Escuela de Computación
Bachillerato en Ingeniería en Computación

TEC | Tecnológico de Costa Rica

Proyecto #1

Alineamiento de Hileras

IC-8022 Introducción a la Biología Molecular Computacional

Tecnológico de Costa Rica, Sede Central Cartago

Escuela de Computación, Ingeniería en Computación

II Semestre 2017

Prof. Ing. Esteban Arias Méndez

La comparación de secuencias es la operación primitiva más importante en la Biología Molecular Computacional que sirve de base para otras manipulaciones de datos más complejas. La operación de alineamiento consiste en obtener un valor de similaridad que permite esta comparación de secuencias.

INTRODUCCIÓN

Propósitos

- Poner en práctica los conocimientos aprendidos sobre alineamiento de secuencias de ADN, ARN o proteínas.
- Poner en práctica la implementación de algoritmos en un lenguaje de programación para su uso y pruebas.

Ayuda útil:

Linux: Si desea, puede realizar una partición de su disco, donde una parte va a tener algún sistema de Linux, esto es: cada vez que arranque la computadora, podrán elegir con cual sistema operativo iniciar (esto es opcional, si instala la máquina virtual, ya esto no hace falta). Este método de doble boot hace que Linux funcione al 100% del rendimiento de la máquina y por ende la mejor opción.

https://www.youtube.com/watch?v=-UwzcoMWljQ

PROGRAMACIÓN

Utilizando algoritmos de Programación Dinámica, su programa debe encontrar alineamientos óptimos (con diversas variantes) entre dos hileras de símbolos de tamaños arbitrarios. Estas hileras podrían ser secuencias de ADN, ARN, proteínas expresadas en su estructura lineal o hileras de cualquier otro origen.

Se trabajará sólo sobre el Sistema Operativo Linux, usando solamente lenguajes C o Python. Se recomienda trabajar su código en módulos, funciones o procedimientos, incluso en varios archivos para facilitar el trabajo. Debe hacer uso del git de la Escuela de Computación para llevar todo el desarrollo de su proyecto: git.ec.tec.ac.cr

Este trabajo es para ser desarrollado por parejas de 2 personas máximo.

Su proyecto deberá trabajar tomando en cuenta las siguientes consideraciones:

1. Proveer una interfaz de uso amigable al usuario: gráfica, textual o web si lo desea; para facilitar el trabajo al usuario final.
2. Ninguna estructura de datos asociada a las hileras debe estar definida estáticamente, sino que serán creadas y destruidas en el momento que se necesiten.
3. Cualquier resultado incluirá información para el usuario sobre los requerimientos de memoria y de tiempo de CPU que se usaron para atender la consulta del usuario.
4. El programa debe manejar correctamente entradas de cualquier tamaño. Por supuesto, si se exceden las capacidades de la máquina en particular, el usuario será informado de esta situación sin que el programa “!se caiga!”.
5. Los pesos utilizados en los alineamientos serán por defecto +1, -1 y -2; pero el usuario los podrá alterar a conveniencia.

6. **Entrada:**
   Dos hileras (secuencias de ADN, ARN, proteínas o texto arbitrario. Estas pueden venir de archivos de texto o tecleadas interactivamente por el usuario (cualquier combinación de estas dos posibilidades debe ser permitida). Estos archivos podrán ser de cualquier tamaño permitido por el sistema operativo.

7. **Salida:**
   Siguiendo las indicaciones del usuario, el programa encontrará el alineamiento óptimo entre las dos hileras (ya sea global, local o semiglobal con sus respectivas variantes). Se debe mostrar el alineamiento alcanzado y el valor óptimo para el mismo. Si el usuario lo solicita, según los comandos siguientes, se debe desplegar la tabla final del alineamiento, mostrando las "flechas" y ruta seleccionada.

8. Comandos especiales:
   a. #ayuda : brindar un texto explicando las operaciones implementadas por su proyecto y un mini manual que explique el uso general del mismo. Si luego del comando #ayuda aparece una palabra clave de la operación o algoritmo, se debe explicar solo esa operación y la forma en que se muestra la salida del mismo, indicando y explicando sus partes.
   b. #tablas : El usuario indica que se debe mostrar las tablas o estructuras de datos internas de trabajo. Con este comando se puede encender o apagar el modo de ejecución de las operaciones, en el cual se debe mostrar o no el procedimiento completo de la operación en la tabla realizada, mostrando las "flechas" y ruta seleccionada.
   c. #listar : mostrará el listado de los algoritmos implementados
   d. #val : mostrar el valor actual de los pesos
   e. #match : mostrar el valor actual de match o establecer un nuevo valor de match si viene como parámetro del comando
   f. #mismatch : mostrar el valor actual de mismatch o establecer un nuevo valor de mismatch si viene como parámetro del comando
   g. #gap : mostrar el valor actual de gap o establecer un nuevo valor de gap si viene como parámetro del comando
   h. #salir : terminar el programa. Se mostrará el total de espacio consumido durante toda la ejecución del programa, el tiempo total de las ejecuciones realizadas en la sesión y terminará la ejecución. Debe mostrar como última salida sus nombres y los datos del curso, periodo, profesor, curso, año, carrera, TEC y periodo.

