EJERCICIOS Temas 2 y 3. Soluciones con ejemplo.

1. Un camionero conduce desde Bilbao a Málaga siguiendo una ruta dada y llevando un camión que le permite, con el tanque de gasolina lleno, recorrer n kilómetros sin parar. El camionero dispone de un mapa de carreteras que le indica las distancias entre las gasolineras que hay en su ruta. Como va con prisa, el camionero desea pararse a repostar el menor número de veces posible. Diseñar un algoritmo voraz para determinar en qué gasolineras tiene que parar.

2. Se tienen n esquiadoras de alturas H = [h1, h2, . . . , hn] y n pares de esquíes de longitudes S = [s1,s2, . . . ,sn]. Diseñar un algoritmo que usando un esquema voraz, asigne los esquíes a las esquiadoras de modo que el promedio de la diferencia (en valor absoluto) entre la altura de la esquiadora y la longitud de los esquíes asignados sea mínima.

3. Hay M farolas en las posiciones y1,…,yM de una recta y N puntos x1,…,xN. Cada farola tiene un radio de iluminación ri, tal que la i-ésima farola ilumina puntos en el intervalo [yi−ri,yi+ri]. Se quiere encender el mínimo número de farolas tales que cada uno de los N puntos x1,…xN. esté iluminado por al menos una farola. Encuentra este mínimo número.

4. Se tiene un vector V[1..N] formado por números enteros, de manera que todos ellos distintos, y que están ordenados de manera creciente. Se dice que un vector de estas características es coincidente si tiene al menos una posición tal que es igual al valor que contiene el vector en esa posición. Por ejemplo, en el vector

1	2	3	4	5	6	7	8
-14	-6	3	6	16	18	27	43

   puede verse que V[3] = 3; por lo tanto, este vector es coincidente.
   Diseñar un algoritmo Divide y Vencerás que determine en un orden de eficiencia no superior a O(log N) si un vector es coincidente, recibiendo como datos el vector y su tamaño.

5. En Acelandia el deporte nacional es el tenis. Existe un ranking, donde cada jugador tiene asignado un número de puntos en función de su calidad, es decir, el mejor jugador de ese país es el que tiene más puntos. Cada año se debe seleccionar una pareja entre todos los tenistas de Acelandia para jugar un torneo de dobles a nivel internacional. El procedimiento de selección es un poco peculiar. Se coloca la puntuación de cada uno de los jugadores en una lista, de forma totalmente aleatoria, sin ningún tipo de ordenación. Una vez hecha la lista, cada jugador solo puede formar pareja con un jugador contiguo dentro de la lista, es decir, que esté delante o detrás de él en esa lista. Obviamente, el primer jugador de la lista solo puede formar pareja con el segundo y el último con el penúltimo, pero el resto tiene dos posibles opciones para formar la pareja de dobles, correspondientes a los jugadores anterior y posterior de la lista. Con esta restricción, se elige la mejor pareja

de dobles posible, que es aquella en la que la suma de los puntos de sus dos componentes sea mayor. Diseñar un algoritmo cuya función principal siga el esquema de divide y vencerás, que decida qué pareja de dobles debe competir en Acelandia. Razonar cuál es la complejidad del algoritmo.

Material Didáctico: Explicación de Algoritmos para los Ejercicios de los Temas 2 y 3

Ejercicio 1: un camionero conduce desde Bilbao a Málaga siguiendo una ruta dada y llevando un camión que le permite, con el tanque de gasolina lleno, recorrer n kilómetros sin parar. El camionero dispone de un mapa de carreteras que le indica las distancias entre las gasolineras que hay en su ruta. Como va con prisa, el camionero desea pararse a repostar el menor número de veces posible. Diseñar un algoritmo voraz para determinar en qué gasolineras tiene que parar.

Planteamiento del problema: Un camionero debe viajar desde Bilbao a Málaga con un camión que, con el tanque lleno, puede recorrer hasta *n* kilómetros sin repostar. Dispone de un mapa con la ubicación de *G* gasolineras en la ruta y necesita hacer el menor número de paradas posibles.

Explicación del algoritmo:

1. Partimos de Bilbao con el tanque lleno.
2. Siempre avanzamos lo más lejos posible sin repostar.
3. Nos detenemos en la gasolinera más lejana que podamos alcanzar con la gasolina disponible.
4. Repetimos hasta llegar a Málaga.

