

TRABAJO PRACTICO N° 2

PARADIGMAS DE PROGRAMACIÓN

Dr. Pablo Javier Vidal

Unidad 2

Ejercicio 1.

1. Indique cuales son los valores de las variables al finalizar cada uno de los siguientes segmentos de código JAVA, asumiendo que todas han sido declaradas como enteras

- a) `a=2; a++;`
- b) `a=2; b=a++;`
- c) `a=2; b=++a;`
- d) `a=2; b=1; b+=a;`
- e) `b=1; b*=5;`
- f) `a=2; b=1; b += -a + 5;`
- g) `a=2; b=2; a += b-;`
- h) `a = 2; b = 3; c = 5; a++; b += a; c *= b; b -= 3; a %= 2; c /= 5;`
- i) `a = 1; b = 2; b++; b = ++a; a *= 2; b += a; a = ++b + 2;`
- j) `a = 1; b = 4; a++; b += a; a *= 4; b -= a; ++b; a = ++b; b = -a + b;`

2. Reescriba las porciones de código del inciso (i) utilizando solamente los operadores `=`, `+`, `-`, `*`, `/` y
3. Indique cuales son los valores de las variables al finalizar cada uno de los siguientes segmentos de código Java, asumiendo que a y b se declararon como enteras y v de tipo boolean.

- a) `a=1; b=2; v=(a++<b);`
- b) `a=1; b=2; v=(++a<b);`
- c) `a=1; b=2; v=(++a>=b);`
- d) `a=1; v= (a`
- e) `a=1; v= (a != 1);`
- f) `a=1; b=2; v=(b++<10 && a==1);`
- g) `a=1; b=2; v=(b==1 && a>=1);`
- h) `a=1; b=2; v=(b<10 || a++==2);`
- i) `a=1; b=2; v=(-b<=1 || a<=10);`
- j) `a=1; b=2; v=(a--==2 || b==1);`
- k) `a=1; b=2; v=!(a==1 && ++b==1);`

Introducción a Java

Para la realización de los siguientes ejercicios, el alumno deberá definir una clase y por cada ítem solicitado deberá colocar la implementación del mismo en un método

Ejercicio 2.

1. Implementar un programa que dado dos números informe cual es el mayor.
2. Declarar dos variables X e Y de tipo `int`, dos variables N y M de tipo `double` y asignar a cada una un valor. A continuación mostrar por pantalla el resultado de una serie de operaciones matematicas basicas entre ellas(las operaciones son a elección del alumno).
3. Declarar cuatro variables enteras A, B, C y D y asignarle un valor diferente a cada una. A continuación, realizar las instrucciones necesarias para que: B tome el valor de C, C tome el valor de A, A tome el valor de D, D tome el valor de B. Mostrar los valores iniciales y los valores finales de cada variable.
4. Calcular el volumen de una esfera. Recordar que para calcular el volumen se debe utilizar la siguiente formula $V = \frac{4}{3} * \pi * r$ siendo r^3 el radio de la esfera.
5. Calcular el área de un triángulo a partir de la longitud de sus lados. El calculo de un semiperímetro es $p = \frac{1}{2} * (a + b + c)$ siend a, b y c las longitudes.

Ejercicio 3.

1. Implementar un programa que defina un vector de tamaño 10 y completar con números del 1 al 10.
2. Reutilizando el código anterior, definir un programa que muestre la tabla del 3, luego la del 4 y finalmente la del 8.
3. Definir un vector y completar cada posición con números del 1 al 20. Una vez completado el vector sumar los valores generados. NO se debe realizar la misma operación de asignación del dato y suma en la misma estructura de repetición. NO utilizar una estructura `for`.
4. Definir una matriz de 2×2 . Para cada posición (i,j), asignar el valor resultante de la operación $valor = (2 * i) + (j + 3)$.
5. Definir una matriz de 20×20 elementos. Completar dicha matriz con los valores generados por $valor = fila * 20 + columna$.

Ejercicio 4.

