

Algoritmos y Estructuras de Datos

Tarea 6 – Examen Final

Profesores: José Galaviz y Sergio Rajsbaum
Ayudante: Armando Castañeda

fecha de hoy: 4 de diciembre 2008

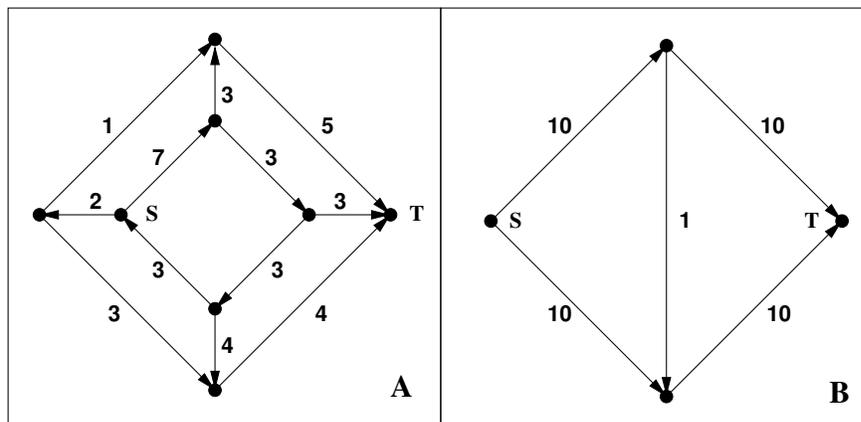
fecha de entrega: 11 de diciembre 2008 – No se aceptan tareas después de esta fecha
— *explica en detalle y con claridad todas tus respuestas* —
— *Tus algoritmos deberán ser lo más eficiente posibles* —
— *explica el funcionamiento de tus algoritmos informalmente, luego escribe el código, y luego demuestra correctez y complejidad.* —

Se permite trabajar en equipos de hasta DOS personas.

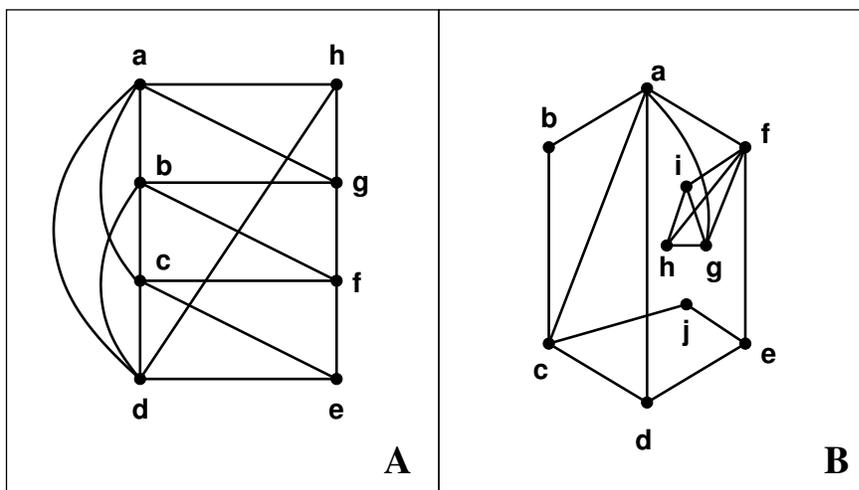
Pero cada uno debe entregar la tarea resuelta por separado, e indicar el nombre de su compañero de equipo.

Tema: Programación Dinámica, Flujo en redes y planaridad.

1. Determina el flujo máximo que se puede establecer entre S y T en las siguientes redes. Las etiquetas al lado de las aristas son las capacidades. ¿Qué ocurrió en la red B ?, ¿qué ocurriría si la capacidad de las aristas en el perímetro del rombo fuera 1,000,000?, ¿eso indica algo acerca del tiempo de ejecución del algoritmo de Ford-Fulkerson? Averigua que ocurre si las capacidades son números irracionales.



2. Utiliza el algoritmo DMP (Demoucron, Malgrange y Pertuiset) para determinar si las siguientes gráficas son planas o no. De acuerdo con el teorema de Kuratowsky, si alguna no lo es, entonces tiene una subgráfica homeomorfa a K_5 o $K_{3,3}$. Si A o B resultan no ser planas, exhibe la subgráfica homemomorfa a K_5 o $K_{3,3}$.



