

Algoritmos y Estructuras de Datos II – 7 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. (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. (Algoritmos voraces) Llegó el final del cuatrimestre y tenés la posibilidad de rendir algunas de las materias  $1, \dots, n$ . Para cada materia  $i$  conocés la fecha de examen  $f_i$ , y la cantidad de días inmediatamente previos  $d_i$  que necesitás estudiarla de manera exclusiva (o sea, no podés estudiar dos materias al mismo tiempo). Dar un algoritmo voraz que obtenga la mayor cantidad de materias que podés rendir.

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.