Name: __________________________________________

You are to honor the Emory Honor Code and the Math/CS SPCA. This is a closed-book and closed-notes exam. You have 75 minutes to complete this exam.

### Long answers

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>Σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perfect</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>8</td>
<td>5</td>
<td>8</td>
<td>3</td>
<td>3</td>
<td>44</td>
</tr>
<tr>
<td>Earned</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Code completion

<table>
<thead>
<tr>
<th></th>
<th>11</th>
<th>12</th>
<th>Σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perfect</td>
<td>15</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>Earned</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total points

______________________
Long Answers

1. Explain how to measure the conditional probabilities of $P(x_j|x_i)$, $P(x_{unseen}|x_i)$, and $P(x_j|x_{unseen})$ using Laplace smoothing. Describe two issues with Laplace smoothing. (5 points)

2. Explain how the Naive Bayes classifier measures the probability of $P(c|F)$ for a class $c$ and a feature set $F = \{f_1, \ldots, f_m\}$. Your answer should include the final form of the Naive Bayes classifier and descriptions of each derivation step to this final form. (6 points)

3. Explain how the Hidden Markov model measures the probability of $P(c_1, \ldots, c_n|o_1, \ldots, o_n)$ for a sequence of observations $o_1, \ldots, o_n$ and a sequence of classes $c_1, \ldots, c_n$. Your answer should include the final form of the Hidden Markov model and descriptions of each derivation step to this final form. (6 points)

4. The following equation shows how to measure $P(y = true|x)$ using logistic regression, where $f$ represents the feature vector of $x$. Explain how to derive this equation (hint: start with the odd of $y$ being $true$). (8 points)

\[
P(y = true|x) = \frac{1}{1 + e^{-\theta \cdot f}}
\]

5. Given a set of classes $\{c_1, \ldots, c_n\}$, explain how to measure $P(c_i|x)$ for each class $c_i$ using multi-class logistic regression. Describe how many hyperplanes are used to perform multi-class classification using logistic regression. What is the advantage of having this many hyperplanes over a fewer number of hyperplanes? (5 points)

6. What are the complexities of exhaustive decoding, top-$k$ decoding, beam-search decoding, and Viterbi decoding, where there are $T$ number of observations and $N$ number of classes? Explain your answers. (8 points)

7. What are the differences between generative and discriminate models? Give an example for each model. (3 points)

8. Explain how to represent string features into a vector. Describe any unique aspect about these feature vectors. (3 points)
Code Completion

Use only methods in the provided interfaces, classes, and Java built-in APIs for all questions. Extra codes that either hurt or do not contribute to the assigned tasks will be penalized.

9. Complete the “segment” method in pseudo-code that populates the list of sequences segmented from the string. (15 points)

```java
interface Sequence {
    /** @return the previous word in this sequence. */
    String getPreviousWord();

    /** @return a copy of this sequence. */
    Sequence copy();

    /** Adds a word and its likelihood to this sequence. */
    String add(String word, double likelihood);
}

/** @return the likelihood of word1 and word2. */
abstract double getLikelihood(String word1, String word2);

/**
 * @param list containing all generated sequences.
 * @param s the string to be segmented.
 * @param beginIndex the begin index of the next word.
 * @param sequence the currently generated sequence.
 */
void segment(List<Sequence> list, String s, int beginIndex, Sequence sequence) {
    // To be filled.
}
```
10. Complete the “decode” method in pseudo-code that takes a Hidden Markov model and a list of observations, and returns a list of tag sequence found by the Viterbi algorithm. (15 points)

```java
interface HiddenMarkov
{
    /** @return the total number of states. */
    int getStateSize();

    /** @return the index'th state. */
    String getState(int index);

    /** @return the overall likelihood given the previous state, the current state, and an observation. */
    double getOverallLikelihood(String previousState, String currentState, String observation);
}

/** *
 * @param hmm Hidden Markov model.
 * @param observations list of observations.
 */
List<String> decode(HiddenMarkov hmm, List<String> observations)
{
    final int N = hmm.getStateSize();
    final int T = observations.size();

    double[][] viterbi = new double[N][T];
    int[][] backpointer = new int[N][T];

    // To be filled.

    return getTagList(viterbi, backpointer, hmm);
}
```