

Algoritmos y Estructuras de Datos II – 29 de Julio de 2021
Examen Final Teórico-Práctico

Alumno: Email:

Siempre se debe explicar la solución. Una respuesta correcta no es suficiente si no viene acompañada de una justificación lo más clara y completa posible. Los algoritmos no deben escribirse utilizando código c o de bajo nivel, sino el código de la materia y evitando la utilización innecesaria de punteros. La no observación de estas recomendaciones resta puntaje.

1. (Algoritmos voraces) Es viernes a las 18 y usted tiene ganas de tomar limonada con sus amigos. Hay n bares cerca, donde cada bar i tiene un precio P_i de la pinta de limonada y un horario de happy hour H_i , medido en horas a partir de las 18 (por ejemplo, si el happy hour del bar i es hasta las 19, entonces $H_i = 1$), en el cual la pinta costará un 50% menos. Usted toma una cantidad fija de 2 pintas por hora y no se considera el tiempo de moverse de un bar a otro. Escribir un algoritmo que obtenga el menor dinero posible que usted puede gastar para tomar limonada desde las 18 hasta las 02 am (es decir que usted tomará 16 pintas) eligiendo en cada hora el bar que más le convenga.

Se pide lo siguiente:

- (a) Indicar de manera simple y concreta, cuál es el criterio de selección voraz para construir la solución?
 - (b) Indicar qué estructuras de datos utilizarás para resolver el problema.
 - (c) Explicar en palabras cómo resolverá el problema el algoritmo.
 - (d) Implementar el algoritmo en el lenguaje de la materia de manera precisa.
2. Te encontrarás 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.