

CONOCIMIENTO DE ALGORITMOS

1. Con sus propias palabras explique las diferencias entre switch e if
2. Con sus propias palabras explique las diferentes estructuras repetitivas
3. Con sus propias palabras defina y dé un ejemplo de los siguientes términos:
 - a. Array
 - b. Vector
 - c. Matriz
 - d. Variable
 - e. Constante
 - f. Operadores Lógicos
 - g. Operadores Matemáticos

R= 1. La diferencia entre switch e if, es que if es funcional siempre y cuando condicione 2 datos, mientras que switch es una condición que estudia varios datos, por ejemplo si resolviéramos un problema de varias condiciones con if en vez de switch, tendríamos que colocar muchos if, if, if mientras que con switch esto no sería necesario.

R=2. Ciclo For: Se usa para repetir una función varias veces (finito). (for i<=0, i<=10, i++).

Ciclo While: Se usa para repetir una función o un apartado de código varias veces, While=Mientras la condición siempre se mantenga verdadera. Los bucles while deben cerrarse con un Break; ya que si esto no se cierra quedara en un bucle infinito.

DoWhile: While y Do While son ciclos, que básicamente “hacen lo mismo” aunque con la pequeña diferencia de: While repite la función siempre y cuando la condición sea verdadera, si la condición es falsa omite cualquier proceso no repite nada y sale del ciclo, Mientras que Do While primero estudia el proceso, luego estudia la condición si la condición dice que es falso, el proceso se ejecutara 1 vez, y si es verdadero las veces que se la a dado la instrucción.

A y b. Vector o Array, esto lo que es, es un conjunto de variables, un conjunto de datos que para acceder a ellos se debe tener una posición o índice, por ejemplo:

A⁰, B¹, C², D³, E⁴ ← Acá podemos ver unas variables A, B, C, D, E, este conjunto de variables tienen un sub índice (^{0,1,2,3,4}) cada sub índice, representa la posición del dato o variable y por medio de esta posición se accede al dato.

Array bidimensional, básicamente también es un conjunto de variables, pero al hablarse de bidimensional o que tiene varias dimensiones, serán varios conjuntos de variables, para acceder a una variable dentro de este tipo de array bi, se necesitan dos posiciones la posición del conjunto de variables (A) y la posición de la variable (B), ejemplo: queremos usar la variable I, M, B.

0	1	2
A⁰, B¹, C², D³, E⁴	F⁰, G¹, H², I³, J⁴	K⁰, L¹, M², N³, Ñ⁴.

Variable I: Posición A= 1, Posición B= 3 || Variable M: Posición A=2, Posición B=2

Variable B: Posición A=0, B=1.

- c. Matriz, una matriz es como un arreglo bidimensional en el que no solo podremos agregar datos o variables de forma lineal, si no también de forma bidimensional o sea arriba y abajo, estos datos también tienen unas posiciones que serán la posición de fila y de columna en la cual se encuentre el dato, recordemos como es un array

A⁰, B¹, C², D³

F⁰, G¹, H², I³

K⁰, L¹, M², N³

Una matriz con estos datos podría ser así:

	0	1	2	3
0	A	B	C	D
1	F	G	H	I
2	J	K	L	M
3	N	M	O	P

- d. Una variable es un tipo de memoria dentro de la computadora, que lo que hace es guardar dentro de la memoria o variable un tipo de dato o información. Existen diferentes tipos de datos dentro de las variables algunos podrían ser variables de tipo: Float, Boolean, Integer, Character, etc.

En este ejemplo definiremos una variable:

PHP: \$A = 0; (Integer, entero) <- Aquí vemos que estamos definiendo una variable A con un valor entero de 0

- e. Una constante es una variable o un tipo de dato, que siempre es el mismo, ósea que en el código no cambia, por ejemplo, un valor constante podría ser el número PI. Que todos sabemos que es 3,1415, esto sirve de ejemplo para saber que es una variable constante. Las variables constantes suelen escribirse en letras mayúsculas.
- f. Operadores lógicos, son operadores que ayudan a comparar datos o resultados y con esto arrojar un valor booleano y este será Verdadero o Falso (**1,0**).

