

3. Una panadería recibe n pedidos por importes m_1, \dots, m_n , pero sólo queda en depósito una cantidad H de harina en buen estado. Sabiendo que los pedidos requieren una cantidad h_1, \dots, h_n de harina (respectivamente), determinar el máximo importe que es posible obtener con la harina disponible.
4. Usted se encuentra en un globo aerostático sobrevolando el océano cuando descubre que empieza a perder altura porque la lona está levemente dañada. Tiene consigo n objetos cuyos pesos p_1, \dots, p_n y valores v_1, \dots, v_n conoce. Si se desprende de al menos P kilogramos logrará recuperar altura y llegar a tierra firme, y afortunadamente la suma de los pesos de los objetos supera holgadamente P . ¿Cuál es el menor valor total de los objetos que necesita arrojar para llegar sano y salvo a la costa?
5. Sus amigos quedaron encantados con el teléfono satelital, para las próximas vacaciones ofrecen pagarle un alquiler por él. Además del día de partida y de regreso (p_i y r_i) cada amigo ofrece un monto m_i por día. Determinar el máximo valor alcanzable alquilando el teléfono.
6. Un artesano utiliza materia prima de dos tipos: A y B . Dispone de una cantidad MA y MB de cada una de ellas. Tiene a su vez pedidos de fabricar n productos p_1, \dots, p_n (uno de cada uno). Cada uno de ellos tiene un valor de venta v_1, \dots, v_n y requiere para su elaboración cantidades a_1, \dots, a_n de materia prima de tipo A y b_1, \dots, b_n de materia prima de tipo B . ¿Cuál es el mayor valor alcanzable con las cantidades de materia prima disponible?
7. En el problema de la mochila se buscaba el máximo valor alcanzable al seleccionar entre n objetos de valores v_1, \dots, v_n y pesos w_1, \dots, w_n , respectivamente, una combinación de ellos que quepa en una mochila de capacidad W . Si se tienen dos mochilas con capacidades W_1 y W_2 , ¿cuál es el valor máximo alcanzable al seleccionar objetos para cargar en ambas mochilas?

8. Una fábrica de automóviles tiene dos líneas de ensamblaje y cada línea tiene n estaciones de trabajo, $S_{1,1}, \dots, S_{1,n}$ para la primera y $S_{2,1}, \dots, S_{2,n}$ para la segunda. Dos estaciones $S_{1,i}$ y $S_{2,i}$ (para $i = 1, \dots, n$), hacen el mismo trabajo, pero lo hacen con costos $a_{1,i}$ y $a_{2,i}$ respectivamente, que pueden ser diferentes. Para fabricar un auto debemos pasar por n estaciones de trabajo $S_{i_1,1}, S_{i_2,2}, \dots, S_{i_n,n}$ no necesariamente todas de la misma línea de montaje ($i_k = 1, 2$). Si el automóvil está en la estación $S_{i,j}$, transferirlo a la otra línea de montaje (es decir continuar en $S_{i',j+1}$ con $i' \neq i$) cuesta $t_{i,j}$. Encontrar el costo mínimo de fabricar un automóvil usando ambas líneas.

9. El juego $\searrow U \uparrow P \nearrow$ consiste en mover una ficha en un tablero de n filas por n columnas desde la fila inferior a la superior. La ficha se ubica al azar en una de las casillas de la fila inferior y en cada movimiento se desplaza a casillas adyacentes que estén en la fila superior a la actual, es decir, la ficha puede moverse a:

- la casilla que está inmediatamente arriba,
- la casilla que está arriba y a la izquierda (si la ficha no está en la columna extrema izquierda),
- la casilla que está arriba y a la derecha (si la ficha no está en la columna extrema derecha).

Cada casilla tiene asociado un número entero c_{ij} ($i, j = 1, \dots, n$) que indica el puntaje a asignar cuando la ficha esté en la casilla. El puntaje final se obtiene sumando el puntaje de todas las casillas recorridas por la ficha, incluyendo las de las filas superior e inferior.

Determinar el máximo y el mínimo puntaje que se puede obtener en el juego.

2. Sos la única programadora de una flamante empresa que provee desarrollo en distintos proyectos. Tenés n proyectos posibles a los cuales ofrecer servicio y la posibilidad de trabajar H horas como máximo. Para cada proyecto $i \in \{1..n\}$ ya calculaste la cantidad de horas h_i que requiere de trabajo, y la paga p_i que recibirás si lo hacés. Tenés la posibilidad de pedirle a un amigo que te ayude con algunos proyectos, en cuyo caso te va a tomar la mitad de las horas (división entera) realizarlo, pero vas a cobrar la mitad del dinero (ya que la otra mitad se la darás a tu amigo). Tu tarea es calcular la máxima ganancia que podés obtener eligiendo qué proyectos tomar y cuándo recurrir a la ayuda de tu amigo.
 - (a) (Backtracking) Resolvé el problema utilizando la técnica de backtracking dando una función recursiva. Para ello:
 - Especificá precisamente qué calcula la función recursiva que resolverá el problema, indicando qué argumentos toma y la utilidad de cada uno.
 - Da la llamada o la expresión principal que resuelve el problema.
 - Definí la función en notación matemática.
 - (b) (Programación dinámica) Implementá un algoritmo que utilice Programación Dinámica para resolver el problema.
 - ¿Qué dimensiones tiene la tabla que el algoritmo debe llenar?
 - ¿En qué orden se llena la misma?
 - ¿Se podría llenar de otra forma? En caso afirmativo indique cuál.

