Práctica 10

**Ejercicio 1:** Un programa en un lenguaje procedural es una secuencia de instrucciones o

comandos que se van ejecutando y producen cambios en las celdas de memoria, a través de las

sentencias de asignación. ¿Qué es un programa escrito en un lenguaje funcional? y ¿Qué rol

cumple la computadora?  
  
Es un programa basado en el uso de funciones. Es muy popular en la resolución de problemas de inteligencia artificial, matemática, lógica y procesamiento paralelo.  
  
El rol que cumple abarca principalmente:

* Vista uniforme de programa y función
* Tratamiento de funciones como datos
* Liberación de efectos colaterales
* Manejo automático de memoria

Desventajas:

* Ineficiencia de ejecución

**Ejercicio 2:** ¿Cómo se define el lugar donde se definen las funciones en un lenguaje funcional?  
  
Se debe distinguir entre el VALOR y la DEFINICIÓN de una función.

Existen muchas maneras de DEFINIR una misma función, pero siempre dará el mismo valor, ejemplo: DOBLE X = X + X DOBLE’ X= 2 \* X

Denotan la misma función pero son dos formas distintas de definirlas

**Ejercicio 3:** ¿Cuál es el concepto de variables en los lenguajes funcionales?

La noción de Variable es la de “variable matemática”, no la de celda de memoria.

**Ejercicio 4:** ¿Qué es una expresión en un lenguaje funcional? ¿Su valor de qué depende?

La expresión es la noción central de la programación Funcional

Característica más importante: “Una expresión es su VALOR”

El valor de una expresión depende ÚNICAMENTE de los valores de las sub expresiones que la componen.

Las expresiones también pueden contener VARIABLES, (valores desconocidos)

**Ejercicio 5:** ¿Cuál es la forma de evaluación que utilizan los lenguajes funcionales?

La forma de evaluar es a través de un mecanismo de REDUCCIÓN o SIMPLIFICACIÓN  
cuadrado (3 + 4)

=> cuadrado 7 (+)  
=> 7 \* 7 (cuadrado)  
=> 49 (\*)

Otra forma sería: cuadrado (3 + 4) => (3 + 4) \* (3 + 4) (cuadrado) => 7 \* (3 + 4) (+) => 7 \* 7 (+) => 49 (\*)

**Ejercicio 6:** ¿Un lenguaje funcional es fuertemente tipado? ¿Qué tipos existen? ¿Por qué?

Un lenguaje funcional no depende del tipo de datos de sus funciones   
Un lenguaje fuertemente tipado es en el cual si no se permiten violaciones de los tipos de datos, es decir, dado el valor de una variable de un tipo concreto, no se puede usar como si fuera de otro tipo distinto a menos que se haga una conversión,   
  
Tipo de una función

Puede estar definida explícitamente dentro del SCRIPT, por ejemplo:  
cuadrado::num  
num cuadrado x= x + x

O puede deducirse/inferirse el tipo de una función

**Ejercicio 7:** ¿Cómo definiría un programa escrito en POO?

Un programa escrito con una lenguaje OO es un conjunto de OBJETOS que INTERACTÚAN mandándose MENSAJES.

**Ejercicio 8:** Diga cuáles son los elementos más importantes y hable sobre ellos en la

programación orientada a objetos.

Objetos:

Son entidades que poseen estado interno y comportamiento

Es el equivalente a un dato abstracto

Mensajes:

Es una petición de un objeto a otro para que este se comporte de una determinada manera, ejecutando uno de sus métodos

TODO el procesamiento en este modelo es activado por mensajes entre objetos.

Métodos:

Es un programa que está asociado a un objeto determinado y cuya ejecución solo puede desencadenarse a través de un mensaje recibido por éste o por sus descendientes

Clases:

Es un tipo definido por el usuario que determina las estructuras de datos y las operaciones asociadas con ese tipo

Cada objeto pertenece a una clase y recibe de ella su funcionalidad

Primer nivel de abstracción de datos: definimos estructura, comportamiento y tenemos ocultamiento.

La información contenida en el objeto solo puede ser accedida por la ejecución de los métodos correspondientes

Instancia de clase:

Cada vez que se construye un objeto se está creando una INSTANCIA de esa clase

Una instancia es un objeto individualizado por los valores que tomen sus atributos

**Ejercicio 9:** La posibilidad de ocultamiento y encapsulamiento para los objetos es el primer nivel de abstracción de la POO, ¿cuál es el segundo?

