

TEORÍA DE ALGORITMOS 1

Guía de ejercicios

Por: ING. VÍCTOR DANIEL PODBEREZSKI

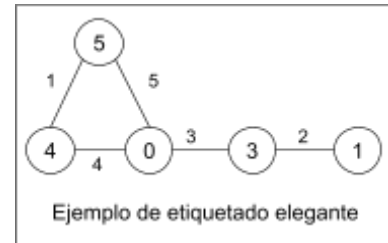
vpodberzski@fi.uba.ar

Búsqueda Exhaustiva

1. Contamos con un conjunto de "n" puntos (x,y) en el plano cartesiano. Un par de puntos es el más cercano si la distancia euclidiana entre ellos es menor a la de cualquier otro par. Resuelva el problema mediante un algoritmo naïve que nos informe cuales son los 3 pares de puntos más cercanos.
2. Un cuadrado mágico de tamaño "n" es una disposición de los números enteros desde 1 a n^2 en una matriz de $n \times n$ que cumple la siguiente condiciones. Cada número aparece solo una vez. La suma de cada fila, columna y diagonal principal da el mismo valor. Proponer un algoritmo por generar y probar que dado un valor "n" calcule, si existe, un cuadrado mágico de ese tamaño.
3. Se encuentran en un río 3 caníbales y 3 vegetarianos. En la orilla hay un bote que permite pasar a dos personas atravesarlo. Los 6 quieren cruzar al otro lado. Sin embargo existe un peligro para los vegetarianos: si en algún momento en alguna de las márgenes hay más caníbales que vegetarianos estos los atacarán. Tener en cuenta que el bote tiene que ser manejado por alguien para regresar a la orilla. Determinar si es posible establecer un orden de cruces en el que puedan lograr su objetivo conservando la integridad física. Explicar cómo construir el grafo de estados del problema. Determinar cómo explorarlo para conseguir la respuesta al problema. Brinde, si existe, la respuesta al problema.
4. Resuelva el problema de las reinas en el tablero de ajedrez mediante backtracking planteado como permutaciones. Brinde el pseudocódigo y determine la cantidad máxima posible de subproblemas a explorar.
5. En un tablero de ajedrez (una cuadrícula de 8×8) se ubica la pieza llamada "caballo" en la esquina superior izquierda. Un caballo tiene una manera peculiar de moverse por el tablero: Dos casillas en dirección horizontal o vertical y después una casilla más en ángulo recto (formando una forma similar a la letra "L"). El caballo se traslada de la casilla inicial a la final sin tocar las intermedias, dado que las "salta". Se quiere determinar si es posible, mover esta pieza de forma sucesiva a través de todas las casillas del tablero,

pasando una sola vez por cada una de ellas, y terminando en la casilla inicial. Plantear la solución mediante backtracking.

6. En la teoría de gráficos, se conoce como etiquetado de vértices a asignarle a cada vértice una etiqueta diferente. De igual manera se puede realizar un etiquetado de ejes. Generalmente el etiquetado se puede representar mediante un número entero. Existen diferentes etiquetados posibles. Trabajaremos con el "etiquetado elegante" (graceful labeling). Dado un grafo $G=(V,E)$ con m ejes se asignará como etiqueta a cada uno de sus vértices un número entre 0 y m . Se computará para cada eje la diferencia absoluta entre las etiquetas de vértices y esa será su etiqueta. Se espera que los ejes queden etiquetados del 1 al m inclusive (y que obviamente cada una sea única). Construya mediante generar y probar un algoritmo que dado un grafo determine un etiquetado elegante (si es posible).



7. Se cuenta con "n" trabajos por realizar y la misma cantidad de contratistas para realizarlos. Ellos son capaces de realizar cualquiera de los trabajos aunque una vez que se comprometen a hacer uno, no podrán realizar el resto. Tenemos un presupuesto de cada trabajo por cada contratista. Queremos asignarlos de forma tal de minimizar el costo del trabajo total. Proponer un algoritmo por branch and bound que resuelva el problema de la asignación.
8. Contamos con un conjunto de "n" de equipamientos que se deben repartir entre un conjunto de "m" equipos de desarrollo. Cada equipo solicita un subconjunto de ellas. Puede ocurrir que un mismo equipamiento lo soliciten varios equipos o incluso que un equipamiento no lo solicite nadie. Queremos determinar si es posible seleccionar un subconjunto de equipos de desarrollo entregándoles a todos ellos todo lo que soliciten y al mismo tiempo que ninguno de los equipamientos quede sin repartir. Resolver el problema mediante backtracking.
9. Un ciclo euleriano en un grafo es un camino que pasa por cada arista una y solo una vez, comenzando por un vértice y finalizando en el mismo. Sea un grafo $G=(V,E)$ se busca generar si es posible un ciclo euleriano. Resolverlo mediante generar y probar.
10. Contamos con un conjunto de "n" actividades entre las que se puede optar por realizar. Cada actividad x tiene una fecha de inicio I_x , una fecha de finalización f_x y un valor v_x que obtenemos por realizarla. Queremos seleccionar el subconjunto de actividades compatibles entre sí que maximice la ganancia a obtener (suma de los valores del subconjunto). Proponer un algoritmo por branch and bound que resuelva el problema.
11. Se cuenta con "n" servidores especializados en renderización de videos para películas animadas en 3d. Los servidores son exactamente iguales. Además contamos con "m"

escenas de películas que se desean procesar. Cada escena tiene una duración determinada. Queremos determinar la asignación de las escenas a los servidores de modo tal de minimizar el tiempo a esperar hasta que la última de las escenas termine de procesarse. Determinar dos metodologías con la que pueda resolver el problema y presente como realizar el proceso.

12. Un granjero debe trasladar un lobo, una cabra y una zanahoria a la otra margen de un río. Cuenta con un bote donde solo entra él y un elemento más. El problema es que no puede quedar solo el lobo y la cabra. Dado que la primera se comería a la segunda. De igual manera, tampoco puede dejar solo a la cabra y la zanahoria. La primera no dudaría en comerse a la segunda. ¿Cómo puede hacer para cruzar? Explicar cómo construir el grafo de estados del problema. Determinar cómo explorarlo para conseguir la respuesta al problema. Brinde, si existe, la respuesta al problema.

Algoritmos Greedy

Para cada ejercicio considere, además de resolver lo solicitado, calcular la complejidad temporal, espacial, justificar la optimalidad y su pertenencia a la metodología greedy. En muchas ocasiones existen varias soluciones alternativas óptimas greedy. Tómese un tiempo para buscarlas.

1. Una ruta tiene un conjunto de bifurcaciones para acceder a diferentes pueblos. El listado (ordenado por nombre del pueblo) contiene el número de kilómetros donde está ubicada cada una. Se desea ubicar la menor cantidad de patrullas policiales (en las bifurcaciones) de tal forma que no haya bifurcaciones con vigilancia a más de 50 km. Proponer un algoritmo que lo resuelva.

Ejemplo (ciudad, Bifurcación): (Castelli, 185), (Gral Guido, 249), (Lezama 156), (Maipu, 270), (Sevigne, 194). Si incluimos un patrullero en la bifurcación de Lezama, cubre además de esta a Castelli y Sevigne. Pero no Gral Guido y Maipú. Se necesitaría en ese caso, ubicar otro. Al agregar otro patrullero en Gral Guido, se cubren todas las ciudades restantes. Con 2 móviles policiales en bifurcaciones se cubren todos los accesos a todas las ciudades con distancia menor a 50km.

2. Conocemos al algoritmo de Kruskal y Prim sobre un grafo conexo y ponderado para obtener su árbol recubridor mínimo. Analice la siguiente estrategia de resolución y determine si corresponde a un algoritmo óptimo. Si lo es, detalle con qué estructuras lo implementaría de la forma más eficiente posible.

Iniciar con el grafo completo

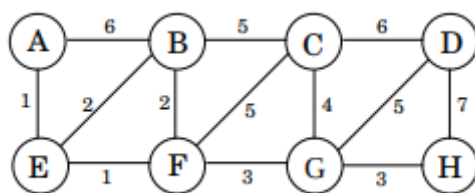
Mientras existan ciclos en el grafo

*Obtener la arista de mayor peso cuya remoción mantenga la conectividad del grafo
Eliminar la arista seleccionada.*

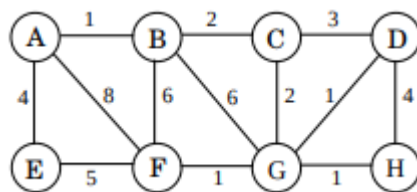
3. Sea C el conjunto de ejes seleccionados por el algoritmo de Dijkstra para un grafo $G=(V,E)$ pesado no dirigido. Probar o dar un contraejemplo las siguientes afirmaciones:
 - a. C corresponde a un árbol recubridor
 - b. C corresponde a un árbol recubridor mínimo.
4. Para la realización del próximo congreso de “charlas motivacionales para el joven de hoy” se contrató un hotel que cuenta con m salas de exposición. Existirán “ n ” oradores. Cada uno solicitó un tiempo de exposición definido por un horario de ingreso y una duración. Los organizadores quieren asignar las salas con un intervalo entre charla y charla de 15 minutos y desean utilizar la menor cantidad de salas posibles. Presentar un algoritmo greedy que resuelve el problema indicando la cantidad de salas a utilizar y la asignación de las charlas. En caso de sobrepasar el máximo de salas disponibles informar. Analice complejidad y optimalidad
5. Una carrera tipo “Ironman” es un triatlón compuesto por 3 instancias: natación (3,86 km de natación), ciclismo (180 km) y carrera a pie (42,2km). Para conocer al ganador se suman los tiempos realizados en cada una de las etapas. Tanto el ciclismo como la carrera a pie se puede realizar en simultáneo con todos los inscriptos. Pero, por una regulación se prohibió que más de 1 persona realice la etapa de nado en el lago en simultáneo. Se conoce el tiempo estimado de cada participante para cada evento. Proponga un orden de salida de tal forma de minimizar el tiempo total de toda la competencia.
6. Las membresías al club de vino “Varietales de cuyo” son de 4 categorías: “Titanio”, “Oro”, “Plata” y “Básico”. Cada socio tiene una membresía. Por mes envían un pack de degustación que llaman “alfa”, “beta”, “gamma” y “epsilon”. Un socio “Titanio” sólo puede recibir un pack “alfa”, Un socio “Oro” puede recibir un pack “alfa” o “beta”, un socio “Plata” sólo puede recibir “alfa” o “gamma”. Finalmente un “Básico” puede recibir cualquier pack. Diseñar una estrategia greedy para resolver el siguiente problema: Sean P_a, P_b, P_g, P_d los packs disponibles de cada tipo y S_t, S_o, S_p, S_b los socios de cada categoría. Informar cómo se puede satisfacer la distribución de packs entre socios (o si no se puede satisfacer).
7. El club de amigos de la república Antillense prepara un ágape en sus instalaciones en la que desea invitar a la máxima cantidad de sus “ n ” socios. Sin embargo por protocolo cada persona invitada debe cumplir un requisito: Sólo puede asistir si conoce a al menos otras 4 personas invitadas. Nos solicita seleccionar el mayor número posible de invitados. Proponga una estrategia greedy óptima para resolver el problema.
8. Considere el problema anterior, pero con la adición de una nueva restricción: Los organizadores desean que cada invitado pueda conocer nuevas personas. Por lo que nos

solicitan que sólo puede asistir si NO conoce al menos otras 4 personas invitadas. Modifique su propuesta para satisfacer esta nueva solución.

