

Nombre: _____

Matemáticas para la Computación

(10) 1. Determina si la función $f(x) = x^3$ definida en $f: \mathbb{R} \rightarrow \mathbb{R}$ es biyectiva.

2. Demostrar que:

(15) a) La suma de los primeros n impares positivos es igual a n^2 ; es decir $1+3+5+\dots+(2n-1)=n^2$

(15) b) Dadas las proposiciones p, q, r : $[(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$ es una tautología.

(20) 3. Con los números 1, 2, 3, 4, 5 y 6.

a) ¿Cuántos números de cuatro cifras distintas se pueden formar?

b) ¿Cuántos empiezan por 1?

c) ¿Cuántos acaban en 24?

d) ¿En cuántos está el número 5?

e) ¿En cuántos no está el número 5?

(Puedes dejar las expresiones en términos de factoriales)

4. Los siguientes algoritmos evalúan un polinomio de la forma $a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ en $x=c$.

(10) a) Calcula la función de complejidad temporal para un tamaño de problema igual a n y las operaciones básicas de suma y multiplicación (no consideres los incrementos a las variables de ciclo).

(10) b) Determina el orden de complejidad de dichas funciones y selecciona el algoritmo más eficiente (justifica tu selección).

```
procedure polynomial( $c, a_0, a_1, \dots, a_n$ : real numbers)
  power := 1
  y :=  $a_0$ 
  for  $i := 1$  to  $n$ 
    power := power *  $c$ 
    y := y +  $a_i$  * power
  return y { $y = a_n c^n + a_{n-1} c^{n-1} + \dots + a_1 c + a_0$ }
```

```
procedure Horner( $c, a_0, a_1, a_2, \dots, a_n$ : real numbers)
  y :=  $a_n$ 
  for  $i := 1$  to  $n$ 
    y := y *  $c$  +  $a_{n-i}$ 
  return y { $y = a_n c^n + a_{n-1} c^{n-1} + \dots + a_1 c + a_0$ }
```

5. Suponga que se tienen n subconjuntos S_1, S_2, \dots, S_n del conjunto $S=\{1, 2, 3, \dots, n\}$, y en cada subconjunto hay k elementos.

(10) a) Describe un algoritmo que determine si hay un par de subconjuntos que son disjuntos (no tienen elementos en común). El algoritmo deberá comparar cada subconjunto S_i con los demás subconjuntos, y para cada par de subconjuntos (S_i, S_j) deberá checar si alguno de los k elementos de S_i está también en S_j .

(5) b) Describe las características de la instancia del mejor caso y del peor caso para el algoritmo planteado.

(5) c) Calcula el orden de complejidad del algoritmo propuesto considerando el peor caso.

Nombre: _____

Programación Orientada a Objetos

- (10)** 1. De la siguiente lista señalar los elementos que son utilizados para definir los objetos.
- Características de los objetos.
 - Comportamiento de los objetos.
 - Nombre de cada objeto.
 - Instancia de un objeto.
- (10)** 2. Determinar si las siguientes afirmaciones son verdaderas o falsas:
- “Una clase define a un grupo de objetos que tienen las mismas características pero diferente comportamiento” (V / F).
 - “El comportamiento de un objeto se implementa con los métodos de la clase” (V / F).
- (10)** 3. Menciona dos formas posibles de realizar el encapsulamiento en una Clase.
- (10)** 4. Mencionar dos criterios (anexar ejemplo para c/u) para generar una familia de clases utilizando la Herencia.
- (10)** 5. Dar un ejemplo del uso de las Excepciones, y justificar por qué el mecanismo de excepciones dará beneficios en tu ejemplo.
- (10)** 6. Proporcionar dos usos que se le pueden dar a los diagramas de Casos de Uso en el proceso de desarrollo de software; mencionar actividades y roles (analista, ingeniero de pruebas, desarrollador, líder, cliente, etc.)
- (10)** 7. ¿Cuál es la diferencia entre el patrón de diseño Factory y Singleton?, recordar que ambos pertenecen al grupo de patrones de diseño de creación.
- (10)** 8. Proporcionar un ejemplo donde se requiera el patrón de diseño Facade; justifique su uso e incluya un diagrama de clase.
- (10)** 9. Menciona tres responsabilidades que tiene el componente “Controlador”, en el patrón arquitectónico MVC.
- (10)** 10. ¿Cuál es la diferencia entre la agregación y la composición?