Ejemplo ilustrativo: Si *n = 400 km* y las gasolineras están a las siguientes distancias desde Bilbao: 150, 300, 350, 500, 700 km, la estrategia sería:

- Partimos de Bilbao.
- La primera parada sería en el km 300.
- Luego, avanzamos hasta el km 500.
- Finalmente, alcanzamos Málaga en el km 700.

Esto minimiza las paradas necesarias.

Ejercicio 2: Se tienen n esquiadoras de alturas H = [h1, h2, . . . , hn] y n pares de esquíes de longitudes S = [s1,s2, . . . ,sn]. Diseñar un algoritmo que usando un esquema voraz, asigne los esquíes a las esquiadoras de modo que el promedio de la diferencia (en valor absoluto) entre la altura de la esquiadora y la longitud de los esquíes asignados sea mínima.

Planteamiento del problema: Tenemos *n* esquiadoras con alturas *H = [h1, h2, ..., hn]* y *n* pares de esquíes con longitudes *S = [s1, s2, ..., sn]*. Queremos asignar los esquíes de forma que se minimice la diferencia absoluta entre la altura de cada esquiadora y la longitud de sus esquíes.

Explicación del algoritmo:

1. Ordenamos ambas listas (H y S) de menor a mayor.
2. Asignamos a cada esquiadora los esquíes en la misma posición ordenada.

Ejemplo ilustrativo: Si tenemos las alturas *H = [150, 160, 170]* y los esquíes *S = [155, 165, 175]*, las asignaciones serían:

- Esquiadora de 150 cm → esquíes de 155 cm.
- Esquiadora de 160 cm → esquíes de 165 cm.
- Esquiadora de 170 cm → esquíes de 175 cm.

Ordenar y asignar directamente garantiza la mejor combinación.

Ejercicio 3: Hay M farolas en las posiciones y1,…,yM de una recta y N puntos x1,…,xN. Cada farola tiene un radio de iluminación ri, tal que la i-ésima farola ilumina puntos en el intervalo [yi−ri,yi+ri]. Se quiere encender el mínimo número de farolas tales que cada uno de los N puntos x1,…xN. esté iluminado por al menos una farola. Encuentra este mínimo número.

Planteamiento del problema: Dada una serie de farolas con radios de iluminación y una serie de puntos en una recta, queremos encender la menor cantidad de farolas para cubrir todos los puntos.

Explicación del algoritmo:

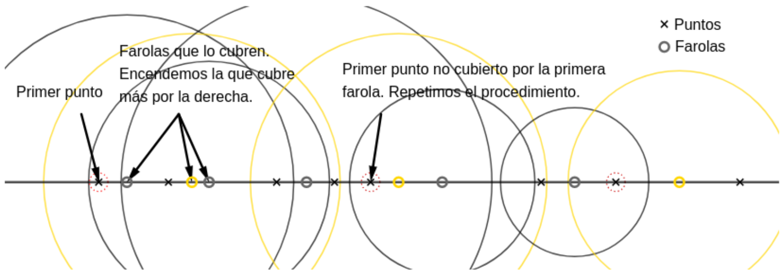
1. Ordenamos las farolas por su punto más a la derecha iluminado.
2. Recorremos los puntos y encendemos la farola más lejana posible que los cubra.
3. Seguimos hasta iluminar todos los puntos.

Ejemplo ilustrativo: Si tenemos farolas en *y = [2, 4, 6]* con radios *r = [1, 2, 1]*, sus intervalos de iluminación serían:

- Farola en 2 ilumina [1,3]
- Farola en 4 ilumina [2,6]
- Farola en 6 ilumina [5,7]

Encender solo las farolas en 4 y 6 sería suficiente.

Para resolver este problema, nos imaginamos los puntos y las farolas en la recta, ordenados de izquierda a derecha. Fijémonos en el primer punto (por la izquierda): tendrá que haber una farola que lo ilumine. Entre todas las que lo pueden iluminar, ¿cuál es la mejor? Claramente no nos importa cuánto cubra por la izquierda de ese punto, ya que no hay ningún punto a la izquierda de ese que nos interese iluminar. Sin embargo, sí que nos interesa que cubra lo máximo posible por la derecha. Por tanto, entre las farolas que cubren el primer punto, elegimos una cuyo alcance por la derecha sea máximo.