9. Indicar si las siguientes afirmaciones son verdaderas o falsas. Justificar la respuesta.
 - a. El algoritmo de Prim genera el mismo resultado que el algoritmo de Kruskal, al calcular el MST de un Grafo.
 - b. El algoritmo de Prim aplicado a un Grafo genera siempre el mismo MST
 - c. El algoritmo de Kruskal aplicado a un Grafo genera siempre el mismo MST
 - d. Si un Grafo posee todas sus aristas distintas, el MST es siempre el mismo sin importar qué algoritmo se utilice
 - e. El MST del siguiente Grafo posee 2 MST



- f. El MST del siguiente Grafo es: {AB, AE, BC, CG, GD, GF, GH}



- g. Dado un grafo G no dirigido con costos en cada arista $c(e)$. Si suponemos que e^* es la arista de menor costo ($c(e^*) < c(e)$ para cada arista de $G / e \neq e^*$), entonces podemos afirmar que cualquier MST de G contendrá a e^* .
10. Un importante museo nacional tiene “ n ” piezas de arte en diferentes bóvedas de seguridad (1 pieza por bóveda inicialmente). Se ha programado una exposición especial que requiere que esas n piezas se reúnan para luego ser transportadas al lugar designado. Para la movilización se ha contratado un seguro que cobra un valor cada vez que una pieza corre riesgo de robo (cuando la bóveda que la contiene se abre). Cada pieza fue tasada con un valor determinado. Cuando se realiza un traslado hay 2 bóvedas involucradas. Se abren tanto la bóveda de origen como la de destino. Por lo tanto se debe pagar al seguro los valores de los que están almacenados en ambas bóvedas. En el traslado se pueden seleccionar cualquier cantidad de las piezas que están en las bóvedas involucradas. Deseamos determinar qué traslados realizar de forma de lograr unificar en una sola bóveda todas las piezas abonando la menor cantidad de plata a la aseguradora. Proponer un algoritmo greedy que solucione el problema (y que sea óptimo).

Ejemplo: Si tengo 4 piezas con sus respectivos valores en 4 bóvedas. A: $p_1=5$ B: $p_2=4$ C: $p_3=2$ D: $p_4=6$ Si comienzo trasladando p_1 a la bóveda B. Deberé abrir la bóveda A y B. Se pagará al seguro $5+4=9$ por el proceso. Si ahora se desea trasladar p_3 a la bóveda B se deberá pagar $2+5+4=11$. El proceso continúa hasta que todas las piezas estén unificadas.

11. Un fabricante de perfumes está intentando crear una nueva fragancia. Y desea que la misma sea del menor costo posible. El perfumista le indicó un listado de ingredientes. Por cada uno de ellos determinó una cantidad mínima (puede ser cero) y una máxima que debe contar en la fórmula final. Cada ingrediente tiene asociado un costo por milímetros cúbicos. Sabiendo que en la presentación final es de X milímetros cúbicos. Presentar una solución utilizando metodología greedy que resuelva el problema.
12. El ajedrez se juega con un tablero cuadrado. La pieza llamada "Rey" puede moverse en cualquiera de los 8 cuadrados adyacentes a su posición actual comiendo cualquier otra pieza que esté en ellos. Contamos con un tablero especial de $n \times m$ cuadrados y una cantidad ilimitada de piezas "Rey". Queremos ubicar la mayor cantidad de reyes sin que estos se puedan comer entre sí. Proponer un algoritmo greedy para resolverlo. Brindar complejidad. Justificar la optimalidad de su propuesta.
13. Un importante club de campo debe cerrar durante un mes sus instalaciones. En ese tiempo tiene que licenciar a la mayoría de sus empleados. Entre ellos a los guardias de seguridad. Cada uno de ellos trabaja 1 vez por mes en un horario de corrido por mes iniciando un día determinado y finalizando unos días después. Habitualmente varios guardias se superponen en algunos momentos del mes. Pero sabemos que siempre hay al menos 1 de ellos en las instalaciones. Lo que nos solicitan es que dejemos a la mínima cantidad de guardias posibles y que aun siempre haya al menos uno custodiando durante el mes. Proponga un algoritmo greedy eficiente que lo resuelve.
14. Un centro de distribución de repuestos ferroviarios se encuentra en un punto de la red de este transporte. Es la encargada de distribuir a demanda los materiales y recursos para las reparaciones que solicitan las diferentes estaciones. Como la red es antigua y está mal mantenida la cantidad de kilos que se puede transferir sobre cada trayecto es variable. Esto para ellos es un problema porque quieren enviar la mayor cantidad posible de material por viaje. Tanto es así que no les importa realizar un camino más largo siempre que eso implique transportar más materiales. Se pueden armar diferentes caminos que unan el centro de distribución con cada estación. Estos estarán conformados por una secuencia de trayectos, cada uno con su propia limitación de kilos que soporta. Llamamos cuello de botella al valor mínimo entre ellos. Construir un algoritmo greedy que permita calcular el camino con el máximo cuello de botella entre el punto de partida y el resto de los puntos.
15. Una jefa de trabajos prácticos (jtp) está a cargo de un grupo de N ayudantes, cada uno de los cuales tiene que trabajar un turno completo durante la semana. Hay muchas actividades asociadas con cada turno (atender el laboratorio, dar clase de consultas, etc.)

pero podemos pensar en un turno como un intervalo de tiempo contiguo. Puede haber más de un turno simultáneamente. La Jtp está tratando de construir un subconjunto de esos N ayudantes para formar un comité de supervisión, que esté en contacto con todos los ayudantes. Un comité de supervisión está en contacto con todos los ayudantes si, cualquiera sea el turno de un ayudante, hay alguien en el comité de supervisión cuyo turno se superpone, aunque sea parcialmente, con el turno de ese ayudante. Ejemplo: $N=3$, ayudante 1 = Lunes de 16 a 20, ayudante 2 = Lunes de 18 a 22, ayudante 3 = Lunes de 21 a 23. En este caso, la solución es {ayudante 2}. Dado un conjunto de ayudantes, diseñar un comité de supervisión lo más pequeño posible, usando una estrategia greedy.

16. Una familia planea un viaje. Tienen pensado salir desde su ciudad a una determinada hora y llegar lo antes que puedan a la ciudad de destino. Para hacer el viaje cuentan con diferentes opciones de recorridos. Diferentes rutas pasan por diferentes pueblos y se ramifican en otras rutas diferentes. Cada trayecto cuenta con una longitud que - teniendo en cuenta el límite de velocidad - les permite establecer el tiempo que les llevará recorrerlo. Por otro lado, conocen el tráfico por hora de cada trayecto. Esto es, el tiempo extra que le insume realizar un trayecto si inician el recorrido en una determinada hora. Construir un algoritmo greedy que determine el mejor camino a realizar en el menor tiempo posible.
17. La república soberana de Antillense tiene su territorio en una isla paradisiaca. Los últimos años sufrió grandes problemas de destrucción de su infraestructura luego del estallido de su volcán, un terremoto, un maremoto y un ataque de mosquitos mutantes radioactivos. Entre sus planes de reconstrucción deben unir mediante carreteras sus " n " poblaciones principales. En un estudio previo conocen el costo de generar caminos desde cada uno de los poblados hacia los otros. Como no tienen fondos deben solicitar préstamos. Desean gastar el menor valor posible, construyendo la menor cantidad de rutas y al menor costo total. Se debe tener en cuenta que únicamente se puede solicitar un préstamo por año para construir una única ruta. Y que la inflación en el país hace que los costos de construcción aumenten 100% . Debe indicar que rutas construir y en qué orden para lograr minimizar el costo total. Presentar una solución greedy que solucione el problema.
18. Una empresa de telecomunicaciones ganó una licitación para construir un conjunto de " r " líneas de comunicación bidireccional para " n " ciudades. Cada línea de comunicación " l " une dos ciudades y de generarla le permite obtener una ganancia mensual G_l . Puede elegir construir cualquier línea a menos que una nueva línea conecte 2 ciudades previamente conectadas que un camino que pase por 1 o más líneas. Nos solicitan nuestro asesoramiento para determinar qué líneas construir maximizando la ganancia obtenida.
19. Para habilitar la realización de un importante evento multideportivo se solicitó como precondition que durante el lapso que dura cada actividad exista junto a la misma personal médico. Conocemos para cada una de las " n " actividades a realizar, el momento

de inicio y final. Como encargados de la inspección nos solicitan que programemos la menor cantidad de inspecciones posibles en las que constatamos que (al menos al momento de la inspección) se cumple la precondition. Una inspección verifica únicamente aquellos eventos que se están llevando a cabo en el momento. Ninguna actividad debe quedar sin inspeccionar. Presentar una solución greedy óptima al problema.

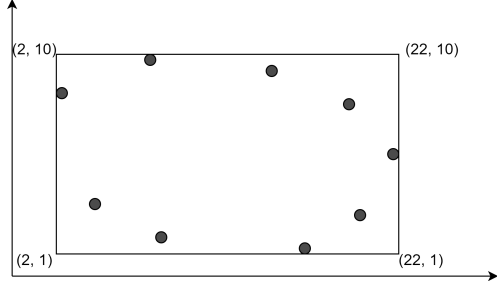
20. Nos proponen el siguiente juego de cartas en el que tenemos que adivinar la carta que tiene un rival. El mazo tiene 1 carta de "1 de Oro", 2 cartas de "2 de Oro" y así hasta 9 cartas de "9 de Oro". El rival mezcla y selecciona una carta. Mediante preguntas que solo se pueden responder por sí o por no tenemos que averiguar en la menor cantidad de consultas cual es la carta. (ejemplos: "La carta es mayor a 4?", "La carta es un "1" o un "3", etc). Proponer un algoritmo greedy que resuelva el problema minimizando la cantidad probable de preguntas a realizar.
21. Sean A y B dos sets de "n" puntos en el plano $p=(x,y)$. Un punto $a_i=(x_i,y_i)$ de A domina a un punto $b_j=(x_j,y_j)$ de B si y sólo si $x_i \geq x_j$ y $y_i \geq y_j$. Un emparejamiento (match) entre un punto a_i de A y uno b_j de B es posible si a_i domina a b_j . Llamamos matching M a un conjunto de emparejamientos $\{(a_1, b_1), (a_2, b_2), \dots, (a_k, b_k)\}$ y su tamaño corresponde a k. Un matching es máximo si no existe otro posible matching con mayor cantidad de puntos. Proponga una estrategia greedy óptima que obtenga el matching máximo para cualquier sets de conjuntos A y B. Procurar realizarlo con la menor complejidad espacial y temporal posible.
22. Un servidor de videojuegos se alquila por horas. El contrato dura un tiempo fijo y permite utilizar en forma exclusiva el mismo por una cantidad continua de horas una vez por semana. Por cada contrato que el dueño del servidor establece, se lleva un monto fijo de dinero. Al dueño del servidor le interesa tener la mayor cantidad de contratos posibles (sin importar la duración en horas de los mismos). El servidor funciona las 24hs. Recibe un conjunto de ofertas de contrato y debe seleccionar cuales aceptar. Cada contrato tiene un día y hora de inicio y un día y hora de fin. Durante ese lapso tendrán la exclusividad del servidor. Ese tiempo contiguo no puede durar más de 1 semana (un contrato podría pedir por ejemplo 3 días completos pero nunca superar la semana).. Y esa fecha se repite todas las semanas. Los contratos aceptados no deben superponerse. Proponer una solución greedy que solucione el problema de forma óptima. Tenga en cuenta que es posible contratos que empiecen al finalizar la semana y terminen horas después del inicio de la misma.
23. Una fotocopidora cada mañana recibe un conjunto de pedidos de clientes. El pedido del cliente i demora t_i en ejecutarse. Para una planificación dada (es decir un cierto orden de las tareas) C_i es la hora en la cual el pedido i termina de ejecutarse (por ejemplo, si el pedido j es el primero que se ejecuta, $C_j = t_j$; si el pedido j se ejecuta a continuación del pedido i, $C_j = C_i + t_j$). Cada cliente tiene un peso w_i que representa su importancia. Se supone que la felicidad de un cliente depende de cuán rápido le entregan el trabajo, por

