



UNIVERSIDAD DE COSTA RICA
FACULTAD DE INGENIERÍA
ESC. DE CC. DE LA COMP. E INFORMÁTICA



CI-1221 Estructuras de Datos y Análisis de Algoritmos
II ciclo 2011, grupo 02

I EXAMEN PARCIAL

Lunes 12 de setiembre

Nombre: _____ Carné: _____

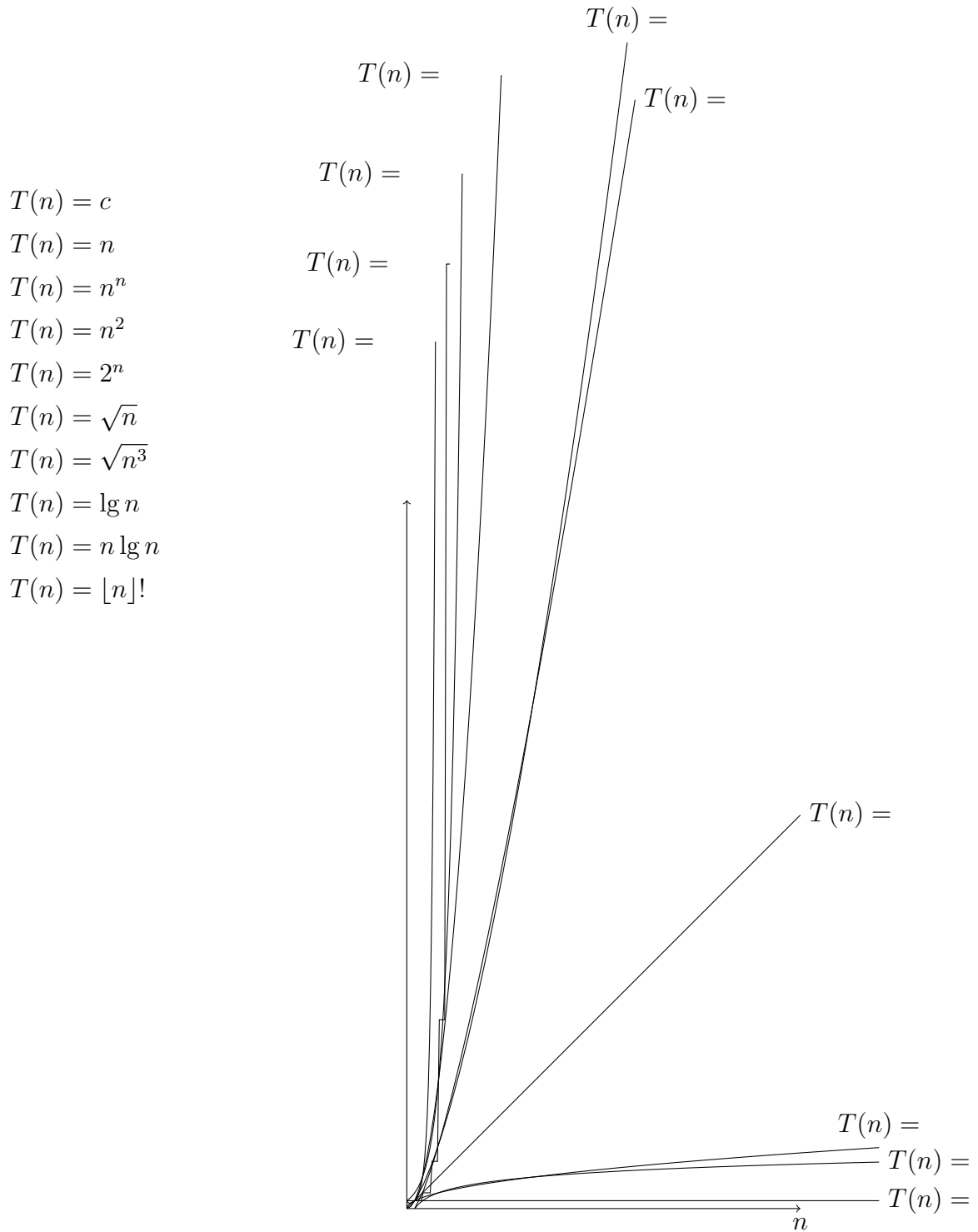
El examen consta de 5 preguntas que suman 114 puntos, pero no se reconocerán más de 110 (10 % extra). Cada pregunta indica el tema tratado y su valor. Si la pregunta tiene subítemes, el puntaje de cada uno de ellos es indicado dentro de los subítemes. Se recomienda echar un vistazo a los temas de las preguntas y a su puntaje antes de resolver el examen, para así distribuir su tiempo y esfuerzo de la mejor manera. Las preguntas se pueden responder en cualquier orden, pero se debe indicar en la tabla mostrada abajo los números de página del cuaderno de examen en las que están las respuestas. Para ello las hojas del cuaderno de examen deben estar numeradas en la esquina superior externa de cada página.