3. Vuelve a jugar *untangle* (<http://www.nonoba.com/chris/untangle>). Ahora que ya conoces un método para hacer un encaje de la gráfica en el plano, ¿llegaste a un nivel mayor? o ¿se te facilitó más la labor?
4. El problema de este ejercicio es el de proponer un algoritmo eficiente, usando la técnica de la programación dinámica sobre intervalos, para resolver un problema fundamental para la biología computacional: **La estructura secundaria del ARN.**

Watson y Crick -los descubridores de la molécula de *ADN*, pilar de la genética- postularon su modelo de la siguiente forma: La molécula de *ADN* está formada por una doble hélice: dos cadenas paralelas de *nucleótidos* (o *bases*) que se “abrazan” entre sí como lo haría el cierre de un pantalón. Esta unión entre hebras es posible gracias a emparejamientos entre bases de la primera con bases de la segunda.

Estos emparejamientos no son arbitrarios, las restricciones para que ocurran son:

- a) Hay cuatro tipos de bases: (**A**)denina, (**C**)itosina, (**G**)uanina y (**T**)imina. La Adenina solo puede emparejarse con la Timina, y la Guanina solo con la Citosina.
- b) El emparejamiento entre bases se llama de esa forma por que una base se empareja con a lo más alguna otra base.

El problema que se pide se relaciona con la molécula de *ARN*, la cual tiene solamente una hebra. El *ARN* es un componente clave de muchos de los procesos que ocurren en el interior de las células, y sigue principios estructurales similares a los del *ADN*. En ella, sin embargo, no hay una segunda hebra sobre la cual pueda replegarse; así que tiende a enrollarse y formar emparejamientos consigo misma, adquiriendo formas interesantes como la que se muestra en la figura 1. El conjunto de parejas (y la forma resultante) formadas por la molécula de *ARN* en este proceso se denomina **estructura secundaria**, y entender la estructura secundaria es esencial para entender el comportamiento de la molécula.

Para nuestros propósitos, una molécula de *ARN* de hebra única puede verse como una secuencia de n símbolos (bases) tomados del alfabeto $\{A, C, G, T\}$. Sea $B = b_1b_2 \dots b_n$ una molécula de *ARN* de hebra única, donde cada $b_i \in \{A, C, G, T\}$. Como una primera aproximación, uno puede modelar su estructura secundaria de la siguiente manera: Como siempre, requerimos que *A* se empareje solo con *T*, y *C* solo con *G*; también requerimos que una base se empareje