lo que la empresa decide minimizar el tiempo de demora ponderado = Suma ($w_i * C_i$).
Diseñar un algoritmo greedy eficiente para resolver este problema.

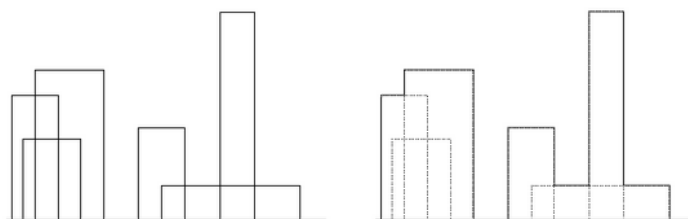
24. Contamos con una impresora central en un centro de cómputos del campus universitario. Entre varios departamentos y laboratorios nos solicitan al inicio de cada mes, la impresión de “n” documentos. Cada uno de ellos tiene una duración determinada y cuenta con una fecha de entrega. Si nos pasamos de esta recibimos un apercibimiento proporcional al retraso más largo del mes. Como a la impresora le falta mantenimiento queremos lograr - siempre que sea posible - tiempo entre los trabajos de impresión. Presentar un algoritmo greedy que dada la lista de tareas proponga la fechas de inicio de publicación minimizando el apercibimiento y dando tiempo entre las tareas siempre que sea posible.

División y conquista

Para cada ejercicio considere, además de resolver lo solicitado, expresar la relación de recurrencia, brindar pseudocódigo y calcular la complejidad temporal, espacial.

- Contamos con una lista ordenada de “n” coordenadas satelitales (latitud-longitud) que conforman un área con forma poligonal convexa. Queremos mostrar ese sector del mapa con el mayor tamaño posible en nuestra pantalla rectangular de la computadora. El programa que muestra el mapa acepta como parámetros 2 coordenadas para construir el rectángulo a mostrar: los correspondientes a los límites inferior izquierdo y superior derecho. Construya un algoritmo eficiente que resuelva el problema con complejidad $O(\log n)$. ¿Existe una solución de fuerza bruta que para “n” pequeños sea más eficiente? ¿para qué tamaño de n esto cambia?
- 
- Un colaborador del laboratorio de “cálculo automatizado S.A” propone un nuevo método de multiplicación de matrices. Utiliza división y conquista partiendo la matrices en bloques de tamaño $n/4 \times n/4$. Y el proceso de combinación de resultados llevara $\Theta(n^2)$. Se muestra vago en indicar la cantidad de subproblemas que creará cada paso. Le indican que este dato es fundamental para decidir si proseguir con esa línea de investigación o no. Actualmente utilizan el algoritmo de Strassen con una complejidad de $O(n \log_2(7))$. Siendo $T(n) = aT(n/4) + \Theta(n^2)$, con a la información a determinar. ¿Cuál es la cantidad de subproblemas más grande que puede tomar la solución para que sea mejor que su algoritmo actual?

3. Se cuenta con un vector de "n" posiciones en el que se encuentran algunos de los primeros "m" números naturales ordenados en forma creciente ($m \geq n$). En el vector no hay números repetidos. Se desea obtener el menor número no incluido. Ejemplo: [1, 2, 3, 4, 5, 8, 9, 11, 12, 13, 14, 20, 22]. Solución: 6. Proponer un algoritmo de tipo división y conquista que resuelva el problema en tiempo inferior a lineal. Expresar su relación de recurrencia y calcular su complejidad temporal.
4. Dado un vector A de "n" números enteros (tanto positivos como negativos) queremos obtener el subvector cuya suma de elementos sea mayor a la suma de cualquier otro subvector en A. Ejemplo: Array: [-2, -5, 6, -2, -3, 1, 5, -6]. Solución: [6, -2, -3, 1, 5]. Resolver el problema de subarreglo de suma máxima por división y conquista.
5. Se realiza un torneo con n jugadores en el cual cada jugador juega con todos los otros n-1. El resultado del partido solo permite la victoria o la derrota. Se cuenta con los resultados almacenados en una matriz. Queremos ordenar los jugadores como P_1, P_2, \dots, P_n tal que P_1 le gana a P_2 , P_2 le gana a P_3 , ..., P_{n-1} le gana a P_n (La relación "le gana a" no es transitiva). Ejemplo: P_1 le gana a P_3 , P_2 le gana a P_1 y P_3 le gana a P_2 . Solución: [P1, P3, P2]. Resolver por división y conquista con una complejidad no mayor a $O(n \log n)$.
6. A raíz de una nueva regulación industrial un fabricante debe rotular cada lote que produce según un valor numérico que lo caracteriza. Cada lote está conformado por "n" piezas. A cada una de ellas se le realiza una medición de volumen. La regulación considera que el lote es válido si más de la mitad de las piezas tienen el mismo volumen. En ese caso el rótulo deberá ser ese valor. De lo contrario el lote se descarta. Actualmente cuenta con el proceso "A" que consiste en para cada pieza del lote contar cuántas de las restantes tienen el mismo volumen. Si alguna de las piezas corresponde al "elemento mayoritario", lo rotula. De lo contrario lo rechaza. Un consultor informático impulsa una solución (proceso "B") que considera la más eficiente: ordenar las piezas por volumen y con ello luego reducir el tiempo de búsqueda del elemento mayoritario. Nos contratan para construir una solución mejor (proceso "C"). Se pide:
 - a. Exprese mediante pseudocódigo el proceso "A".
 - b. Explique si la sugerencia del consultor (proceso "B") realmente puede mejorar el proceso. En caso afirmativo, arme el pseudocódigo que lo ilustre.
 - c. Proponga el proceso "C" como un algoritmo superador mediante división y conquista. Explíquelo detalladamente y brinde pseudocódigo.
7. Para la elaboración de un juego se desea construir un cielo nocturno de una ciudad donde se vea el contorno de los edificios en el horizonte. Cada edificio "ei" está representado por rectángulos mediante la tripla (izquierda, altura, derecha). Dónde "izquierda" corresponde a la coordenada x menor, "derecha" la coordenada x mayor y altura la coordenada y. Todos los edificios inician en la coordenada 0 de y. Se



cuenta con una lista de N edificios que llegan sin un criterio de orden específico. Se desea emitir como resultado el contorno representado como una lista de coordenadas "x" y sus alturas. Tenga en cuenta el siguiente ejemplo: Lista de edificios: (1, 11, 5), (2, 6, 7), (3, 13, 9), (12, 7, 16), (14, 3, 25), (19,18,22). Contorno: (1,11),(3,13),(9,0),(12,7),(16,3),(19,18),(22,3),(25,0). Presentar un algoritmo utilizando división y conquista que dado el listado de edificios retorne como resultado el contorno de la ciudad.

8. Esta peculiar empresa se dedica a cubrir patios cuadrados de $n \times n$ metros ("n" es un número entero potencia de 2 y mayor o igual a 2). Cuenta con baldosas especiales que tienen forma en L (como se muestra en celeste en la imagen). Las baldosas no se pueden cortar. Todo patio cuenta con un único sumidero de agua de lluvia. Ocupa 1x1 metro y su ubicación depende del patio (Se muestra en la figura de ejemplo como un cuadrado negro). Nos piden que, dado un patio con un valor "n" y una ubicación del sumidero en una posición x,y desde la punta superior izquierda, determinemos cómo ubicar las baldosas. Presentar un algoritmo que lo resuelva utilizando división y conquista.

(0,0)	(0,1)	(0,2)	(0,3)
(1,0)	(1,1)	(1,2)	(1,3)
(2,0)	(2,1)	(2,2)	(2,3)
(3,0)	(3,1)	(3,2)	(3,3)

9. Se realiza un experimento de conductividad de un nuevo material en aleación con otro. Se formaron muestras numeradas de 1 a n. A mayor número, mayor concentración del nuevo material. Además se realizaron "n" mediciones a diferentes temperaturas de conductividad para cada muestra. Los resultados fueron expresados en una matriz M de $n \times n$. Se observa que un mismo material cuanto mayor temperatura tiene mayor conductividad. Además, cuanto mayor concentración a la misma temperatura, también mayor conductividad. En conclusión podemos, al analizar la matriz, ver dos progresiones. Cada fila tiene números ordenados de forma creciente y cada columna tiene números ordenados de forma creciente. Dada la matriz M, los experimentadores quieren encontrar en qué posición se encuentra un determinado número. Proponga una solución utilizando división y conquista.
10. Todos los años la asociación de un importante deporte individual profesional realiza una preclasificación de los n jugadores que terminaron en las mejores posiciones del ranking para un evento exclusivo. En la tarjeta de invitación adjuntan el número de posición en la que está actualmente y a cuantos rivales invitados superó en el ranking comparado el año pasado. Contamos con un listado que tiene el nombre del jugador y la posición del ranking del año pasado ordenado por el ranking actual. Ejemplo: LISTA: A,3 | B,4 | C,2 | D,8 | E,6 | F,5. Se puede ver que el jugador "A" superó al jugador "C". El jugador "B" superó al jugador "C". El jugador "C" no superó a ninguno de los invitados. Etc. Proponer una solución utilizando la metodología de división y conquista.

11. Una agencia gubernamental tiene un conjunto de "n" agentes dispersos por el país. Para una misión urgente requiere utilizar dos de ellos. Cada agente tiene una ubicación (x,y). Se dispone de un helicóptero para buscarlos. Generar una solución por división y conquista que indique cuáles son los 2 agentes más cercanos, cuál es su distancia y dónde debería ir el helicóptero a buscarlo.

