Interfaz de programación de aplicaciones (API) e Interfaz gráfica (GUI) para algoritmos de busqueda de subcadenas.

Resumen:

Una de la tareas más frecuentes en la biología molecular es el análisis de secuencias (ácido desoxirribonucleico y/o aminoácidos). El fin de dicho análisis es encontrar características, intrínsecas en las secuencias, que permitan determinar si dicha secuencia tiene alguna función biológica o no. Dicho análisis puede ser abordado por un área de la ciencia computacional conocido como Bioinformática.

El presente proyecto modular tiene como objetivo desarrollar una API, utilizando el paradigma orientada a objetos, que permita a los bioinformáticos construir componentes de software para el análisis de secuencias. En el presente trabajo se diseñaran e implementaran algoritmos de string-matching. Además, con el fin de que esta herramienta sea utilizada por profesionales del área de biología molecular (sin conocimientos de programación), se desarrollará una interfaz gráfica de usuario (GUI) que utilice el API antes mencionada.

Antecedentes:

La bioinformática es la ciencia que se encarga de aplicar métodos computacionales para hacer descubrimientos biológicos. Se trata de un campo interdisciplinario que engloba la biología, ciencias de la computación, matemáticas y estadística [1].

Existen diferentes algoritmos computacionales aplicados para el análisis de secuencias entre los que se encuentran alineamiento de secuencias, identificación de patrones (memorias asociativas, redes neuronales), algoritmos de busquedas de subcadenas (string-matching, por sus siglas en ingles), entre otros. Estas técnicas los sirven a diversos propósitos como clasificación de proteínas, identificación de genes, entre otras [1, 3].

Los algoritmos de búsqueda de subcadenas busca encontrar las ocurrencias de un patrón determinado dentro de una secuencia de caracteres, dicha búsqueda puede tener como objetivo encontrar ocurrencias de una cadena específica, lo que se conoce como *concordancia exacta* (exact matching)*,* o un conjunto de cadenas que cumplan con un criterio determinado, posiblemente mediante expresiones regulares, lo que se conoce como *concordancia no-exacta* (inexact matching) [2].

Desde un punto de vista computacional, las secuencias de ADN pueden ser vistas de un modo más abstracto como cadenas de caracteres, éste enfoque nos permite traducir problemas de bioinformática a problemas de string-matching.

Existen diferentes algoritmos para busquedas de subcadenas como Naive matching, Z-algorithm, set matching using keyword trees, suffix trees, suffix arrays, que ayudan a resolver problemas como longest common substring, k-mismatches, entre otros [2].

Por otro lado, existen diferentes API’s para el análisis de secuencias, algunos ejemplos son BioPerl, BioPhyton, BioJava, BioC++, SeqAn entre otros. Para el lenguaje C++, no existe actualmente algun proyecto que utilice el paradigma orientado a objetos, BioC++ dejo de resivir soporte desde el año 2006. Para la mayoria de los proyectos anteriores, el principal objetivo fue el de mantener un mejor desempeño y eficacia de los algoritmos por sobre la facilidad en la construcción de nuevos componentes de software; en este proyecto utilizaremos el paradigma orientado a objetos, que aunque implicaría una disminución en el eficiencia de los algoritmos, nos permitirá desarrollar y construir nuevos componentes de software con mayor facilidad.

Justificación:

Este proyecto se lleva a cabo con la finalidad de integrar un entorno de trabajo en dos niveles diferentes: 1) nivel *intermedio* orientado a desarrolladores de aplicaciones del área de bioinformática; y 2) nivel *alto* una interfaz gráfica de usuario, para especialistas y estudiantes del campo de la biología molecular, para que tenga herramientas (i.e. algoritmos de busquedas de subcadenas) que faciliten su trabajo. Además, dicha interfaz gráfica tendrá la facilidad de ampliarse a través de componentes de software conocidos como plugins.

Objetivos:

1. Objetivo general: Desarrollar una biblioteca de funciones con utilidades especialmente diseñadas para trabajar con secuencias de ADN y aminoácidos así como un entorno de trabajo que con la capacidad de hacer uso de dichas funciones de manera intuitiva y visual.
2. Objetivos específicos: La biblioteca de funciones será escrita con el propósito de que sea utilizada para la implementación de nuevos programas de análisis y alineamiento de secuencias. Por otro lado, el entorno de trabajo visual tiene como objetivo facilitar las tareas cotidianas de aquellos que trabajen usualmente con los algoritmos proporcionados por la biblioteca.

Metodología:

El primer paso es investigar los algoritmos y técnicas de programación que se utilizan en el alineamiento de secuencias y otros problemas similares del campo al que se dedica este proyecto, para luego identificar aquellos que deban codificarse e integrarse en nuestra API.

Una vez identificados los algoritmos que se deben incluir es estudiar cada uno de ellos para entender su complejidad computacional y ser capaz de implementarlos.

El siguiente paso es hacer el diseño y la documentación de la biblioteca, pensando en las interfaces y las clases que serán necesarios, a partir de este punto tendremos en cuenta los detalles de la implementación.

Después de eso se procederá a la codificación, es decir, a la implementación de las interfaces. Con este paso realizado solamente resta publicar la biblioteca en Internet.

En el caso del entorno de trabajo para investigadores el primer paso es el diseño del sistema teniendo en cuenta la comunicación con API y pensando en cuál es la forma de hacer uso de ella de una manera sencilla e intuitiva.

A partir del diseño se llevará a la fase de desarrollo del entorno de trabajo y será publicado también en Internet.

Prototipo:

El primer producto es la biblioteca de funciones junto con su documentación correspondiente. La biblioteca estará escrita en el lenguaje C++ para mantener un diseño intuitivo orientado a objetos y al mismo tiempo conseguir un manejo eficiente de los recursos computacionales disponibles. Se procurará que la biblioteca esté lo más completa posible y los algoritmos sean eficientes tanto en tiempo como en memoria. Además para facilitar la ampliación y mejora de esta API, esta será de código abierto.

El segundo producto es un entorno de trabajo visual basado en el IDE “Eclipse”, el cuál tendrá las funcionalidades necesarias para hacer un uso completo de las funciones en la API antes mencionada de una manera intuitiva y que permita trabajar con distintos formatos de archivos como .gb y .fasta. Este programa será también de código abierto.

Referencias:

1. Baldi, P. & Brunak, S. (1998) *Bioinformatics: the machine learning approach.* MIT Press, Cambridge.
2. Haubold B. & Wiehe, T . (2006). *Introduction to Computational Biology: An Evolutionary Approach*. Alemania: Birkäuser.
3. Mount, D. W. (2004). *Bioinformatics: Sequence and Genome Analysis*. (2da ed.) New York: Cold Spring Harbor Laboratory Press.