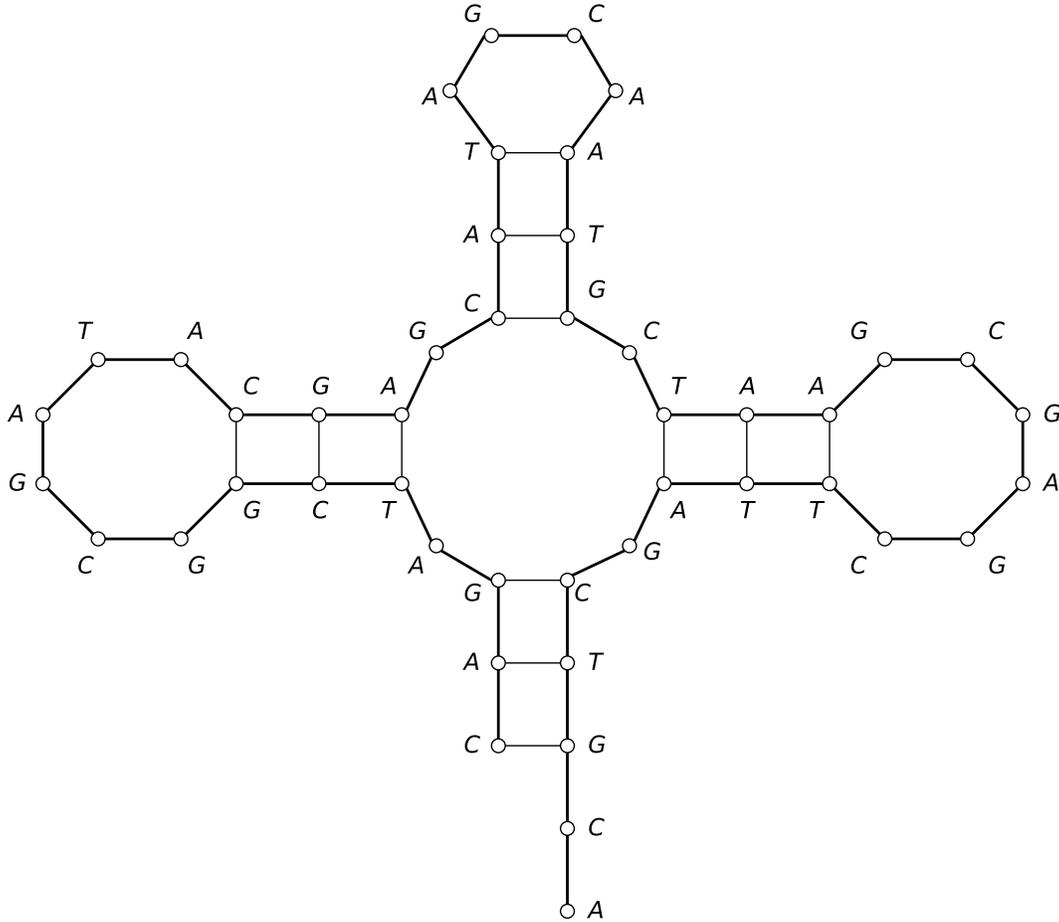


Figura 1: Estructura secundaria de ARN. Líneas gruesas conectan elementos adyacentes de la secuencia; líneas delgadas indican pares de elementos que están emparejados.

con a lo más alguna otra. También ocurre que las estructuras secundarias tienden a ser “libres de nudos”, lo cual formalizamos como una especie de condición de *no-cruce* a continuación.

Concretamente, decimos que una estructura secundaria en B es un conjunto de pares $S = \{(i, j)\}$, donde $i, j \in 1, 2, \dots, n$, y satisface las siguientes condiciones.

- (No hay curvas cerradas). Los extremos de cada par en S están separados por lo menos por 4 bases intermedias; es decir, si $(i, j) \in S$, entonces $i < j - 4$.
- (Correspondencia de bases). Los elementos de cada par en S consisten de algún (A, T) o (C, G) (en cualquier orden).
- (S es un emparejamiento). Ninguna base aparece en más de una pareja.
- (Condición de no-cruce). si (i, j) y (k, l) son dos parejas en S , entonces no podemos tener $i < k < j < l$. (Véase la figura 2 para una ilustración).

Nota que la estructura secundaria de la figura 1 satisface las propiedades (a) a (d).

Estamos interesados en calcular cuales, de entre todas las posibles estructuras secundarias que una molécula puede tener, son las más probables de aparecer en la naturaleza. La hipótesis

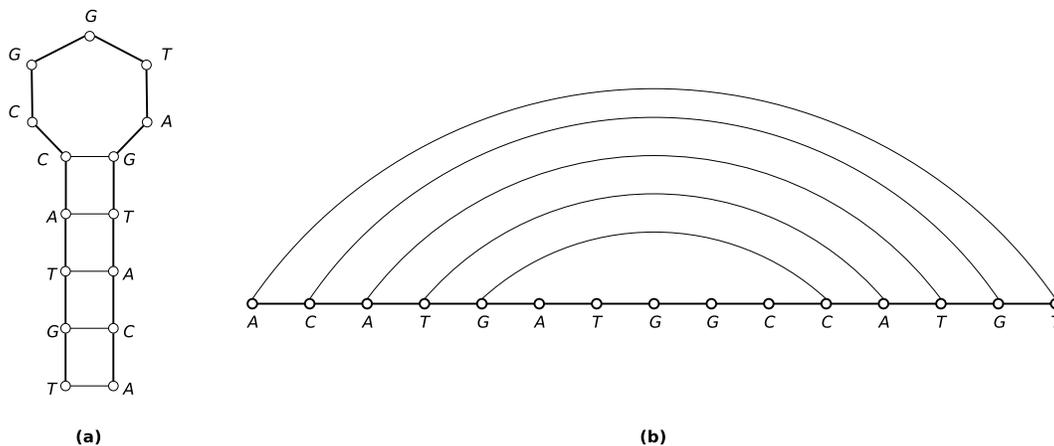


Figura 2: Dos vistas de una estructura secundaria de ARN. En la segunda vista, (b), la molécula se ha enderezado, y los enlaces que unen pares emparejados aparecen como burbujas sobre las moléculas, nota que no se cruzan entre si.