12. Para determinar si un número es primo existen varios algoritmos propuestos. Entre ellos el test de Fermat. Este es un algoritmo randomizado que opera de la siguiente manera: Dado un número entero "n", seleccionar de forma aleatoria un número entero "a" coprimo a n. Calcular a^{n-1} módulo n. Si el resultado es diferente a 1, entonces el número "n" es compuesto. La parte central de esta operatoria es la potenciación. Podríamos algorítmicamente realizarla de la siguiente manera:

pot = 1

Desde i=1 a n-1

pot = pot * a

En este caso se realizan $O(n)$ multiplicaciones. Proponga un método usando división y conquista que resuelva la potenciación con menor complejidad temporal.

13. Dado "L" un listado ordenado de "n" elementos y un elemento "e" determinado. Deseamos conocer la cantidad total de veces que "e" se encuentra en "L". Podemos hacerlo en tiempo $O(n)$ por fuerza bruta. Presentar una solución utilizando división y conquista que mejore esta complejidad.

14. Un conjunto de "n" personas votó de forma anónima entre un conjunto de "o" opciones (con $0 < o \leq n$). El resultado de la votación lo tenemos en un vector de n posiciones ordenado por opción seleccionada. Queremos determinar cuántos votos tuvo cada una de las opciones. Podemos hacerlo simplemente recorriendo el vector en $O(n)$. Sin embargo, utilizando división y conquista se puede lograr en un tiempo inferior. Presentar y analizar una solución utilizando división y conquista que logre lo solicitado.

15. Una encuesta de internet pidió a personas que ordenen un conjunto de "n" películas comenzando por las que más les gusta a las que menos. Con los resultados quieren encontrar quienes comparten gustos entre sí. Nos solicitan generar un algoritmo, que basándose en el concepto de inversión, compare entre pares de personas y determine qué tan compatibles o incompatibles son. Proponer un algoritmo utilizando división y conquista que lo resuelva.

16. Dentro de un país existen dos colonias subacuáticas cada una de ellas con "n" habitantes. Cada habitante tiene su documento de identidad único identificado por un número. Para una tarea especial se decidió seleccionar a aquella persona que vive en alguna de las colonias cuyo número de documento corresponda a la mediana de todos los números de documento presentes en ellas. Por una cuestión de protocolo no nos

quieren dar los listados completos de documentos. Solo nos responden de cada colonia ante la consulta "Cual es el documento en la posición X de todos los habitantes de la isla ordenados de mayor a menor". Utilizando esto, proponer un algoritmo utilizando división y conquista que resuelva el problema con la menor cantidad posibles de consultas. Analizar complejidad espacial y temporal.

17. Se cuenta con un vector V de " n " elementos. Este vector visto de forma circular está ordenado. Pero no necesariamente en la posición inicial se encuentra el elemento más pequeño. Deseamos conocer la cantidad total de rotaciones que presenta " V ". Ejemplo: $V = [6, 7, 9, 2, 4, 5]$ se encuentra rotado en 3 posiciones. Podemos hacerlo en tiempo $O(n)$ por fuerza bruta. Presentar una solución utilizando división y conquista que mejore esta complejidad.
18. Sea S un conjunto de n puntos en el plano. Definimos que un punto $p=(x,y) \in S$ es un punto máximo en S si no existe otro punto $p'=(x',y') \in S$ tal que $x \leq x'$ e $y \leq y'$. Queremos obtener todos los puntos de S que sean máximos. Plantear un algoritmo de división y conquista para que lo resuelva. Analizar su complejidad computacional. (HINT: De ser requerido puede realizar un preorden del conjunto)

Programación dinámica

Para cada ejercicio considere, además de resolver lo solicitado, definir el problema, expresar la relación de recurrencia, brindar pseudocódigo y calcular la complejidad temporal, espacial.

1. Para una inversión inmobiliaria un grupo inversor desea desarrollar un barrio privado paralelo a la una ruta. Con ese motivo realizaron una evaluación de los diferentes terrenos en un trayecto de la misma. Diferentes inversores participarán, pero a condición de comprar algún terreno en particular. El grupo inversor determinó para cada propiedad su evaluación de ganancia. El mismo surge como la suma de inversiones ofrecida por el terreno menos el costo de compra. Debemos recomendar que terrenos contiguos comprar para que maximicen sus ganancias. Ejemplo: $S = [-2, 3, -3, 4, -1, 2]$. La mayor ganancia es de 5, comprando los terrenos de valor $[4, -1, 2]$. Solucionar el problema mediante un algoritmo de programación dinámica.
2. Dado un Grafo dirigido, acíclico $G(V, E)$ con pesos en sus aristas y dos vértices " s " y " t "; queremos encontrar el camino de mayor peso que exista entre " s " y " t ". Resolver mediante programación dinámica.
3. El dueño de una empresa desea organizar una fiesta de la misma. Cómo quiere que la fiesta sea lo más tranquila posible, no quiere que asistan un empleado y su jefe directo,

pero tampoco quiere que se salteen más de 1 nivel jerárquico. Charles le pidió a su encargado que le ponga un rating de convivencia (R_i) a cada empleado y con dicha información más un organigrama de la compañía, le pidió a un especialista en informática que diseñe un algoritmo para obtener el listado de los empleados que deberá invitar a la fiesta. Teniendo en cuenta que: $R_i > 0$, que Charles no cuenta para el armado del algoritmo y que la estructura jerárquica es en forma de árbol. Se pide resolver mediante programación dinámica.

4. La organización de una feria internacional tiene que programar diferentes eventos a realizar en su escenario principal. Para ello pueden elegir, en los diferentes días del evento, entre algunos de los siguientes rubros: un cantante, una compañía de danza, un show de variedades o un humorista. Disponen de una oferta de cada tipo para cada día y la posible ganancia por venta de entradas. Existen ciertas restricciones que se aplican. No se pueden repetir 2 días seguidos el mismo rubro. Además por el tiempo de preparación un día después de un cantante solo puede presentarse un humorista. Plantear la resolución mediante programación dinámica.
5. Una agencia de inteligencia ha conseguido interceptar algunos mensajes encriptados de una agencia rival. Mediante espionaje logran saber algunas cosas para descryptar los mensajes. Por ejemplo, que para dificultar la tarea de los criptoanalistas los mensajes enviados no contienen espacios. Se ha organizado un grupo de trabajo para generar un algoritmo para quebrar la seguridad. El trabajo se dividió en diferentes partes. A usted le toca, dado un string "descryptado" y sin espacios, determinar si lo que se lee corresponde a un posible mensaje en idioma castellano. El proceso debe ser rápido dado que se debe utilizar muchas veces. Cuenta con un diccionario de "n" palabras. y con una cadena de texto con el posible mensaje.

Ejemplo: si el diccionario es "peso", "pesado", "oso", "soso", "pesa", "dote", "a", "te", para la cadena de texto "osopesadotepesa". Existe un posible mensaje con las palabras "oso", "pesado", "te", "pesa". Para la cadena de texto "ososoapesadote". No existe un posible mensaje

- a. Construir una solución que informe si la cadena de entrada es un posible texto utilizando programación dinámica.
 - b. Existe la posibilidad de que una cadena de texto puede corresponder a más de un mensaje. Modifique su solución para que se informen todos los posibles mensajes. Determine el impacto en las complejidades en el nuevo algoritmo.
6. Una empresa de transporte aéreo cuenta con un avión que cubre una ruta que une dos ciudades. Se especializa en llevar maquinaria pesada. Puede realizar un vuelo diario y trasladar una cantidad de peso determinada. Por cada kilo transportado cobra una tarifa. Por día recibe un pedido de transporte que puede aceptar o rechazar en su totalidad (no se pueden fraccionar). Si rechaza un pedido, lo pierde y no lo puede realizar otro día. Existe una regulación especial que indica que la carga máxima a transportar

disminuye según una fórmula por día. Se restablece a su máximo únicamente si el avión pasa por revisión e inspección. Este proceso insume 3 días seguidos. El gerente de la empresa cuenta con una planilla que detalla los envíos solicitados en el próximo mes y la tabla de peso máximo por día pasado desde la última revisión. El avión llegó el día anterior de la revisión. Construir un algoritmo que permita maximizar las ganancias.

7. Contamos con una secuencia de "n" matrices (A_1, A_2, \dots, A_n) a ser multiplicadas. Queremos computar el siguiente producto $A_1 * A_2 * \dots * A_n$. Considere utilizar la multiplicación estándar de matrices. si tengo la matriz A de tiene una dimensión de $p \times q$, y la matriz B tiene una dimensión de $q \times r$, realizar $C = A * B$ requiere $p \times q \times r$ multiplicaciones escalares. Sabemos que podemos elegir diferentes modos de multiplicar las matrices. La elección del orden de la multiplicación tiene un impacto importante en el costo total del cálculo. Por ejemplo, para 4 matrices $A_1 * A_2 * A_3 * A_4$ podemos multiplicar de las siguientes maneras:

$$\begin{aligned} (A_1 * (A_2 * (A_3 * A_4))) &= \\ (A_1 * ((A_2 * A_3) * A_4)) &= \\ ((A_1 * A_2) * (A_3 * A_4)) &= \\ ((A_1 * (A_2 * A_3)) * A_4) &= \\ (((A_1 * A_2) * A_3) * A_4) &= \end{aligned}$$

Plantee mediante programación dinámica un mecanismo de seleccionar el orden de multiplicación de forma de minimizar la cantidad de multiplicaciones escalares a realizar.

8. Contamos con una carretera de longitud M km que tiene distribuidos varios carteles publicitarios. Cada cartel "i" está ubicado en un "ki" kilómetro determinado (pueden ubicarse en cualquier posición o fracción de kilómetro) y a quien lo utiliza le asegura una ganancia "gi". Por una regulación no se puede contratar más de 1 cartel a 5km de otros. Queremos determinar qué carteles conviene contratar de tal forma de maximizar la ganancia a obtener.
9. Un ramal ferroviaria pone en concesión los patios de comida en todas las estaciones. Son en total "n" estaciones. Por cada estación se cuenta con el promedio de facturación de los últimos 5 años. Por normativa antimonopólica existe como limitante que ninguna empresa puede explotar 3 o más estaciones contiguas. Pero ,no existe una cantidad máxima de estaciones a explotar. Un oferente nos solicita que le digamos cuales son las estaciones que le conviene obtener para maximizar sus ganancias. Plantee la solución mediante programación dinámica.
10. Un amigo le propone un juego de cartas a otro. Las reglas son sencillas. Se mezcla un mazo de "n" cartas y se preparan en una fila boca arriba. El juego es por turnos. Un primer participante debe seleccionar una de las cartas de los extremos y se la queda. A continuación el contrincante debe seleccionar entre las cartas restantes una de los extremos. Repiten el procedimiento hasta que no queden cartas en la fila. Gana el

jugador cuya suma de los valores de las cartas que eligió suman más. Los dos son buenos jugadores y muy competitivos. Es posible que tengan sus métodos para intentar conseguir el mejor puntaje. Proponer un método utilizando programación dinámica para obtener el mayor puntaje posible siendo el jugador que empieza el juego. Suponer que el adversario usa el mismo método e intenta obtener para sí el mayor valor posible.