9. Puede hacer uso de los utilitarios programados para la Tarea 1, documente el uso de los mismos, y si debió hacerles cambios o ajustes.

10. **Algoritmos a implementar:** con una interfaz apropiada el usuario podrá parametrizar el tipo de alineamiento que desea realizar:

    a. Alineamiento global estándar
    b. Alineamiento semiglobal estándar
    c. Alineamiento local estándar
    d. Alineamiento global, semiglobal y local con espacio lineal
    e. Alineamiento global con K-Band, cualquier longitud de hileras
    f. EXTRA: Alineamiento global con K-Band y espacio lineal
    g. EXTRA: Alineamiento global, local y semiglobal con función afín de costo por gap

EVALUACIÓN

1. Rúbrica de evaluación:

    - El proyecto se calificará con los siguientes criterios:

        i. 90% - Funcionamiento correcto del programa según lo solicitado, bajo una estrategia que permita una buena implementación del proyecto.
        ii. 10% - Documentación del trabajo.

2. El proyecto debe resolverse, implementándolo de la mejor manera.

3. Se revisará el código del proyecto, para asegurar que se cumpla con lo solicitado de forma interna y no solo el comportamiento del programa sea como el solicitado.

4. De forma global, se evaluará la presentación del trabajo según los parámetros solicitados, estrategias empleadas y la calidad, la entrega a tiempo del trabajo y la documentación completa correspondiente.

5. Sobre la documentación y presentación:

    a. 2pts - El subject del correo a ser enviado debe ser:
    [BMC] – Proyecto # 1 – Sus Nombres Completos
    b. 2pts - El correo debe contener de forma separada:
        i. los archivos de texto de los códigos fuentes que permiten la solución y funcionalidad del mismo.
        ii. un archivo PDF con la documentación completa

        No envíe archivos ejecutables o binarios.

    c. La documentación en PDF con el nombre de archivo igual al subject del correo enviado. Esta documentación debe tener un apartado, que indique los pasos a seguir, para poder ejecutar el código (librerías a instalar y otros).
        i. 5pts – La documentación debe incluir una portada con los datos completos: TEC, carrera, sede, curso y código, profesor, periodo, fecha de entrega, número de proyecto, título del proyecto, nombres completos con número de

carnet de la pareja de trabajo y un abstract en inglés en la misma portada.

ii. 5pts – La introducción del documento es una descripción breve del trabajo realizado y herramientas usadas. Como mini-marco teórico incluya las referencias a los algoritmos implementados y las herramientas empleadas, así como otras fuentes de consulta utilizadas.

iii. 10pts – Como desarrollo debe explicar, los procedimientos, rutinas, la lógica que utilizo para resolver el problema. indicar los ejemplos de código que ha usado como guía para el desarrollo de los mismos usando las referencias bibliográficas correspondientes. Explicar el uso y funcionamiento.

iv. 20pts – Análisis de resultados, explicando el trabajo implementado, la forma de realización, funcionamiento, general, ejemplos documentados, problemas presentados, estructuras de datos empleadas, algoritmos usados, etc.

v. 10pts – Una sección de conclusiones y/o observaciones sobre el proyecto.

vi. 10pts – En una sección de Apéndices incluya el código fuente documentado del proyecto y su explicación. Explique la estructura del código empleada en su proyecto, módulos, etc.

6. La tarea puede realizarse de forma individual o en parejas de 2 máximo.

7. Se debe entregar en digital a más tardar el Martes 17 de Octubre, antes de la media noche. Debe hacerlo de forma simultánea a los correos siguientes y copiarse usted mismo y su compañero de trabajo. Cada día de atraso serán 15pts menos de la nota de la tarea:

    a. earias@ic-itcr.ac.cr
    b. erohernandez@ic-itcr.ac.cr

8. Cualquier consulta puede hacerla al foro, o personalmente en clase o al correo del profesor con copia al asistente a los correos anteriores.

9. Durante la revisión del proyecto deben estar presentes ambos miembros de la pareja de trabajo, la no presentación les restará 10pts a cada uno de los ausentes.