El examen es individual y no es permitido el uso de libros, apuntes ni equipos electrónicos.
Nombre y código:

Pregunta	Invariantes	Dividir y conquistar	Orden asintótico y el Teorema Maestro	Programación dinámica	Total
Puntos	15	15	10	10	50
Bono	0	10	0	0	10
Puntaje					

Pregunta 1: Invariantes . . . . (15 puntos)

Considere el siguiente problema:

Entrada: un número real x y un arreglo de números reales a[0::N), N 0.

Salida: el valor del polinomio $P(x) = (+i \mid 0 \leq i < N : a[i]\, x^i)$.

El siguiente fragmento de código Python implementa la regla de Horner para evaluar P(x):

```
1 def  P(x,  a):
2     N  =  len(a)
3     r,n  =  0,N
4     while  n >0:
5              r  =  a[n -1]+ x*r
6              n  =  n -1
7     return  r
```

(a) (3 puntos) En términos de notación O, ¿cuál es el tiempo de ejecución de la implementación de la regla de Horner? Puede suponer que las operaciones aritméticas y de asignación toman tiempo O(1).

(b) (12 puntos) Considere los siguientes invariantes para el ciclo while en las líneas 4-6

Q_0**:** $0 \leq n \leq N$

Q_1**:** $r = (+i \mid 0 \leq i < N - n : a[i]\, x^i)$

Demuestre las propiedades de iniciación, estabilidad y terminación para los invariantes Q_0 y Q_1. Concluya que, al momento de terminar, la función P(x, a) calcula el polinomio P(x).

Pregunta 2: Dividir y conquistar . . . . (15 puntos)

Considere un arreglo A[0..n), n ≥ 0, de números enteros. Un segmento constante de A es un subarreglo A[I, j), con 0 ≤ i < j ≤ n, tal que

A[i] = A[i + 1] = . . . = A[j-1]

(a) (2 puntos) De un ejemplo de una arreglo que tenga segmentos constantes de longitud 3 y 5.

(b) (3 puntos) Especifique el problema de determinar la longitud máxima de un segmento constante en un arreglo.

(c) (10 puntos) Usando la técnica de "dividir y conquistar", diseñe un algoritmo que resuelva el problema especificado en el numeral anterior en orden temporal de O(n log n) (esto debe demostrarlo).

(d) (10 +) Usando la técnica de "programación dinámica", diseñe un algoritmo que resuelve el problema especificado anteriormente en orden temporal de O(n).

Pregunta 3: Orden asintótico y el Teorema Maestro . . . . (10 puntos)

(a) (2 puntos) Sea f una función de los números naturales en los reales positivos. Defina el conjunto O(f).
(b) (2 puntos) Enuncie el Teorema Maestro
(c) (3 puntos) Clasifique ascendentemente por orden asintótico las siguientes funciones:
(d) (3 puntos) Use el teormea Maestro para calcular T(n) en cada uno de los siguientes casos:

i. $T(n) = T(n/2) + \sqrt{n}$
ii. $T(n) = 7T(n/8) + 1$
iii. $T(n) = 2T(5n/6) + n^2$

Pregunta 4: Programación dinámica . . . . (10 puntos)

Dados dos arreglos, digamos A[0..M) y B[0..N), una subsecuencia común de A y B es una subsecuencia de A que es también subsecuencia de B (o viceversa).
Considere el siguiente problema:

Entrada: dos arreglos A[0..M) y B[0..N), con $M \geq 0$ y $N \geq 0$.

Salida: el máximo entre las longitudes de las subsecuencias comunes de A[0..M) y B[0..N).

También considere la siguiente función, con $0 \leq m \leq M$ y $0 \leq n \leq N$:

(m;n) : “el máximo entre las longitudes de las subsecuencias comunes de A[0..m) y B[0..n).”

definida recurrentemente de la siguiente manera:

$$\phi(m,n) = \begin{cases} 0 & \text{, si } m = 0 \vee n = 0 \\ 1 + \phi(m-1, n-1) & \text{, si } m > 0 \wedge n > 0 \wedge A[m-1] = B[n-1] \\ \phi(m, n-1) \uparrow \phi(m-1, n) & \text{, si } m > 0 \wedge n > 0 \wedge A[m-1] \neq B[n-1] \end{cases}$$

(a) (2 puntos) Escriba el objetivo del problema planteado inicialmente en términos de
(b) (2 puntos) Aplique la función , con los parámetros adecuados, para calcular la máxima longitud de una subsecuencia común para A = [1;2;1;4] y B = [1;4;2].
(c) (6 puntos) Diseñe una tabulación para la función y, usando pseudo-código, diseñe un algoritmo que procese la tabulación.