11. La detección de ciclos negativos tiene una variedad de aplicaciones en varios campos. Por ejemplo en el diseño de circuitos electrónicos VLSI, se requiere aislar los bucles de retroalimentación negativa. Estos corresponden a ciclos de costo negativo en el grafo de ganancia del amplificador del circuito. Tomando como entrada de nuestro problema un grafo ponderado con valores enteros (positivos y/o negativos) dirigido donde un nodo corresponde al punto de partida, queremos conocer si existe al menos un ciclo negativo y en caso afirmativo mostrarlo en pantalla. Proponer una solución al problema que utiliza programación dinámica.
12. Un dueño de un camión de transporte se comprometió a trasladar una carga de una ciudad A a la ciudad B. Para realizar el recorrido puede optar por diferentes caminos que pasan por diferentes ciudades intermedias. Nos acerca un mapa donde para cada tramo que une las diferentes ciudades indica el costo de realizar el mismo. Vemos que algunos costos son positivos (combustible, peaje, etc) y otros negativos o cero (yendo por esos tramos puede ganar unos pesos haciendo algunos encargos "particulares"). Nos solicita que le informemos cuál sería el recorrido óptimo para minimizar el gasto total del viaje. Presentar un algoritmo polinomial utilizando programación dinámica que lo resuelva.
13. En un grafo se conoce como conjunto independiente a un subconjunto de vértices del mismo tal que ninguno sea adyacente a otro. Es decir, es un conjunto X de vértices tal que para ningún par de ellos existe alguna arista que los conecte. No se conoce un algoritmo eficiente que resuelva el problema. Sin embargo, para algunos casos especiales de grafos si es posible. Considerar el siguiente caso: Un grafo es un camino si se pueden escribir sus nodos como una sucesión V_1, V_2, \dots, V_n donde cada nodo V_i tiene un eje únicamente con V_{i-1} y V_{i+1} . (Excepto en los extremos donde solo tienen un eje con el siguiente o el anterior). Considerar que cada nodo tiene un peso entero positivo. Construir un algoritmo utilizando programación dinámica que encuentre el set independiente de mayor peso.
14. Para un nuevo satélite a poner en órbita una empresa privada puede optar por incluir diversos sensores a bordo (por ejemplo: variación de temperatura, humedad en tierra, caudal de ríos, etc). Cada uno de ellos tiene un peso " p_i " y una ganancia " g_i " calculado por su uso durante la vida útil del satélite. Si bien les gustaría incluir todos, el satélite tiene una carga máxima P que puede llevar. Nos piden que generemos un algoritmo (utilizando programación dinámica) para resolver el problema. Indique si su solución es polinomial.

15. Sea $G=(V,E)$ un grafo dirigido. No se conoce un algoritmo polinomial para encontrar el camino más largo simple (El largo del camino corresponde a la cantidad de nodos por los que pasa). Sin embargo, para ciertos casos se puede resolver en forma eficiente. Consideremos que G corresponde a un grafo ordenado. En este caso se pueden disponer ordenados los nodos de forma que: i) Desde cada nodo pueden existir ejes salientes a nodos con mayor "índice" (posteriores en su ordenamiento) pero no a nodos de menor índice. y ii) Cada nodo excepto el último tiene ejes salientes. Proponer un algoritmo eficiente que resuelva este problema.
16. Luego de aprender programación dinámica un estudiante en una charla de café le explica a un amigo que atiende un negocio el algoritmo de cambio mínimo de monedas. Con eso, afirma, podrá optimizar el despacho de mercadería. El comerciante despacha pedidos de cierto producto que viene en empaques prearmados de distintas cantidades o sueltos en unidades. Sin embargo, el amigo le replica que su algoritmo es poco realista. Supone que uno tiene una cantidad ilimitada de empaques de cada presentación. Luego de pensar unos momentos, el estudiante llega a una variante de este problema teniendo en cuenta esta restricción. ¿Podría usted detallar cuál es esta solución?
17. Se conoce como "Longest increasing subsequences" al problema de, dado un vector de numérico, encontrar la subsecuencia más larga de números (no necesariamente consecutivos) donde cada elemento sea mayor a los anteriores. Ejemplo: En la lista $\rightarrow 2, 1, 4, 2, 3, 9, 4, 6, 5, 4, 7$. Podemos ver que la subsecuencia más larga es de longitud 6 y corresponde a la siguiente "1, 2, 3, 4, 6, 7". Resolver el problema mediante programación dinámica.
18. Se define el problema 2-Partition de la siguiente manera: Se cuenta con un conjunto de "n" elementos. Cada uno de ellos tiene un valor asociado. Se desea separar los elementos en 2 conjuntos que cumplan con: La suma de los valores de cada conjunto sea igual entre ellos. No se conoce un algoritmo eficiente para resolver este problema. Sin embargo - al igual que otros problemas puede ser resuelto utilizando programación dinámica de forma pseudopolinomial. Presente una solución al problema utilizando dicha metodología.
19. Una variante del problema de la mochila corresponde a la posibilidad de incluir una cantidad ilimitada de cada uno de los elementos disponibles. En ese caso, tenemos una mochila de tamaño "k" y un conjunto de "n" elementos con stock ilimitado. Cada elemento tiene un peso y un costo. Queremos seleccionar el subconjunto de elementos que maximice la ganancia de la mochila sin superar su capacidad. Solucione el problema utilizando programación dinámica.
20. Dada una matriz booleana de $n \times m$ queremos encontrar la mayor submatriz cuadrada cuyos elementos sean sólo "true". Diseñar un algoritmo mediante programación dinámica para resolverlo.

21. Recordemos el problema de selección de intervalos. Partimos de un recurso y un conjunto de intervalos. Cada intervalo cuenta con una fecha de inicio y otra de finalización. Queremos maximizar la cantidad de intervalos a seleccionar siempre que los mismos no se superpongan entre sí. Conocemos un algoritmo greedy que los resuelve de forma óptima. Queremos una variante donde la respuesta al problema corresponda a maximizar el tiempo de ocupación del recurso. Proponer un algoritmo utilizando programación dinámica que lo solucione.
22. Una empresa constructora planea iniciar un nuevo proyecto complejo. ha realizado un estudio de cada una de las tareas a realizar. Para cada tarea se conoce la duración de la misma y cuales son las tareas anteriores requeridas para comenzar a realizarla. Se pueden realizar en paralelo tantas tareas como podamos. Como requisito para iniciar se debe contratar un seguro por accidentes. Este seguro debe cubrir todo el ciclo de desarrollo. Es por eso que nos solicitan calcular (sin tener en cuenta posibles retrasos) cual es el tiempo máximo que nos puede demandar la conclusión del mismo. Ejemplo: Si tengo 4 tareas. A = (23 días, Requiere: C y D). B=(5 días, sin requerimientos). C = (5 días, requiere D y B), D=(10 días, sin requerimientos). El tiempo de desarrollo máximo es $38 = 10$ (completar B y D) + 5 (completar C) + 23 (completar A). Resolver el problema utilizando programación dinámica.
23. Se proyectó la construcción de una línea de tensión eléctrica. En su trayecto pasa cerca de "n" ciudades que tienen diferentes demandas de consumo eléctrico. Para la interconexión entre la línea y una central se debe construir un nexo. Por cuestiones regulatorias y constructivas no se pueden construir nexos a menos de "x" kilómetros entre sí. Contamos con el listado de las ciudades. Para cada ciudad nos informan su población y la ubicación en la línea del nexo a construir. Mediante programación dinámica proponga una solución que permita seleccionar qué ciudades conectar para maximizar la cantidad total de población cubierta por esta línea.
24. El dueño de una cosechadora está teniendo una demanda muy elevada en los próximos 7 meses. Desde "n" campos lo quieren contratar para que preste sus servicios. Lamentablemente no puede hacer todos los contratos puesto que varios de ellos se superponen en sus tiempos. Cuenta con un listado de los pedidos donde para cada uno de ellos se consigna: fecha de inicio, fecha de finalización, monto a ganar por realizarlo. Su idea es seleccionar la mayor cantidad de trabajos posibles. Mostrarle que esta solución puede no ser la óptima. Proponer una solución utilizando programación dinámica que nos otorgue el resultado óptimo (que trabajos elegir y cuanto se puede ganar).
25. Una empresa que realiza ciencia de datos debe realizar en las próximas "n" semanas procesos y cálculos intensivos. Para eso debe contratar tiempo de cómputo en la nube. Realizando una estimación conocen cuantas horas de cómputo necesitaran para cada una de las semanas. Por otro lado, luego de negociar con los principales proveedores tienen 2 opciones que puede combinar a gusto. Opción 1: Contratar a la empresa

“Arganzón” por semana. En esa semana se cobra proporcional al tiempo de cómputo según un parámetro “r” (horas computo x r). Opción 2: Contratar a la empresa “Fuddle” por un lapso de 5 semanas contiguas. Durante el lapso contratado se paga una tarifa fija de “c”. Proponer una solución utilizando programación dinámica que nos indique la secuencia de elecciones a realizar para minimizar el costo total de cómputo.

Redes de Flujo

Para cada ejercicio considere, explicar el proceso de creación de la red de flujo, el proceso de resolver el problema mediante la red de flujo y la interpretación del resultado. Presentar pseudocódigo y realizar análisis de complejidad. Brinde uno o varios ejemplos representativos del problema.

1. La sala de guardia de un hospital tiene que tener al menos un médico en todos los feriados y en los fines de semana largos de feriados. Cada profesional indica sus posibilidades: por ejemplo alguien puede estar de guardia en cualquier momento del fin de semana largo del 9 de julio (p. ej. disponibilidad de A para el 9 de julio = (Jueves 9/7, Viernes 10/7, Sábado 11/7, Domingo 12/7)), también puede suceder que alguien pueda sólo en parte (por ejemplo, disponibilidad de B para 9 de julio = (Jueves 9/7, Sábado 11/7, Domingo 12/7)). Aunque los profesionales tengan múltiples posibilidades, a cada uno se lo puede convocar para un solo día (se puede disponer de B sólo en uno de los tres días que indicó). Para ayudar a la sala de guardia a planificar cómo se cubren los feriados durante todo el año debemos resolver el problema de las guardias: Existen k períodos de feriados (por ejemplo, 9 de julio es un período de jueves 9/7 a domingo 12/7, en 2019 Día del Trabajador fue un período de 1 día: miércoles 1 de mayo, etc.). D_j es el conjunto de fechas que se incluyen en el período de feriado j -ésimo. Todos los días feriados son los que resultan de la unión de todos los D_j . Hay n médicos y cada médico i tiene asociado un conjunto S_i de días feriados en los que puede trabajar (por ejemplo B tiene asociado los días Jueves 9/7, Sábado 11/7, Domingo 12/7, entre otros).
Proponer un algoritmo polinomial (usando flujo en redes) que toma esta información y devuelve qué profesional se asigna a cada día feriado (o informa que no es posible resolver el problema) sujeto a las restricciones:
 - Ningún profesional trabajará más de F días feriados (F es un dato), y sólo en días en los que haya informado su disponibilidad.
 - A ningún profesional se le asignará más de un feriado dentro de cada período D_j .
2. Definimos el Problema de la Evacuación de la siguiente manera: Se tiene un grafo dirigido $G = (V, E)$ que describe una red de caminos. Tenemos una colección de nodos $X \subset V$ que son los nodos poblados (ciudades) y otra colección de nodos $S \subset V$ que son los nodos de refugio (supondremos que X y S son disjuntos). En caso de una emergencia

queremos poder definir un conjunto de rutas de evacuación de los nodos poblados a los refugios. Un conjunto de rutas de evacuación es un conjunto de caminos en G tales que (i) cada nodo en X es el origen de un camino, (ii) el último nodo de cada camino es un refugio (está en S), y (iii) los caminos no comparten aristas entre sí. Se pide: dados G , X y S , mostrar cómo se puede decidir en tiempo polinomial si es posible construir un conjunto de rutas de evacuación (usar flujo en redes para eso. Construir la red adecuada).