El segundo nivel de abstracción consiste en agrupar las clases en jerarquías de clases (definiendo SUB y SUPER clases), de forma tal que una clase A herede todas las propiedades de su superclase B (suponiendo que tiene una)

**Ejercicio 10:** ¿Qué tipos de herencias hay? Cuál usa Smalltalk y C++

Tipos de herencia:

* Simple: Cada clase puede tener UNA sola super-clase.  
  Esquema de árbol
* Múltiple: Una clase puede heredar de varias super-clases
* Estricta: Una sub-clase NO puede redefinir en forma contradictoria atributos defindidos en la superclase.
* No estricta: Alguna sub-clase NO quiere heredar algún atributo de la super-clase.

C++: Simple, hereda de C. Incorporó características de la POO.  
Smalltalk: …

**Ejercicio 11:** En el paradigma lógico ¿Qué representa una variable? ¿y las constantes?

Es un paradigma en el cual los programas son una serie de aserciones lógicas.

El conocimiento se representa a través de reglas y hechos

Los objetos son representados por términos, los cuales contienen constantes y variables

PROLOG es el lenguaje lógico más utilizado.

* Variables:
  + Se refieren a elementos indeterminados que pueden sustituirse por cualquier otro.
  + “humano(X)”, la X puede ser sustituída por constantes como: juan, pepe, etc.
  + Los nombres de las variables comienzan con mayúsculas y pueden incluir números.
* Constantes:
  + A diferencia de las variables son elementos determinados.
  + “humano(juan)”
  + La constantes son string de letras en minúsculas (representan objetos atómicos) o string de dígitos (representan números).

**Ejercicio 12:** ¿Cómo se escribe un programa en un lenguaje lógico?

Un programa escrito en un lenguaje lógico es una secuencia de “cláusulas”.

Las cláusulas pueden ser: un “Hecho” o una “Regla”.

Hecho:

* Expresan relaciones entre objetos
* Expresan verdades
* Son expresiones del tipo p(t1,t2,….tn) Ejemplos:
* tiene(coche,ruedas)  representa el hecho que un coche tiene ruedas
* longuitud([],0)  representa el hecho que una lista vacía tiene longuitud cero
* virus(ithaqua)  representa el hecho que ithaqua es un virus.

Regla:

* Cláusula de Horn
* Tiene la forma: conclusión :- condición. Dónde:
* :- indica “Si”
* conclusión es un simple predicado y
* condición es una conjunción de predicados, separados por comas. Representan un AND lógico

**Ejercicio 13:** Teniendo en cuenta el siguiente problema, se lee una variable entera por teclado y si es par se imprime “El valor ingresado es PAR” y si es impar imprime “El valor ingresado es

impar”, implemente este ejemplo en cada uno de los paradigmas presentados en esta práctica.

**Ejercicio 14:** Describa las características más importantes de los Lenguajes Basados en Scripts.

Mencione diferentes lenguajes que utilizan este concepto. ¿En general, qué tipificación utilizan?

Un script o lenguaje de script es un lenguaje de programación para un entorno especial de ejecución que automatiza la ejecución de tareas; las tareas podrían ser ejecutadas alternativamente una por una por un operador humano. Los lenguajes de scripting son a menudo interpretados (en lugar de compilados).

Lenguajes basados en scripts:

VBScript: Implementación de Visual Basic para crear aplicaciones para Internet Explorer.

JScript: Implementación de Microsoft del lenguaje de scripting basado en Java.

JavaScript: Lenguaje de elaboración de scripts para incrustarse en archivos HTML, que es con diferencia el lenguaje de script más usado en Internet.

ActionScript: Lenguaje de script de Macromedia para la aplicación Flash.

ECMAScript: Lenguaje de scripting que soporta el estándar ECMA-262 (European Computer Manufacturers Association).

**Ejercicio 15:** ¿Existen otros paradigmas? Justifique la respuesta

Paradigmas:

* Paradigma funcional
* Paradigma procedural
* Paradigma lógico
* Paradigma orientado a objetos
* Paradigma por procedimientos o imperativo
* Paradigma declarativo