Práctica 3 - Programación dinámica

NOTAS PRELIMINARES
Los objetivos de esta práctica son:

- Introducir la técnica de programación dinámica.
- Identificar los pasos requeridos para alcanzar una solución por programación dinámica.
- Desarrollar optimizaciones para alcanzar una mayor eficiencia de los algoritmos.

Para cada uno de los ejercicios, se debe indicar claramente la función recursiva usada, el esquema de llenado de la memoria auxiliar, el algoritmo que encuentra el valor buscado y, si corresponde, una solución concreta para esa valor óptimo. Además se debe indicar y justificar la complejidad temporal del algoritmo propuesto.
Los ejercicios marcados con el símbolo * constituyen un subconjunto mínimo de ejercitación. Sin embargo, aconsejamos fuertemente hacer todos los ejercicios.

Ejercicio 1 *

La complejidad espacial[1] del algoritmo de programación dinámica visto en la teórica para calcular el número de Fibonacci es lineal. Cómo podría mejorarse?

[1] *La memoria consumida por variables usadas en forma intermedia para producir la salida.*

Ejercicio 2 *

El coeficiente binomial (o número combinatorio) $\binom{n}{m}$ representa la cantidad de subconjuntos de m elementos que se pueden tomar de un conjunto de n. Se define como $\binom{n}{m} = \frac{n!}{m!(n-m)!}$ y una propiedad (fácil de verificar usando la definición) indica que para $n \geq 2$ y $m \geq 1$, $\binom{n}{m} = \binom{n-1}{m-1} + \binom{n-1}{m}$

a. Escribir un algoritmo para calcular los coeficientes binomiales que use esta propiedad recursiva sin repetir cálculos. *Ayuda: Indique claramente los casos base de la recursión.*
b. Determinar la complejidad temporal y espacial para el algoritmo de **a** y para un algoritmo que calcule el coeficiente binomial por su definición.
c. Revisar el ejercicio 1 y ver que se puede realizar una optimización de espacio, aunque sólo si se elige una estrategia *bottom-up*.

Ejercicio 3 *

Sea $M \in \mathbb{N}^{m \times n}$ una matriz de números naturales. Se desea obtener un camino que empiece en la casilla superior izquierda ($[1,1]$), termine en la casilla inferior derecha ($[m,n]$), y tal que minimice la suma de los valores de las casillas por las que pasa. En cada casilla $[i,j]$ hay dos movimientos posibles:

- ir hacia abajo (a la casilla $[i+1,j]$), o
- ir hacia la derecha (a la casilla $[i,j+1]$)

a. Diseñar un algoritmo eficiente basado en programación dinámica que resuelva este problema.
b. Determinar la complejidad del algoritmo propuesto (temporal y espacial).
c. Exhibir el comportamiento del algoritmo sobre la matriz que aparece a continuación.

$$\begin{bmatrix} 2 & 8 & 3 & 4 \\ 5 & 3 & 4 & 5 \\ 1 & 2 & 2 & 1 \\ 3 & 4 & 6 & 5 \end{bmatrix}$$

Ejercicio 4 *

Sea $v = (v_1, v_2, \ldots v_n)$ un vector de números naturales, y sea $w \in \mathbb{N}$. Se desea intercalar entre los elementos de v las operaciones $+$ (suma), $\times$ (multiplicación) y $\uparrow$ (potenciación) de tal manera que al evaluar la expresión obtenida el resultado sea w.
Para evaluar la expresión se opera de izquierda a derecha ignorando la precedencia de los operadores.
Por ejemplo, si $v = (3, 1, 5, 2, 1)$ y las operaciones elegidas son $+$, $\times$, $\uparrow$ y $\times$ (en ese orden), la expresión obtenida es $3 + 1 \times 5 \uparrow 2 \times 1$, que se evalua como $(((3 + 1) \times 5) \uparrow 2) \times 1 = 400$.

a. Diseñar un algoritmo basado en programación dinámica que dados v y w, encuentre una secuencia de operaciones como la deseada, en caso de que tal secuencia exista.
b. Determinar la complejidad del algoritmo propuesto (temporal y espacial). Se puede usar la estrategia del ejercicio 2?

Ejercicio 5 *

Una fábrica electrónica tiene un contrato para entregar las siguientes cantidades de radios en los próximos 3 meses:

- mes 1, 200 radios
- mes 2, 300 radios
- mes 3, 300 radios.

Cada radio fabricada en los meses 1 y 2 cuesta \$10 y si es fabricada en el mes 3, \$12. El costo de stock es de \$1,50 por mes. El costo fijo de iniciar la producción de radios un mes dado es \$250. Las radios fabricadas durante un mes dado pueden ser entregadas ese mismo mes o después. La producción de cada mes debe ser un número de radios mltiplo de 100. El stock inicial es 0.

- Determinar el plan de producción óptimo.
- Generalizar la solución para n meses con una producción requerida de p_i y un costo de c_i para el mes $i \in \{1 \dots n\}$. Recomendamos considerar $P = \sum_{i=1}^{n} p_i$.\

Ayuda: utilizar como función recursiva una que, dado un mes i y un stock j, determine la forma óptima de dejar j unidades al final de i (por lo tanto j debe ser al menos igual a la demanda para i), cumpliendo con las demandas de todos los meses anteriores.

Ejercicio 6

Dar un algoritmo de programación dinámica para el problema de dar el vuelto con el menor número de billetes posible suponiendo que se cuenta con billetes de 1, 2, 5, 10, 20 y 50 pesos. Decir si funciona en todos los casos o sólo en algunos casos (infinito número de monedas disponibles, para monedas de cualquier valor, etc.). Dar la complejidad del algoritmo.

Ejercicio 7 *

Sean u y v dos string de caracteres. Queremos transformar u en v con el menor número de operaciones posibles de alguno de los siguientes tipos:

- borrar un caracter
- agregar un caracter
- cambiar un caracter

Por ejemplo uno puede transformar $u = abbac$ en $v = abcbc$ en tres pasos, (borrar b del segundo lugar de u, agregar b en el penltimo lugar, cambiar la a que est en el tercer lugar a c).
Es esta forma de transformar u en v óptima?
Escribir un algoritmo de programación dinámica que encuentre el mínimo número de operaciones necesarias para transformar u en v e informe cuales son las operaciones necesarias. Cuál es la complejidad del algoritmo en función de las longitudes de u y v?