3. Para un evento solidario un conjunto de n personas se ofrecieron a colaborar. En el evento hay m tareas a desarrollar. Cada tarea tiene un cupo máximo de personas que la puede realizar. A su vez cada persona tiene ciertas habilidades que la hacen adecuadas para un subconjunto de tareas. Proponga una solución mediante red de flujos que maximice la cantidad de personas asignadas a las tareas. ¿Hay forma de lograr asegurarnos un piso mínimo de personas en cada tarea? ¿Cómo impacta en la solución presentada en el punto anterior?
4. La red de transporte intergaláctico es una de las maravillas del nuevo imperio terráqueo. Cada tramo de rutas galácticas tiene una capacidad infinita de transporte entre ciertos planetas. No obstante, por burocracia - que es algo que no los enorgullece - existen puestos de control en cada planeta que reduce cuantos naves espaciales pueden pasar por día por ella. Por una catástrofe en el planeta X , la tierra debe enviar la mayor cantidad posible de naves de ayuda. Por un arreglo, durante un día los planetas solo procesaran en los puestos de control aquellas naves enviadas para esta misión. Tenemos que determinar cuál es la cantidad máxima de naves que podemos enviar desde la tierra hasta el planeta X . Sugerencia: considerar a este un problema de flujo con capacidad en nodos y no en ejes
5. La compañía eléctrica de un país nos contrata para que le ayudemos a ver si su red de transporte desde su nueva generadora hidroeléctrica hasta su ciudad capital es robusta. Nos otorgan un plano con la red eléctrica completa: todas las subestaciones de distribución y red de cableados de alta tensión. Lo que quieren que le digamos es: cuantas secciones de su red se pueden interrumpir antes que la ciudad capital deje de recibir la producción de la generadora? (Sugerencia: investigue sobre el Teorema de Menger) Puede informar cual es el subconjunto de ejes cuya falla provoca este problema?
6. La red ARPANET, antecesora de internet, se creó para seguir funcionando incluso antes fallas en parte de su red. El país "Atrasoña" - que se mantuvo cerrado a los avances tecnológicos de las últimas décadas - ha decidido construir su propia red de redes. Han leído la documentación desclasificada de ARPANET y se han instruido en conectividad de redes. Proponen una red informática para unir sus principales organismos estatales. Nos convocan para que validemos su diseño. Debemos responder: ¿Cuántos cables de datos de la red se tienen que romper antes que la conectividad del grafo se rompa? (tener en cuenta que los cables de datos son bidireccionales) ¿Cuántos nodos se tienen que

romper antes que el grafo restante deje de ser conexo? (Sugerencia: piense en transformar de alguna forma los nodos para resolverlo mediante lo creado en el punto a)

7. La policía de la ciudad tiene " n " comisarías dispersas por la ciudad. Para un evento deportivo internacional deben asignar la custodia de " m " centros de actividades. Una comisaría y un centro de actividades pueden ser emparejados si y sólo si la distancia entre ellos no es mayor a un valor d . Contamos con la distancia entre todos los centros y las comisarías. Una comisaría sólo puede custodiar un centro. El centro puede ser custodiado por una comisaría. Determinar si es posible la asignación de tal forma que todos los centros estén custodiados. ¿Cómo modificaría la resolución del problema si en lugar de que cada centro de actividades i tenga que ser asignado a una sola comisaría, tenga que ser asignado a m_i comisarías? ¿Cómo modificaría la resolución del problema si además hubiera una restricción entre comisarías que implicaría que una comisaría N_i y una N_j no pudieran ser asignadas juntas a un centro M_i ? ¿Para qué casos dejaría de ser eficiente la resolución?
8. Una compañía minera nos pide que la ayudemos a analizar su nueva explotación. Ha realizado el estudio de suelos de diferentes vetas y porciones del subsuelo. Con estos datos se ha construido una regionalización del mismo. Cada región cuenta con un costo de procesamiento y una ganancia por extracción de metales preciosos. (En algunos casos el costo supera al beneficio). Al ser un procesamiento en profundidad ciertas regiones requieren previamente procesar otras para acceder a ellas. La compañía nos solicita que le ayudemos a maximizar su ganancia, determinando cuales son las regiones que tiene que trabajar. Tener en cuenta que el costo y ganancia de cada región es un valor entero. Para cada región sabemos cuales son aquellas regiones que le preceden. Resolver el problema planteado utilizando una aproximación mediante flujo de redes.
9. Un taller metalúrgico cuenta con un conjunto de M controles numéricos computarizados (CNC). Cada uno de ellos puede realizar trabajos de duración en bloques de 1 hora. Por otro lado, cuenta con N tareas a realizar. Cada tarea i tiene una duración de horas de desarrollo, una hora de posible comienzo (cuando puede comenzar a realizarse) y una hora de entrega (momento donde debe estar finalizada). Las tareas pueden realizarse parcialmente y utilizarse para su elaboración una o varias máquinas. Por ejemplo si la tarea A requiere 4 horas. Podría realizarse 1 hora en la máquina 1 y luego de un intervalo concluir su desarrollo en la máquina B (3 horas restantes). El jefe de planta nos solicita nuestra ayuda para que le ayudemos a determinar si podrá cumplir con la finalización de las tareas en tiempo y forma. Y en caso afirmativo le indiquemos cómo organizarla. Se solicita utilizando redes de flujos dar una solución al problema.
10. Una red de espías se encuentran diseminados por todo el país. Cada uno de ellos únicamente conoce a un número limitado de sus pares con los que puede tener contacto dejando un mensaje escrito en una ubicación conocida. Este conocimiento no es recíproco. En caso de una crisis la agencia puede enviar mensajes utilizando esta red

desde su base principal a un determinado agente especial. Una cuestión importante es que una vez utilizado un espía para transmitir un mensaje durante el resto de la crisis no se vuelve a utilizar. La agencia desea conocer, dada su red y un agente de destino de sus mensajes. ¿Cuál es la mínima cantidad de espías que un rival podría neutralizar para reducir en un 30% la cantidad de mensajes máximos que puede enviar desde la base al agente? Utilizando redes de flujos dar una solución al problema.

11. En un juego multijugador cooperativo de "p" participantes se muestra una grilla de $n \times n$ celdas. Inicialmente en una posición al azar se colocan los avatares de los jugadores e igual cantidad de cuevas. Cada "gusano" se desplaza 1 celda por turno ocupando una circundante que esté vacía (como máximo tiene 4: arriba, abajo, izquierda y derecha). A medida que se desplaza crece y por lo tanto sigue ocupando por las que pasó anteriormente. El objetivo del juego es lograr que los "p" jugadores lleven a sus gusanos a las cuevas (es indiferente a cual pero cada uno tiene que estar en una cueva diferente). Nos solicitan que realicemos el pseudocódigo que verifique en un turno determinado si aún es posible que los jugadores ganen. Es decir que todos los gusanos puedan llegar a la cueva.
12. Se cuenta con un grafo $G=(V,E)$ con capacidad 1 en cada uno de sus ejes. Existen 2 nodos que tomaremos como "s" fuente y "t" sumidero. Podemos determinar el flujo máximo F entre s-t. Se pide proponer un algoritmo eficiente que dado un valor "k", determine la cantidad mínima de ejes y cuáles de ellos eliminar para que el nuevo flujo máximo sea F-k. Determinar su complejidad y detallar mediante pseudocódigo y una explicación cómo funciona.
13. Una empresa de autobuses se conformó luego de la fusión de varias compañías menores. Actualmente tienen diferentes rutas que cubrir. Cada una con horario de inicio en una ciudad y finalización en otra. Existe la posibilidad de cubrir con un mismo micro diferentes rutas. Siempre la ruta comienza desde donde parte el micro, pero también puede pasar que el micro tenga tiempo suficiente para trasladarse hasta otro punto y cubrir otra ruta. Cuentan con una flota activa de N micros. Necesitan saber si les es posible cubrir con ella los requerimientos y si pueden contar con micros de backup ante la necesidad de controles programados de algunos de los móviles. Ayudar a resolver el problema mediante el uso de redes de flujo.
14. Contamos con una red de Flujo definida sobre el grafo $G=(V,E)$ y tenemos una asignación de flujo $f(e)$ sobre G. Nos solicitan elaborar un algoritmo que en base a esta información nos indique cómo se actualiza el flujo si uno de los ejes tiene un cambio de capacidad (puede ser positiva o negativa). Debemos evitar volver a ejecutar el algoritmo de Ford-Fulkerson desde cero. Explicar los diferentes casos que podrían suceder. Utilizar los conceptos de flujo máximo/corte mínimo en su explicación. Brindar pseudocódigo de su propuesta y análisis de complejidad.

15. Un grupo de "n" amigos participan en un torneo de voley. En cada partido se debe presentar una planilla de "y" jugadores. Se juegan partidos 1 vez por semana, durante un periodo de 9 meses. Para lograr que juegue la mayoría se propuso que cada amigo no juegue más de 4 partidos. Esos partidos deben estar lo más separados posible, por lo que no pueden jugar más de 1 partido por mes. Finalmente se tiene que tener en cuenta que algunos amigos en ciertas fechas no pueden asistir por cuestiones personales. Proponer un algoritmo polinomial que utilizando esta información realice la asignación de jugadores por partido (o que indique que con esas restricciones no es posible realizarlo)

Clases de complejidad

1. Nos brindan una caja negra A que, dado un grafo no dirigido $G = (V, E)$ y un número k se comporta de la siguiente manera:
- Si G es no conexo devuelve "no conexo".
 - Si G es conexo y tiene un conjunto independiente de tamaño al menos k devuelve "Sí".
 - Si G es conexo y no tiene un conjunto independiente de tamaño al menos k devuelve "No".

Mostrar cómo podríamos resolver el problema del máximo conjunto independiente en tiempo polinomial, usando llamadas a A si suponemos que A corre en tiempo polinomial en el tamaño de G y k. Describir la solución. ¿Tiene sentido la hipótesis de que A corre en tiempo polinomial en el tamaño de G y k? ¿Por qué?