Una vez hemos elegido esta farola, algunos puntos adicionales quedarán cubiertos por ella. Pasamos al siguiente punto por la izquierda que no esté cubierto. Nuevamente tenemos que elegir una farola que lo cubra y otra vez no nos importa cuánto cubra por la izquierda, ya que todos los puntos a la izquierda de él ya están iluminados por la otra farola. Así que elegimos la que más cubra por la derecha. Pasamos al siguiente punto que no está iluminado y repetimos la idea hasta que tenemos todos los puntos iluminados. Esta solución se puede implementar fácilmente en $\mathcal{O}(N \log N + NM)$: se ordenan los puntos en $\mathcal{O}(N \log N)$ y para cada uno de los puntos que vayamos procesando iteramos por todas las farolas en $\mathcal{O}(M)$ y entre las que lo cubren elegimos la que $y_i + r_i$ sea máximo. (En casos normales procesaremos bastante menos que N puntos, como se puede ver en el diagrama, en el que sólo procesamos 3 de 8. Sin embargo, en el peor de los casos procesamos N, y de aquí la complejidad $\mathcal{O}(NM)$).

Ejercicio 4: Se tiene un vector V[1..N] formado por números enteros, de manera que todos ellos distintos, y que están ordenados de manera creciente. Se dice que un vector de estas características es coincidente si tiene al menos una posición tal que

1	2	3	4	5	6	7	8
-14	-6	3	6	16	18	27	43

es igual al valor que contiene el vector en esa posición. Por ejemplo, en el vector

puede verse que V[3] = 3; por lo tanto, este vector es coincidente.
Diseñar un algoritmo Divide y Vencerás que determine en un orden de eficiencia no superior a O(log N) si un vector es coincidente, recibiendo como datos el vector y su tamaño.

Planteamiento del problema: Dado un vector ordenado de enteros distintos, queremos determinar si existe una posición *i* tal que *V[i] = i*.

Explicación del algoritmo:

1. Aplicamos búsqueda binaria:
   - Si *V[m] = m*, retornamos *true*.
   - Si *V[m] > m*, buscamos en la mitad izquierda.
   - Si *V[m] < m*, buscamos en la mitad derecha.
2. Repetimos hasta encontrar el valor o agotar la búsqueda.

Ejemplo ilustrativo: Si *V = [-2, 0, 2, 3, 6]*:

- Mitad en *V[2] = 2*, cumple *V[i] = i*, así que retornamos *true*.

6. **Ejercicio 5:** En Acelandia el deporte nacional es el tenis. Existe un ranking, donde cada jugador tiene asignado un número de puntos en función de su calidad, es decir, el mejor jugador de ese país es el que tiene más puntos. Cada año se debe seleccionar una pareja entre todos los tenistas de Acelandia para jugar un torneo de dobles a nivel internacional. El procedimiento de selección es un poco peculiar. Se coloca la puntuación de cada uno de los jugadores en una lista, de forma totalmente aleatoria, sin ningún tipo de ordenación. Una vez hecha la lista, cada jugador solo puede formar pareja con un jugador contiguo dentro de la lista, es decir, que esté delante o detrás de él en esa lista. Obviamente, el primer jugador de la lista solo puede formar pareja con el segundo y el último con el penúltimo, pero el resto tiene dos posibles opciones para formar la pareja de dobles, correspondientes a los jugadores anterior y posterior de la lista. Con esta restricción, se elige la mejor pareja de dobles posible, que es aquella en la que la suma de los puntos de sus dos componentes sea mayor. Diseñar un algoritmo cuya función principal siga el esquema de divide y vencerás, que decida qué pareja de dobles debe competir en Acelandia. Razonar cuál es la complejidad del algoritmo.

Planteamiento del problema: Dada una lista desordenada de puntuaciones de tenistas, queremos encontrar la pareja contigua con la mayor suma de puntuaciones.
Explicación del algoritmo:

1. Dividimos la lista en dos mitades.
2. Calculamos la mejor pareja en cada mitad.
3. Calculamos la mejor pareja en la frontera entre mitades.
4. La mejor pareja global es la mejor de estos tres casos.

Ejemplo ilustrativo: Si *P = [5, 1, 8, 10, 7, 3]*:

- Mitades: [5, 1, 8] y [10, 7, 3].
- Mejores parejas: (8,10) en la frontera, (1,8) en la izquierda, (7,3) en la derecha.
- La mejor es (8,10).

Complejidad: *O(N log N)*.