El examen se puede realizar con lápiz o lapicero. *No se permite el uso de dispositivos electrónicos (calculadoras, teléfonos, audífonos, etc.).*

Pregunta	Puntos (mín.)	Páginas	Calificación
1. <i>Matemática básica</i>	10		
2. <i>Algoritmos iterativos</i>	16		
3. <i>Algoritmos recursivos</i>	10		
4. <i>Solución de recurrencias</i>	35		
5. <i>Simulación de ejecución de algoritmos</i>	43		
Total	114		

1. *Matemática básica.* [10 pts.]

Identifique en el gráfico de la derecha cada una de las funciones de la izquierda.



2. *Algoritmos iterativos.* [16 pts.]

Considere el siguiente problema:

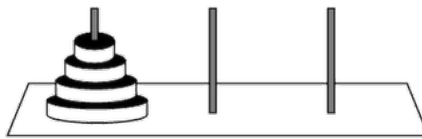
Entrada: Una secuencia de números $A = \langle a_1, a_2, \dots, a_n \rangle$ y un valor v .

Salida: Un índice i tal que $v = A[i]$ o el valor especial NIL si v no está en A .

- Escriba el pseudocódigo del algoritmo BÚSQUEDA-LINEAL, el cual resuelve el problema iterando sobre los elementos de A y retornando el índice en donde se encuentra v si está en el arreglo, y NIL si no está. [3 pts.]
- Demuestre que el algoritmo es correcto. Para ello identifique un invariante apropiado para el ciclo [3 pts.], y muestre como los pasos de inicialización [2 pts.], mantenimiento [3 pts.] y terminación [2 pts.] permiten inferir la correctitud del algoritmo.
- Encuentre una cota asintótica ajustada para el *tiempo* de ejecución del algoritmo en el mejor [1 pto.] y peor [1 pto.] casos.
- Encuentre una cota asintótica ajustada para el *espacio* extra requerido por el algoritmo. [1 pto.]

3. *Algoritmos recursivos.* [10 pts.]

El juego de las torres de Hanoi consiste en mover n discos de una estaca a otra, utilizando como pivote una tercera estaca. Los discos se deben mover uno a la vez, sin colocar discos encima de discos más pequeños. Los discos se encuentran originalmente apilados del más grande al más pequeño en una de las estacas, como se muestra en la ilustración para $n = 4$.



Una forma de resolver el problema es seguir las instrucciones desplegadas por el siguiente algoritmo:

```
Hanoi( n, aquí, allá, pivote ) {  
    si n = 1  
        despliegue: "Mover disco de " aquí " a " allá  
    sino  
        Hanoi( n - 1, aquí, pivote, allá );  
        despliegue: "Mover disco de " aquí " a " allá  
        Hanoi( n - 1, pivote, allá, aquí );  
    }  
}
```

- Escriba una fórmula recursiva para el tiempo de ejecución del algoritmo. [3 pts.]
- Determine una cota asintótica para el tiempo de ejecución del algoritmo. [7 pts.]

4. *Solución de recurrencias.* [35 pts.]

Resuelva las siguientes recurrencias asumiendo que $T(n) = \Theta(1)$ para $n \leq 1$ y $c > 0$.

a) $T(n) = 2T(0,5n) + cn^2$. [4 pts.]

b) $T(n) = 2T(n/2) + n + \lg n$. [5 pts.]

c) $T(n) = 2T(\sqrt{2}n/2) + cn$. [6 pts.]

d) $T(n) = 4T(n/2) + (n \lg n)^2$. [20 pts.]

Observación. $1 + 2^2 + 3^2 + \dots + n^2 = \frac{n(n+1)(2n+1)}{6}$

5. *Simulación de ejecución de algoritmos.* [43 pts.]

Considere los siguientes algoritmos:

a) Ordenamiento por selección.

b) Ordenamiento por inserción.

c) Ordenamiento por mezcla.

d) Ordenamiento usando montículos.

e) Ordenamiento rápido (usando el último elemento como pivote).

f) Ordenamiento por residuos (usando la base óptima, de acuerdo al tamaño del arreglo).

Para cada uno de ellos:

- I. Indique su tiempo asintótico de ejecución en el peor caso, de la forma más precisa posible. [1 pto.]
- II. Explique qué estructura debe poseer un arreglo para llevar el algoritmo al peor caso, en un sentido asintótico. Si la duración asintótica es siempre la misma, independientemente del arreglo, indique que es así. [$1\frac{1}{2}$ pts.]
- III. Escoja un arreglo de tamaño 4 cuya estructura corresponda a la del peor caso y simule la ejecución del algoritmo sobre ese arreglo, mostrando el (los) estado(s) de el (los) (sub)arreglo(s) después de cada “paso” (iteración del ciclo principal o llamado a [sub]función del algoritmo). Después del primer paso incorrecto el resto de pasos no suman puntos. [4 pts. c/ simulación, excepto ordenamiento por montículos: 4 pts. por monticularizar y 4 pts. por ordenar].