2. Una agencia de marketing coloca publicidad en la Web. Se han ilusionado con vender publicidad con la siguiente idea, que llamaremos el problema de la Publicidad Estratégica: Un sitio Web se puede modelar como un grafo $G = (V, E)$. Las acciones habituales de los usuarios que visitan un sitio se pueden modelar mediante "t" recorridos posibles P_1, P_2, \dots, P_t (donde cada P_i es un camino dirigido en G). Dado un número k, se quieren elegir a lo sumo k vértices en G para poner publicidad, de modo tal que todos los "t" recorridos habituales pasen por al menos uno de esos vértices. Tenemos que mostrarle a esta empresa que su idea no es realizable por el momento ya que el problema de la Publicidad Estratégica es NP-completo. Sugerencia: relacionarlo con cubrimiento de vértices.
3. Un almacén registra en una matriz qué productos compra cada uno de sus clientes. Un conjunto de clientes es diverso si cada uno de ellos compra cosas diferentes (tiene intersección vacía con lo que compran los demás). Definimos al problema de los clientes diversos como: Dada una matriz de registro, de tamaño m (clientes) x n (productos), y un número $k \leq m$, ¿existe un subconjunto de tamaño al menos k de los clientes que sea diverso? Probar que el problema es NP-completo. Sugerencia: Reducir polinomialmente conjuntos independientes a clientes diversos.

4. La siguiente es una versión de Conjunto Independiente. Dado un grafo $G = (V, E)$ y un entero k , decimos que $I \subseteq V$ es fuertemente independiente si dados dos vértices u y v en I , la arista (v, u) no pertenece a E y además no hay ningún camino de tamaño 2 (con dos aristas) de u a v . El problema de Conjuntos Fuertemente Independientes consiste en decidir si G tiene un conjunto fuertemente independiente de tamaño al menos k . Probar que el problema de Conjuntos Fuertemente Independientes es NP completo. Utilizar para ello que Conjuntos Independientes es NP completo.
5. Problema del camino más largo en un grafo general con aristas sin peso: Dados $G = (V, E)$ un grafo no dirigido y un natural k , determinar si existe un camino simple en G de longitud $\geq k$. Probar que es un problema NP-completo (Usar el problema del camino hamiltoniano para probarlo).
6. Nos piden que organicemos una jornada de apoyo de estudio para exámenes. Tenemos que poder dar apoyo a " n " materias y hemos recibido currículos de " m " postulantes para ser potenciales ayudantes. Cada ayudante puede ayudar en un determinado subconjunto de materias. Para cada una de las materias hay un subconjunto de postulantes que pueden dar apoyo en ella. La pregunta es: dado un número $k < m$, ¿es posible seleccionar a lo sumo " k " ayudantes de modo tal que siempre haya un ayudante que pueda dar consultas en alguna de las n materias? Este problema se llama Contratación Eficiente. Probar que "Contratación Eficiente" es NP-completo. Sugerencia: se puede tratar de usar Cubrimiento de Vértices.
7. Una de las parejas más ricas del mundo está pasando por un proceso de divorcio. Entre sus bienes cuentan con propiedades, autos, motos, estampillas raras y otros coleccionables. Como no se ponen de acuerdo en la manera de dividirlos, el juez ha dictaminado que un tasador ponga valor a cada bien y luego se haga una partición por valores iguales (el problema abstracto se conoce como 2-partition) El juez nos pide que elaboremos un algoritmo que en forma eficiente haga este trabajo. Demuestre que la solución pedida es NP-completa. Sugerencia: Pruebe con "subset sum".
8. El juego "escaleras y serpientes" consiste en un tablero con celdas numeradas del 1 al n . Cada jugador inicia en la casilla 1 y cada turno arroja un dado de 6 caras para determinar cuántas casillas avanza. En algunas casillas existen escaleras que permiten automáticamente ascender desde su pie hasta la cima. Al caer en ellas el jugador asciende a la casilla donde se apoya la cima de la escalera. En otras casillas existen serpientes que obligan al jugador a descender de su cabeza a cola. Al caer en una casilla donde se encuentra la cabeza de la serpiente el jugador desciende hasta la casilla donde se encuentra la cola de la misma. Diferentes empresas generan diferentes tableros (valor de n y ubicación y cantidad de serpientes y escaleras). El creador de una nueva versión nos muestra su prototipo y quiere que le digamos cuántos turnos durará como mínimo el juego. Se pide:
 - Resolver el problema reduciéndolo a un problema de grafos.
 - Indicar detalladamente todos los pasos de la reducción.

- Plantee el problema como un problema de decisión. ¿Puede afirmar que el problema pertenece a la clase P? ¿Puede afirmar que pertenece a NP?
- 9. Se conoce Bin-Packing al problema de decisión donde se cuenta con “N” elementos de diferentes pesos y con “M” contenedores de cierta capacidad. Queremos saber si es posible acomodar todos los elementos en no más de k contenedores. Se pide demostrar que el problema es NP-Completo. Sugerencia utilizar 2-partition.
- 10. Un grupo de amigos que conviven están mudándose a un departamento nuevo. Han juntado sus pertenencias en cajas de diferentes volúmenes que recolectaron en supermercados y tiendas. Al llegar la compañía de mudanza les informan que por normativa únicamente transportarán utilizando como contenedores sus recipientes de volumen V. Por lo tanto, los amigos deben ingresar sus cajas en los contenedores autorizados. En el camión entran como máximo “r” recipientes. Al llenarse realiza el trayecto para descargar y regresar a cargar otros contenedores. Antes de proceder quieren saber si podrán acomodar todas sus cajas de tal forma que puedan realizar menos de k viajes. Demostrar que es un problema NP-Completo. Sugerencia: Este problema es fácilmente relacionable con Bin Packing.
- 11. Hemos visto como cualquier instancia de SAT se puede reducir polinomialmente a una instancia de 3SAT. Nos concentramos en la variante 2SAT. En esta se cuenta con dos (ni más ni menos) literales por cada cláusula de la expresión booleana. Queremos demostrar que 2SAT pertenece a la clase de complejidad P. Plantee teóricamente que debería realizar para demostrar su pertenencia a la clase P mediante una reducción polinomial. Considere los siguientes sugerencias y elabore la reducción utilizando un problema de grafos:
 - Repensar una cláusula (A o B) como: (“no B” entonces A) y (“no A” entonces B).
 - Pensar en encadenamiento de implicancias y una situación que podría ocasionar un absurdo lógico que impida satisfacer la expresión booleana.
- 12. Definimos el problema Subgrafo denso de la siguiente manera: Dado un grafo $G=(V,E)$ y dos parámetros a y b. Existe en G un subconjunto de “a” vértices con al menos “b” ejes entre ellos. Demostrar que este problema es NP-Completo. Sugerencia: Utilizar el problema del Clique.
- 13. Definimos al problema de Set Packing como: Dado “n” conjuntos S_1, S_2, \dots, S_n y un parámetro k. Queremos saber si existe una colección de tamaño k de los subconjuntos tales que ningún elemento contenido en ellos está repetido en estos “k” conjuntos? Demostrar que este problema es NP-Completo. Sugerencia: Utilizar Conjunto independiente.
- 14. El directorio de una empresa realizará una cena de fin de año. En total son “n” directivos que se deben sentar alrededor de una mesa circular. Lamentablemente existen conflictos entre algunos de ellos que impiden que se sienten uno al lado del otro. Dado

una instancia del problema, que incluye los n directivos y un listado donde se ven aquellos pares de directivos que están peleados entre sí, determinar si es posible sentarse en la mesa. Demostrar que el problema es NP-C. Sugerencia: Utilizar ciclo Hamiltoniano.

15. Dado un grafo $G=(V,E)$ no dirigido se denomina como Feedback set a un subconjunto $X \subseteq V$ de vértices tal que el grafo resultante de eliminar los vértices de X y los ejes adyacentes a estos no tiene ciclos. El problema de decisión de Undirected Feedback Set quiere responder si dado un grafo G no dirigido existe un feedback set de tamaño k o menor. Demostrar que este problema es NP-Completo. Sugerencia: Utilizar Vertex Cover.
16. Definimos el problema DOBLE-SAT como: dado una fórmula booleana determinar si existen dos asignaciones de variables que satisfacen a la misma. Probar que DOBLE-SAT pertenece a NP-C. Sugerencia: Utilizar 3SAT
17. Definimos el problema del Hitting Set como: dado un conjunto finito S de " n " elementos, una colección C de subconjuntos de S , un número positivo $K \leq n$, ¿existe un subconjunto $S' \subseteq S$ tal que S' contiene al menos un elemento de cada subconjunto de C y $|S'| \leq K$? Demostrar que este problema es NP-Completo. Sugerencia: utilizar Vertex Cover.
18. Una compañía multinacional desea contratar cobertura satelital para sus " n " sedes repartidas por el mundo. Han averiguado entre varias empresas que proveen el servicio pero ninguna de ellas tiene cobertura total. Les gustaría poder contratar a " k " o menos empresas. Pero tienen una condición adicional: al menos una de sus sedes debe tener cobertura de todas las empresas que la ofrecen. Con eso pueden iniciar una certificación de calidad que necesitan. Se pide: Demostrar que el problema es NP-Completo. Sugerencia: Utilizar Set Cover
19. El problema del Ciclo Hamiltoniano dirigido corresponde a una variante del problema de Ciclo Hamiltoniano con la diferencia que la instancia corresponde a un grafo dirigido. Demostrar que este problema pertenece a NP-C. Sugerencia: Puede utilizar Ciclo Hamiltoniano.
20. La elaboración de una flota de " n " minisatélites requiere la integración de 4 componentes cuyos códigos de identificación son pA , pB , pC y pD . Contamos con " n " piezas de cada uno de ellos. Un estudio de compatibilidad informa que no cualquier cuadrupla de piezas es viable para ensamblar el satélite. Nos proveen un listado de las cuadruplas que si pueden conformarlos. Queremos saber si es posible seleccionar de forma adecuada las piezas para armar los " n " satélites. Se pide: Demostrar que el problema es NP-Completo. Sugerencia: Se puede utilizar 3 Dimensional Matching
21. Un artesano puede contratar x máquinas de impresión 3D. Se ha comprometido en realizar un conjunto de " N " pedidos. Cada pedido " i " cuenta con un tiempo " t_i " en horas de realización. Dentro de D días debe entregarlas. Nos solicita que encontremos un

procedimiento que le indique si es posible realizarlo en el tiempo disponible y programar que pedido realizar en cada máquina. Demostrar que el problema es NP-C. Sugerencia: Puede utilizar 2-partition

