

## Algoritmos y Estructuras de Datos II

### Trabajo Entregable 2

1. (Voraz) Finalmente tenés la posibilidad de irte  $N$  días (con sus respectivas noches) de viaje y en el recorrido que armaste, cada día/noche  $i$  estarás en una ciudad  $C_i$ . Contás con  $M$  pesos en total de presupuesto para gastar en alojamiento y para cada ciudad conocés el costo  $k_i$  por noche del único hotel que tiene. Cada noche  $i$  podés elegir entre dormir en el hotel de la ciudad, lo que te costará  $k_i$ , o dormir en una carpa que llevaste, que te cuesta 0. Se pide indicar en qué ciudades dormirás en hotel, de manera tal que el monto total gastado en alojamiento en los  $N$  días no supere el presupuesto  $M$ , minimizando la cantidad de noches que dormís en carpa. Para ello:
  - (a) Indicá de manera simple y concreta, cuál es el criterio de selección voraz para construir la solución.
  - (b) Indicá qué estructuras de datos utilizarás para resolver el problema.
  - (c) Explicá en palabras cómo el algoritmo resolverá el problema.
  - (d) Implementá el algoritmo en el lenguaje de la materia de manera precisa.
2. (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.
3. (Programación dinámica) Escribí un algoritmo que utilice Programación Dinámica para resolver el ejercicio del punto anterior.
  - (a) ¿Qué dimensiones tiene la tabla que el algoritmo debe llenar?
  - (b) ¿En qué orden se llena la misma?
  - (c) ¿Se podría llenar de otra forma? En caso afirmativo indique cuál.