**Lab 6 Requirements**

**A. Functional Requirements**

**Part I: Input Specification.**

What data is going to be provided to the software?

1. What is the meaning of the data?

The data in the FASTA formatted file is the DNA sequence of a region of interest. The sequence is made up of nucleotides (A, T, G, C); occasionally an “N” nucleotide will show up. “N” represents all four of the nucleotides, but is otherwise undefined.

The user will input the coordinates of two exons; an exon is a coding region of a specific gene. These two exons are before and after an unknown exon. The coordinates represent the beginning and end of the exon. If the coordinates are given in reverse order (i.e. 5000-4502), this means that the gene is in the negative strand, which is the reverse complement.

1. In what format will the data be provided?

FASTA format is a text file that starts with a “>” and is followed by a line of information that is not needed for our purposes. Starting at the second line is a string of letters that represent the sequence. Each line is 50-120 nucleotides long. These lines should be combined to create a continuous sequence.

The inputted coordinates should be numbers that correspond to a location in the sequence in the given FASTA formatted file.

**Part II: Processing Instructions.**

What is the software supposed to compute?

The software should be able to find the hidden exon in between the two exons coordinates given by the user. There may be more than one answer as there could be multiple splice sites, which are locations that are cut in order to distinguish exons from introns. Introns are similar to exons, but are non-coding regions of a gene. Also, a gene will have exons and introns in an alternating pattern. The splice sites are defined as the following:

* The beginning (5’ site) of an intron is GT.
* The ending (3’ site) of an intron is AG.

Also, as a tool, we will be providing a Hidden Markov Model (HMM) to calculate the likelihood of the intron and exon locations. An HMM is a statistical model that scores different combinations based on given probabilities. Each nucleotide is assigned a state depending whether it is an Exon, 5’SS, Intron, or 3’SS. Different combinations are called state paths. The probabilities are determined through the data in lab 3 and should be user inputted. The following is an example of the probabilities:

User Inputs:

Intron length 🡪 X = 1/(intron length)

Exon length 🡪 Y = 1/(exon length)

* Transition probabilities

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exon (E)** | **5’SS (5)** | **Intron (I)** | **3’SS (3)** | **End** |
| **Exon** | 1-Y | Y | 0.00 | 0.00 | 0.00 |
| **5’** | 0.00 | 0.00 | 1.00 | 0.00 | 0.00 |
| **Intron** | 0.00 | 0.00 | 1-X | X | \* |
| **3’** | 1.00 | 0.00 | 0.00 | 0.00 | 0.00 |

\*If in the second intron, program should end.

User Inputs:

%A in exon = EA

%T in exon = ET

%G in exon = EG

%C in exon = EC

%A in intron = IA

%T in intron = IT

%G in intron = IG

%C in intron = IC

(exon and intron percentages should add up to 1)

* Emission probabilities

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **A** | **T** | **G** | **C** |
| **Exon** | EA | ET | EG | EC |
| **5’** | 0.00 | 0.00 | 1.00 | 0.00 |
| **Intron** | IA | IT | IG | IC |
| **3’** | 0.00 | 0.00 | 1.00 | 0.00 |

An example of the HMM:

Start 🡪 5’SS 🡪 Intron 🡪 3’SS 🡪 Hidden Exon 🡪 5’SS 🡪 Intron 🡪 Exit

Possible problems:

The program should be able to detect whether the user put in coordinates outside of the given FASTA file (pop up an error message?). Another possible error is the hidden exon goes past the start of the 2nd given exon.

**Part III: Output Specification.**

What should be provided in the output?

1. What information should be output?

* The coordinates of the hidden exon

Calculation:

Likelihood = P path and observed sequence / P all state paths for observed sequence

1. In what format(s) should the information be output?

The output should be in a csv file with the top 5 coordinates, formatted as the following (going from most likely to not likely):

|  |
| --- |
| Hidden Exon Coordinates |
| 5342-5864 |
| 5342-5761 |
| … |

**B. Non-functional Requirements**

What properties shall the software have in regards to:

* Performance (how fast?)

The program should be able to analyze the sequence (<1.5 million nucleotides) in 1 minute or less.

* Usability (how easy to use?)

The software should be intuitive and provide simple instructions.

* Precision and accuracy (how accurate?)

The location of the possible hidden exon sites should be precise. No impossible exon sites should be recorded (i.e. intron splice sites are not GT or AG).

**C. Design Constraints**

* Who will be using the software and what is their skill level?

College level biology student with no experience with software or computer science.

* On what systems will the software be executed?

The program should run on Windows, Mac, or Linux.

* Is there a programming language preference?

Java

**D. Process Constraints**

* Is there a need in early prototypes?

There will be a need of early prototypes in order to find any bugs and/or clarify any problems.

* How much documentation is needed?

The program itself should have short instructions listed on the interface of the program, if possible (ex. “Attach a TXT file in FASTA format”).

* What type of documentation is needed?

See above.

* How much hands-on training is needed?

There should not be any hands-on training.