usual es que una molécula de ARN de hebra única formará la estructura secundaria de máxima energía libre. Como modelo sencillo supondremos que la energía libre de una molécula es simplemente proporcional al número de pares de bases emparejadas que contiene.

El planteamiento del problema se reduce entonces a lo siguiente: dada una molécula de ARN de única hebra $B = b_1 b_2 \dots b_n$, determinar una estructura secundaria con un número máximo de pares de bases emparejadas.

Con los antecedentes anteriores:

- a) Explica por que un enfoque de programación dinámica es adecuado para resolver el problema.
- b) Identifica una ecuación recursiva para $OPT(i, j)$, donde $OPT(i, j)$ es el número de enlaces máximo que puede tener una estructura secundaria de la subsecuencia b_i, b_{i+1}, \dots, b_j , cumpliendo evidentemente las restricciones (a) a (d).
- c) Apoyandote en la ecuación anterior, diseña un algoritmo eficiente que resuelva el problema de predecir la estructura secundaria de ARN. Tu algoritmo no solamente debe calcular el número máximo de enlaces que puede tener una estructura secundaria; también debe calcular alguna de esas estructuras óptimas.
- d) Demuestra su corrección y su complejidad.

Tip: Para plantear tu ecuación recursiva, observa los dos casos que pueden suceder al analizar la base j -ésima:

- a) j no se empareja con nadie. Entonces $OPT(i, j) = OPT(i, j - 1)$.
- b) j se empareja con algún t , con $t < j - 4$. Entonces, como ilustra la figura 3, recurrimos a dos subproblemas; $OPT(i, t - 1)$ y $OPT(t + 1, j - 1)$.

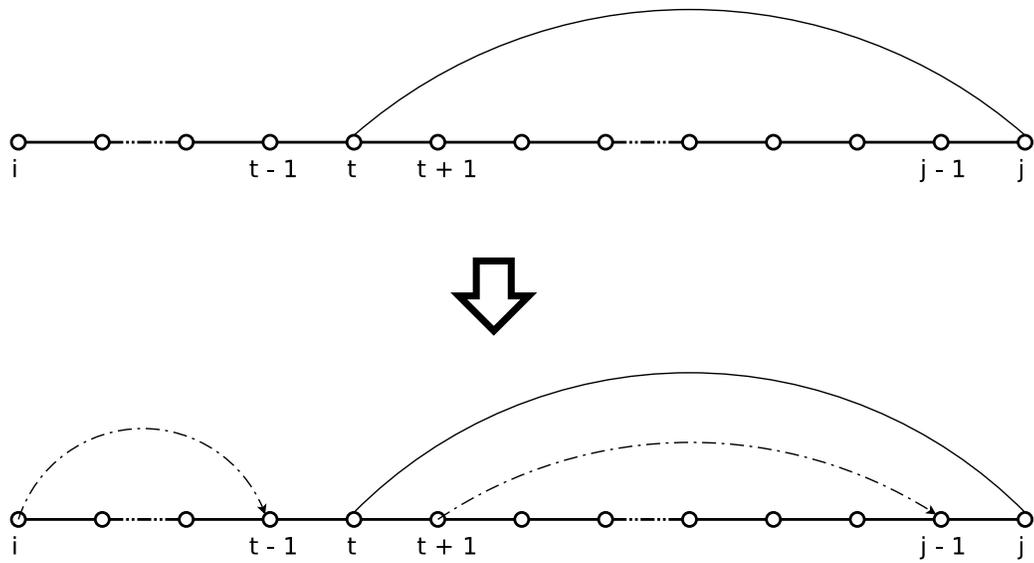


Figura 3: Ilustración de los dos subproblemas que hay que considerar cuando j se empareja con algún t .