22. Un departamento dentro de una universidad adquirió “n” proyectores para dar clases durante el cuatrimestre. Envío un formulario a los docentes de las distintas materias para conocer si los necesitaban como complemento para sus clases. Un subconjunto de docentes respondió afirmativamente. Sabiendo que cada materia tiene clases 1 o más veces por semana en un horario establecido. Y sabiendo que los horarios de varias de esas materias se superponen. Nos solicitan determinar si la cantidad comprada alcanza o si se tiene que dejar a docentes sin acceso a esas cuentas. Demostrar que lo solicitado es NP-COMPLETO. Sugerencia: Tal vez le resulte útil “k” coloreo de grafos.
23. Para elaborar una película de detectives un grupo de escritores se ha juntado para elaborar una trama atrapante y que tenga coherencia. En largas jornadas han propuesto un gran conjunto de premisas, giros argumentales y eventos claves. Lamentablemente algunas de ellas no son compatibles entre sí. Por cada situación han anotado con cual no es compatible. Desean poder seleccionar un conjunto de N premisas compatibles para presentar a los productores como idea inicial. Se pide: Demostrar que el problema es NP-Completo. Sugerencia: Tal vez le resulte útil clique
24. El plan de evacuación ante ataques zombies/alienígenas de la ciudad implica poder trasladar a los más importantes científicos, militares y políticos designados a refugios. Diferentes ramas y organizaciones gubernamentales presentaron diferentes rutas de evacuación. Cada una corresponde un punto de encuentro, un recorrido y un refugio. En total existen N rutas presentadas. Es importante que ninguna ruta comparta o se cruce en su recorrido con otra para maximizar la posibilidad de supervivencia. Debemos responder si con las propuestas se pueden seleccionar al menos K caminos de tal forma que cumpla estas restricciones. Demuestre que lo solicitado es NP-COMPLETO. Sugerencia: Se puede realizar con 3 Dimensional Matching o INDEPENDENT-SET
25. Un país está modificando su red radiofónica. Existen muchas estaciones de radios cada una con su propia frecuencia de transmisión. Desean reasignarlas de tal forma de no usar más de N frecuencias. El problema es que ciertas radios por su ubicación geográfica y potencia pueden interferir con otras. Un estudio informa cuántas radios hay, y para cada radio con cuáles hay riesgo de interferencia. Quisieran determinar si es posible realizar la reasignación. Demuestre que el problema planteado es NP-COMPLETO. HINT!: Tal vez le resulte útil coloreo de grafos.
26. Para un evento a realizar se requiere conformar una selección musical entre el conjunto A de “n” canciones. Podemos enumerar a los elementos de A como a_1, a_2, \dots, a_n . Por otra parte, contamos con un conjunto “B” de “m” personas. Cada una de ellas con un subconjunto de esas canciones que le gustan. Deseamos saber si podemos seleccionar un subconjunto de no más de “k” canciones, de tal forma que existe al menos 1 canción que

le guste a cada uno. Se pide: Demostrar que el problema es NP-Completo. Sugerencia: Se puede utilizar Vertex Cover.

27. Presentamos una variante de SAT conocida como NAE-SAT. En este, se busca satisfacer una expresión en forma normal conjuntiva con la condición adicional de que en cada cláusula al menos un literal tenga valor "false". Nos centraremos en la variante NAE-4-SAT. En esta cada cláusula tiene 4 literales. Demostrar que corresponde a un problema NP-Completo. Sugerencia: Partir de 3-SAT.
28. Habiendo demostrado previamente que el problema NAE-4-SAT es np-completo, demostrar que NAE-3-SAT también. Utilizar para la demostración el primero.
29. Un nivel de un videojuego "escape room" consiste en un conjunto de salas contiguas. Todas las salas son iguales, excepto la primera y la última. Las salas intermedias tienen 2 sets de 3 puertas cada una. El primer trío conduce a la habitación anterior. Y el segundo trío la habitación siguiente. La particularidad de estas puertas es que son de 1 solo sentido. Dependen de una palanca que puede permitir únicamente entrar o salir (pero no las dos). La última sala tiene un cofre con una llave que abre la puerta de salida de la primera habitación. En la primera habitación se encuentra un panel con un conjunto de palancas de 2 posiciones y un diagrama que muestra qué puertas están controladas por cada palanca. Una puerta sólo está controlada por una palanca. Pero una palanca puede controlar 1 o más puertas. Otra particularidad de este ingenio es que una misma palanca puede hacer que una puerta permita acceder a cuartos siguientes y otra puerta a cuartos anteriores. Ejemplo: Activar la palanca A, permite ingresar desde la puerta 1 de la "sala 1" a la "sala 2". Además permite regresar de la "sala 3" a la "sala 2". Si se desactiva la palanca se invierten los sentidos de las puertas. Se desea encontrar una configuración de palancas de tal manera que se pueda acceder a la sala del cofre, retirar la llave y luego regresar a la sala inicial para escapar por la puerta. Demostrar que es un problema NP-Completo. Sugerencia: Utilizar NAE-3-SAT.

Algoritmos randomizados

- 1) Un nuevo regente del país "Atrasoña" llegó al poder de la mano de 3 aliados. Como todo monarca decide repartir los puestos clave de poder para dejar contentos a todos ellos. Por otro lado, no desea que se formen cúmulos de poder que dificulten su dominio. Para ayudarse ha generado un diagrama donde se pueden ver los diferentes puestos y con qué otros puestos claves se relaciona. Lo que intenta es lograr que para cada par de puestos relacionados entre sí, las facciones que lo controlen sean diferentes. Existe un número total de relaciones entre puestos y lo que desea es que la mayor cantidad posible de ellos cumplan el criterio de "balanceo de poder"
 - a) Proponer un algoritmo randomizado para lograr al menos $\frac{2}{3}$ del total posible de relaciones balanceados.

- 2) El problema de set independiente corresponde a la clase NP-C. No obstante, dado la gran cantidad de aplicaciones de este, nos interesaría contar con algún mecanismo para obtener dado un grafo $G=(V,E)$ el subconjunto de vértices independientes de mayor tamaño posible.

Acotamos el problema a un grafo con n vértices, donde cada uno de ellos tiene exactamente d ejes. Analizar el siguiente algoritmo randomizado:

*Para cada vértice i seleccionar de forma uniformemente aleatoria x_i el valor 0 o 1.
Seleccionar para el set independiente únicamente aquellos vértices con el valor 1 que no se conecten mediante ejes con otros vértices que tengan el valor 1.*

- a) Probar que el set resultante es independiente.
 - b) Calcular el tamaño esperado del set independiente (en función de n y d)
 - c) Para el cálculo de x_i utilizamos $p=1/2$. Determinar si este es el valor mas conveniente o podemos encontrar uno mejor. Si es así, de la fórmula del valor esperado con esta nueva probabilidad.
- 3) Un estudio de videojuegos está lanzando un nuevo juego para celular que permite competir entre 2 jugadores. Cada jugador tiene un conjunto de criaturas de las que debe elegir un subconjunto de tamaño N . Una vez que se elijan se enfrentarán uno a uno. Para lograr una pelea más emocionante se mezclan los conjuntos. De esa forma en cada batalla van apareciendo de forma aleatoria las criaturas elegidas.

El siguiente es el pseudocódigo que han implementado:

```
Sea mezclado un vector de 1 a n vacío
Sea criatura un vector de 1 a n con las criaturas
seleccionadas
Desde  $i=1$  a  $n$ 
     $x = \text{random}(1..n)$ 
    Sea encontrado = false
    While not encontrado:
        if (criatura[x] != null):
            mezclado[i] = criatura[x]
            criatura[x] = null
            encontrado = true
        else:
             $x++$ 
    if ( $x > n$ ):
         $x = \text{random}(1..n)$ 
```

Determinar si es un buen método de mezclado aleatorio. En caso negativo proponer otro que lo sea.

Algoritmos de aproximación

- 1) Un río posee un conjunto de buques transportadores, cada uno con una capacidad de K kilos que cruzan automóviles de una orilla a la otra. Es un paso muy concurrido y requiere reservar con algunas horas de anticipación para el cruce. En ese proceso los vehículos son registrados y pesados.

El encargado va cargando los vehículos a medida que se van presentando. En simultáneo va calculando el peso transportado. En el momento que llega un vehículo que de ingresar supere la máxima capacidad de carga. Lo hace esperar. Cierra el barco. El barco cruza, descarga y vuelve por el próximo cargamento.

a) Demostrarle al encargado que su método puede no utilizar la menor cantidad de viajes posibles.

b) Demostrarle al encargado que su método no superará el doble de la menor cantidad de viajes posibles para ningún conjunto posible de vehículos y diferentes valores de k

- 2) En un concurso televisivo nos ofrecen un vale por " D " dólares y un catálogo de productos. Cada producto tiene un valor. La consigna del juego es seleccionar un subconjunto de productos que no supere el precio del vale (si lo hacemos nos quedamos con las manos vacías).

Nuestro compañero sugiere simplemente ir seleccionando los productos en el orden del catálogo, agregando aquellos cuya adición no supere el valor D . Por otro lado, a nosotros no nos parece la mejor alternativa.

a) Dar un ejemplo donde esta estrategia nos puede dar menos de la mitad que la solución máxima posible.

b) Proponer una solución de complejidad no mayor a $O(n \log n)$ que nos asegure obtener una solución en el peor caso más a la mitad que la solución máxima posible.

- 3) Un centro de fotocopiado tiene un conjunto de máquinas para procesar diferentes trabajos. Las mismas las compró en dos momentos diferentes. Teniendo " m " fotocopadoras de la primera compra y " k " fotocopadoras más modernas de la segunda. Como la tecnología mejoró las más nuevas trabajan el doble de rápido que las viejas. Cada día recibe un conjunto de pedidos con diferentes longitudes. Desea asignar las tareas a las máquinas de tal forma de terminar lo antes posible la jornada de trabajo.

Proponer un algoritmo de aproximación de tal forma que la diferencia con el óptimo sea como mucho 3 veces mayor.

- 4) Un rey aburrido (y bastante sádico) ha propuesto un acertijo a todo el reino. Construyó en sus jardines una gran cuadrícula de $n \times n$ celdas. En cada una de ellas depositó una

bolsa con monedas de oro. Un gran cartel informa cuántas monedas hay en cada una de ellas y se puede ver que no hay dos celdas con igual cantidad de monedas. La cuadrícula solo se puede ver desde lo alto de la torre de su castillo.

El desafío consiste en seleccionar un subconjunto de celdas cuya suma sea la mayor posible. Esas celdas no pueden ser contiguas entre sí (una celda central por ejemplo tiene 6 celdas contiguas). Los candidatos irán ingresando de a uno y tendrán un tiempo acotado para indicar su solución. A medida que pasen se los aislará para que no puedan informar a otros que vieron ni que propusieron. El ganador se podrá llevar todo el oro, los perdedores serán llevados como mano de obra esclava a las minas del reino.

Un compañero de aventuras nos cuenta que piensa presentarse y que tiene un método infalible para seleccionar las celdas: Comenzará eligiendo desde la más cara a la menos cara. Y con cada selección desechará aquellas inválidas luego de la elección.

- a) Demuestre a su amigo que su método no es infalible
- b) Demuestre que su método devuelve un conjunto independiente de peso total al menos $1/4$ veces el peso total máximo de cualquier conjunto independiente en el gráfico de cuadrícula G .

HINT: Se puede ayudar demostrando previamente: Sea S el conjunto independiente devuelto por su algoritmo, y sea T cualquier otro conjunto independiente en G . Para cada nodo $v \in T$, el nodo $v \in S$, o existe un $v' \in S$ tal que $w(v) \leq w(v')$ y (v, v') son contiguos.