S$ occurs as a substring of string $T$, where $|S| = m$ and $|T| = n$. (5 points)

2. Describe how to use suffix trees to efficiently search if a string $s$ is a substring of string $t$. (5 points)

3. Construct the suffix array of abracadabra. (5 points)

4. To find the longest common subsequence of two strings, we can modify the edit distance function by make the cost of _______ very high. (5 points)
5. (15 points) Consider the following simple incremental-insertion suffix tree construction algorithm for a string $S$. Start by inserting the first suffix $S[1, \ldots, n]$ (the entire string), then the second suffix $(S[2, \ldots, n])$ by walking down from the root until the suffix separates, and so on for all $n + 1$ suffixes.

Assume you are using a compressed (i.e. linear space) suffix tree representation. Also assume there is a distinct end of string character, so every suffix is associated with a distinct leaf in the tree.

- What is the worst case running time for this algorithm on an $n$ character string for a constant-sized alphabet (e.g. $\alpha = 4$)? Why?

- What is the expected running time for this algorithm on a random $n$ character string for a constant-sized alphabet (e.g. $\alpha = 4$)? Why?

- What is the best case running time for this algorithm on an $n$ character string for a constant-sized alphabet (e.g. $\alpha = 4$)? Why?
6. (20 points) Define a no-deletion alignment between two strings \( X \) and \( Y \) of length \( n \) and \( m \) as one where only insertion, match, and substitution are allowed. No deletions from \( Y \) are allowed. Clearly \( m \geq n \).

(a) Briefly explain how to modify our standard edit distance algorithm to solve this problem in \( O(mn) \) time.

(b) Let \( k = m - n \). Give an \( O(kn) \) algorithm to find the optimal no-deletion alignment.