1. (Backtracking) No es posible correr una carrera de 800 vueltas sin reemplazar cada tanto las cubiertas (las ruedas) del auto. Como los mecánicos trabajan en equipo, cuando se cambian las cubiertas se reemplazan simultáneamente las cuatro. Reemplazar el set de cuatro cubiertas insume un tiempo T fijo, totalmente independiente de cuál sea la calidad de las cubiertas involucradas. Hay diferentes sets de cubiertas: algunas permiten mayor velocidad que otras, y algunas tienen mayor vida útil que otras, es decir, permiten realizar un mayor número de vueltas. Sabiendo que se cuenta con n sets de cubiertas, que t_1, t_2, \dots, t_n son los **tiempos por vuelta** que pueden obtenerse con cada uno de ellos, y que v_1, v_2, \dots, v_n es la vida útil medida en **cantidad de vueltas** de cada uno de ellos, se pide encontrar el tiempo de carrera mínimo cuando la misma consta de m vueltas.
 - (a) (Backtracking) Resolvé el problema utilizando la técnica de backtracking dando una función recursiva. Para ello:
 - Especificá precisamente qué calcula la función recursiva que resolverá el problema, indicando qué argumentos toma y la utilidad de cada uno.
 - Da la llamada o la expresión principal que resuelve el problema.
 - Definí la función en notación matemática.
 - (b) (Programación dinámica) Implementá un algoritmo que utilice Programación Dinámica para resolver el problema.
 - ¿Qué dimensiones tiene la tabla que el algoritmo debe llenar?
 - ¿En qué orden se llena la misma?
 - ¿Se podría llenar de otra forma? En caso afirmativo indique cuál.

2. Te encontrás frente a una máquina expendedora de café que tiene un letrero que indica claramente que la máquina “no da vuelto”. Buscás en tu bolsillo y encontrás exactamente n monedas, con las siguientes denominaciones enteras positivas: d_1, d_2, \dots, d_n . Una rápida cuenta te transmite tranquilidad: te alcanza para el ansiado café, que cuesta C . Teniendo en cuenta que la máquina no da vuelto, dar un algoritmo que determine el menor monto posible que sea mayor o igual al precio del café.

(a) (Backtracking) Resolvé el problema utilizando la técnica de backtracking dando una función recursiva. Para ello:

- Especificá precisamente qué calcula la función recursiva que resolverá el problema, indicando qué argumentos toma y la utilidad de cada uno.
- Da la llamada o la expresión principal que resuelve el problema.
- Definí la función en notación matemática.

(b) (Programación dinámica) Implementá un algoritmo que utilice Programación Dinámica para resolver el problema.

- ¿Qué dimensiones tiene la tabla que el algoritmo debe llenar?
- ¿En qué orden se llena la misma?
- ¿Se podría llenar de otra forma? En caso afirmativo indique cuál.

1. (Backtracking) En el piso 17 de un edificio que cuenta con n oficinas iguales dispuestas de manera alineada una al lado de la otra, se quieren pintar las mismas de modo tal que no haya dos oficinas contiguas que resulten pintadas con el mismo color. Se dispone de 3 colores diferentes cuyo costo por oficina es C_1 , C_2 y C_3 respectivamente. Para cada oficina i , el oficinista ha expresado su preferencia por cada uno de los tres colores dando tres números p_1^i , p_2^i y p_3^i , un número más alto indica mayor preferencia por ese color. Escribir un algoritmo que utilice la técnica de backtracking para obtener el máximo valor posible de (sumatoria para i desde 1 a n , de $p_{j_i}^i / C_{j_i}$, es decir, que maximice $\sum_{i=1}^n p_{j_i}^i / C_{j_i}$), sin utilizar nunca el mismo color para dos oficinas contiguas.

Antes de dar la solución, especificá con tus palabras qué calcula la función recursiva que resolverá el problema, detallando el rol de los argumentos y la llamada principal.

2. (Backtracking) Te vas de viaje a la montaña viajando en auto k horas hasta la base de un cerro, donde luego caminarás hasta el destino de tus vacaciones. Tu auto no es muy nuevo, y tiene un stereo que solo reproduce cds (compact-disks). Buscás en tu vasta colección que compraste en los años 90 y tenés p cds, con $p > k$, que duran exactamente una hora cada uno. Encontrás también un cuaderno donde le diste una puntuación entre 1 y 10 a cada cd de tu colección. Cuanto mayor la puntuación, más es el placer que te da escucharlo. Dado que no sos tan exigente, querés que el puntaje promedio entre dos discos consecutivos que escuches, no sea menor a 6. Así por ejemplo si en la hora 2 escuchás un cd que tiene puntaje 8, en la hora 3 podrías escuchar uno que tenga puntaje al menos 4.

Encontrar una combinación de cds para escuchar en las k horas de viaje, cumpliendo la restricción de que en dos horas consecutivas el puntaje promedio de los dos discos sea mayor o igual a 6, maximizando el puntaje total de los k discos que escucharás.

Se pide lo siguiente:

- (a) Especificá precisamente qué calcula la función recursiva que resolverá el problema, indicando qué argumentos toma y la utilidad de cada uno.
- (b) Da la llamada o la expresión principal que resuelve el problema.
- (c) Definí la función en notación matemática.

2. (Backtracking) Luego de que te dan el alta por intoxicación, Malena te pide de nuevo que le cuides el departamento. Esta vez te deja N productos que no vencen pero los tenés que pagar. Cada producto i tiene un precio p_i y un valor nutricional s_i . Tu presupuesto es M . Se pide comer productos para obtener el máximo valor nutricional sin superar el presupuesto M . No hace falta comer todos los días ni vaciar la heladera.

- (a) Especificá precisamente qué calcula la función recursiva que resolverá el problema, indicando qué argumentos toma y la utilidad de cada uno.
- (b) Da la llamada o la expresión principal que resuelve el problema.
- (c) Definí la función en notación matemática.

