

CI-0116 Análisis de Algoritmos y Estructuras de Datos
II ciclo de 2018

III EXAMEN PARCIAL

Lunes 12 de noviembre

Nombre: _____ Carné: _____

El examen consta de 3 preguntas que suman de 105 a 135 puntos, pero no se reconocen más de 110 (10 % extra). Las preguntas se pueden responder en cualquier orden, pero se debe indicar en el cuadro mostrado abajo los números de página del cuaderno de examen en las que está cada respuesta. Para este efecto, las hojas del cuaderno de examen se deben numerar en la esquina superior externa de la página. El examen se puede realizar con lápiz o lapicero. En ambos casos tendrá derecho a reclamo. No se permite el uso de dispositivos electrónicos (calculadoras, teléfonos, audífonos, etc.).

Pregunta	Puntos (mín.)	Páginas	Calificación
1. <i>Búsqueda exhaustiva</i>	50		
2. <i>Programación dinámica</i>	50		
3. <i>Algoritmos ávidos</i>	5		
Total	105		

En el pueblo de Dnomder, los ciudadanos hablan el idioma wingdingués. Cada palabra en este idioma es una secuencia de símbolos denominados *wingdings*. El idioma tiene la particularidad de que, a diferencia del nuestro, no usa espacios entre palabras, ni signos de puntuación, ni división de palabra al final del renglón. Un ejemplo de texto en wingdingués es el siguiente (solo un dnomderiense podría saber dónde empieza y termina cada palabra):



Su tarea es la siguiente. Dado un texto en wingdingués (una secuencia de palabras en el idioma) y una longitud máxima de renglón m (en el texto anterior $m \geq 12$), debe encontrar una forma de distribuir las palabras en los renglones de forma tal que se aproveche el espacio al máximo, es decir, que la suma de la cantidad de espacios sobrantes a lo largo de todos los renglones ocupados sea mínima. Asuma que m es al menos tan grande como cualquier palabra en wingdingués. Si lo prefiere, puede asumir que, en vez del texto, la entrada al algoritmo es un vector $L = \langle l_0, l_1, \dots, l_n \rangle$ que indica la longitud de cada una de las $n + 1$ palabras del texto (y, por supuesto, la longitud máxima de renglón m).

1. *Búsqueda exhaustiva*. [50 pts.]

- Encuentre una representación vectorial para la solución del problema. Indique el significado *y el dominio* de cada una de las entradas. [5 pts.]
- Determine el espacio E al que pertenece el vector. Defina los conjuntos usados. [5 pts.]
- Determine la cardinalidad de E . [5 pts.]

- d) Si es posible, acote E mediante de una restricción del tipo $E' = \{\sigma \in E : \dots\}$, sino explique por qué es imposible. [5 pts.]
- e) Determine la cardinalidad de E' o acótela de la forma más ajustada posible. [5 pts.]
- f) Escriba un algoritmo que explore todas las soluciones candidatas en E o E' para hallar una solución óptima. Indique el significado de los argumentos de el (los) métodos(s) definidos. [20 pts.]
- g) Calcule una cota asintótica para el tiempo de ejecución del algoritmo. [5 pts.]

2. *Programación dinámica.* [50 pts.]

- a) Formule una solución al problema por medio de un oráculo. Describa su significado en términos de sus argumentos [5 pts.]. Identifique el oráculo meta [5 pts.] . Escriba los pasos base [5 pts.] y recursivo [10 pts.] que permiten calcular el oráculo completo.
- b) Con base en el punto anterior, escriba un algoritmo que resuelva el problema. [20 pts.]
- c) Calcule una cota asintótica para el tiempo de ejecución del algoritmo. [5 pts.]

3. *Algoritmos ávidos.* [5–35 pts.]

- a) Determine si el problema se puede resolver mediante un algoritmo ávido. Explique su respuesta. [5 pts.]
- b) Si su respuesta al punto anterior fue afirmativa:
 - I. Escriba el algoritmo ávido que resuelve el problema. [10 pts.]
 - II. Utilice el *método de la transformación* para demostrar su correctitud. [15 pts.]
 - III. Determine una cota asintótica ajustada para el tiempo de ejecución del algoritmo. [5 pts.]