

**Algoritmia Avanzada**

**Nombre:** \_\_\_\_\_

**Junio 2016**

**Parcial 1**

**7/6/2016**

**Tiempo Limite: 120 Minutes**

---

Este examen consta de 8 páginas (incluida esta página inicial) y 8 preguntas.  
El total es 50.

Cada 10 puntos equivalen a 1 de la nota final de el parcial, es decir 50 puntos de 50 es el equivalente a 5.

Tabla de notas (solo para el uso de el profesor)

Question	Points	Score
1	10	
2	10	
3	5	
4	5	
5	5	
6	5	
7	5	
8	5	
Total:	50	

---

1. (10 points) Considere la siguiente recurrencia.  $T(n) = 2T(\sqrt{n}) + 3n$ .
  - (a) (5 points) Por el método iterativo cual es el valor de la recurrencia y la complejidad  $O()$  (Big-oh).
  - (b) (5 points) Por el método de el teorema maestro cual es el valor de la recurrencia y la complejidad  $O()$  (Big-oh).

2. (10 points) Considere la siguiente función.

```
function CONUNDRUM( $n$ )  
   $r \leftarrow 0$   
  for  $i \leftarrow 1$  to  $n$  do  
    for  $j \leftarrow i + 1$  to  $n$  do  
      for  $k \leftarrow i + j - 1$  to  $n$  do  
         $r \leftarrow r + 1$   
      end for  
    end for  
  end for  
end function
```

- (a) (5 points) Calcular el valor de la anterior función en terminos de  $n$ .
- (b) (5 points) Cuál es la complejidad anterior función (Big-oh)?

3. (5 points) Describa qué son los árboles rojo y negros, la complejidad de las operaciones de búsqueda, inserción y borrado. Cómo se balancean, sus usos prácticos y propiedades.

4. (5 points) La mediana de una lista finita de números se puede encontrar mediante la disposición de todos los números enteros de menor a mayor valor y seleccionando el del medio. Por ejemplo, la mediana de  $\{3, 3, 5, 9, 11\}$  es 5. Si hay un número par de elementos, entonces no hay ningún valor medio. Y en este caso la mediana define como la media de los dos valores medios. Por ejemplo, la mediana de  $\{3, 5, 7, 9\}$  es  $\frac{5+7}{2} = 6$ .

Si estuviéramos leyendo números de un flujo de datos (stream o entrada estándar), defina un algoritmo para encontrar la mediana después de leer cada elemento.

5. (5 points) Diseñe una estructura de datos en la que la búsqueda, inserción y borrado sean procesadas en  $O(1)$ .

6. (5 points) Un problema común que resuelven los compiladores y los editores de texto, es determinar cuándo los paréntesis de una *string* se encuentran balanceados y apropiadamente anidados. Por ejemplo  $((()))()$  se encuentra bien balanceado y anidado, pero  $)()()$  no se encuentra propiamente anidado.
- (a) (4 points) Escriba un algoritmo que retorne *true* si la *string* es balanceada y anidada propiamente, o *false* dado el caso contrario. Su algoritmo debe ser  $O(n)$ . Donde  $n$  es el tamaño de la *string*
  - (b) (1 point) Cuando el algoritmo retorne *false* adicionalmente debe retornar la posición de el primer paréntesis inválido.

7. (5 points) Dado un número  $N$ . Usted puede hacer una de las dos siguientes operaciones.

- Si  $N = a \times b$  ( $a \neq 1, b \neq 1$ ), Podemos cambiar el valor de  $N$  a  $\max(a, b)$
- O siempre existe la opción de cambiar  $N$  a  $N - 1$ .

Encuentre el algoritmo que de la menor cantidad de pasos para llevar  $N$  a 0.

La complejidad máxima debe ser de  $O(n \lg n)$ .

8. (5 points) La altura de un árbol binario es el número de aristas (saltos) entre la raíz del árbol y su hoja más alejada. Esto significa que un árbol que contiene un único nodo tiene una altura de 0.

Con el nodo raíz, encontrar la altura del árbol.

Ejemplo de clase Nodo:

```
class Nodo
    int value;
    Nodo izquierda;
    Nodo derecha;
```