Los operadores lógicos más habituales son: “**And, Or, Not**”, cada uno de estos funciona de diferentes maneras ejemplo:

AND

Verdadero **AND** Verdadero = Verdadero
Verdadero **AND** Falso = Falso
Falso **AND** Verdadero = Falso
Falso **AND** Falso = Falso

OR

Verdadero **OR** Verdadero = Verdadero
Verdadero **OR** Falso = Verdadero
Falso **OR** Verdadero = Verdadero
Falso **OR** Falso = Falso

NOT

NOT Verdadero = Falso
NOT Falso = Verdadero

- g. Operadores Matemáticos, son operadores que le indican a la variable que operación va a realizar.

A continuación ejemplos de operadores:

Operador unario, solo tiene influencia en una variable, podría ser incrementar de 1 en 1 una variable tal y como se hace **i++**.

Operadores binarios, utilizan dos datos o variables para hacer una operación, por ejemplo los operadores aritméticos básicos de matemáticas tales como: Suma+, Resta-, Multiplicación*, División/.

Ejemplo, i=10 | b=15
i+b = 25
i*b = 150
i-b = 5
i/b = 0,6

Cabe resaltar que muchas veces estos operadores, tienen fuerza de jerarquía, es decir unos se ejecutan primero que otros ya que influyen mas o tienen mas prioridad.

CONOCIMIENTO DE PROGRAMACIÓN ORIENTADO A OBJETOS

1. ¿Cuál es la descripción que crees que define mejor el concepto de ‘clase’ en POO?
 - a. Es un método o función

- b. Es un tipo particular de variable
- (X). Es un modelo o plantilla a partir de la cual creamos objetos
- c. ninguna de las anteriores

2. ¿Qué significa instanciar una clase y de un ejemplo?

R=/ Como se responde en el punto 1, una clase es un modelo o una plantilla a partir de la cual creamos objetos, es un molde donde este tiene consigo unos atributos que vendrían siendo sus datos, y unos métodos los cuales serían sus funcionalidades, de este molde podemos crear los objetos que queramos, y siempre y cuando estos objetos sean creados en esta misma clase, tendrán los mismos datos y atributos y mismos métodos o funciones.

3. ¿Qué elementos componen una clase?

R=/ Objetos, métodos, atributos, instancia.

4. ¿Qué elementos componen un objeto?

R=/ Métodos, atributos.

Defina con sus propias palabras qué es una interfaz

R=/

5. Defina con sus propias palabras qué es una herencia

R=/ Ya sabemos que es una clase, una clase es un molde del cual se crean objetos, esta clase tiene, sus atributos y sus métodos, la herencia es, tener una clase padre, y crear otras clases que podrían ser clases "hijas", de esta clase padre, las clases hijas heredarán atributos y métodos de la clase padre, y no necesariamente tienen que ser una copia de la clase padre, simplemente la clase hijas aprovecha atributos y métodos que le puedan funcionar a esta, que la clase padre ya los tiene.

6. Defina con sus propias palabras qué es polimorfismo.

R=/ Dar una instrucción o función a diferentes clases u objetos, y que estos expresen la función a su manera.

7. Defina con sus propias palabras qué es encapsulamiento.

R=/ Encapsulamiento es básicamente protección de clases, ya que al trabajar con la POO habrá muchos objetos entre sí que se estarán comunicando, esto podría traer problemas si dado el caso algún objeto se comunica con uno que no debía, podría ya no funcionar el programa o código como se espera, por eso se usa el encapsulamiento que básicamente son protecciones de datos en las que los objetos podrán comunicarse con estos datos o no si son privados o públicos etc.