1. Diseñar un programa que permita ingresar diferentes caracteres, el programa solo se detendrá si se ingresa una letra X.
2. Leer un nombre y muestre por pantalla: "Buenos días nombre_introducido".
3. Leer un número entero y calcule si es par o impar.
4. Leer un número por teclado que pida el precio de un producto (puede tener decimales) y calcule el precio final con IVA. El IVA será una constante que sera del 21 %.
5. Leer dos números por teclado y mostrar el resultado de la división del primero por el segundo. Se debe comprobar que el divisor no puede ser cero.
6. Calcular el promedio de una serie de números que se leen por teclado.
7. Pedir un día de la semana y que nos diga si es un día laboral o no. Usar una instrucción `switch` para ello.

Estructuras

Ejercicio 5.

1. Analice si los siguientes segmentos de instrucciones son equivalentes en términos del ámbito y los valores de las variables

```
1 float sum=0; int i;  
2 for (i = 10; 1/i > sum; i--){  
3     sum = sum + 1/i;  
4 }  
5
```

```
1 float sum=0;  
2 for (int i = 10; 1/i > sum; i--){  
3     sum = sum + 1/i;  
4 }  
5
```

```
1 for (int sum=0, i = 10;  
2     1/i > sum; i--){  
3     sum = sum + 1/i;  
4 }  
5
```

2. Considere los siguientes fragmentos extraídos de un programa. Asuma la siguiente declaración de variables:

int a,b;
char c;

```
a= 1;  
b= 1;  
if((a=2)>(b=1))  
    a = b;  
else  
    b = a;
```

```
a= 1;  
b= 1;  
if (a > b)  
    b = a;  
    a = 0;  
else  
    a = b;  
    b = 0;
```

```
a = 1;  
b = 1;  
if (a == b) {  
    a=1;  
    b=2;  
}  
else {  
    a=2;  
    b=1;  
}
```

Ejercicio 6.

1. Implementar un programa donde se tiene el dato del día (puede ser número o letra) e informar a que día corresponde.
2. Implementar un programa que sume los números del 1 al 10. Utilizar las tres estructuras de repetición: for, do-while, while.
3. Dado un valor booleano informar si es verdadero o falso.
4. Dado un carácter numérico informar si es un dígito o no.

Ejercicio 7.

¿Qué afirmación es cierta? Justifique su respuesta teniendo en cuenta el siguiente código:

```
1 Ordenador escritorio; Ordenador portatil;  
2 escritorio = new Ordenador();  
3 escritorio.precio(900);  
4 portatil = new Ordenador();
```

```

5 portatil.precio(1100);
6 portatil = escritorio;
7 escritorio = null;

```

1. Cuando se ejecuta la línea 5, la instancia escritorio cuesta 1100.
2. Cuando se ejecuta la línea 5, la instancia portatil cuesta 1100.
3. Al final tanto el objeto escritorio como el objeto portátil apuntan a null.
4. Al final sólo queda un objeto de tipo Ordenador con precio 1100.
5. Al final sólo queda un objeto de tipo Ordenador con precio 900.
6. Al final hay dos objetos de tipo Ordenador, uno con precio 900 y otro con precio 1100.

Ejercicio 8.

Crear los constructores necesarios para poder instanciar un **Arbol** con su altura y nombre de la siguiente manera:

```

1 public static void main(String args[]) {
2     Arbol arbol1 = new Arbol(4);
3     Arbol arbol2 = new Arbol("Roble");
4     Arbol arbol3 = new Arbol();
5     Arbol arbol4 = new Arbol(5, "Pino");
6 }

```

Cada constructor deberá mostrar un mensaje por consola con información, si no tiene datos es un árbol genérico.

Clases con Java

Ejercicio 9.

Para cada una de las clases, definir siempre los métodos getters y setters aparte de los solicitados.

1. Definir la clase Persona, reutilizando lo definido hasta el momento. Implementar al menos 4 métodos propios del comportamiento de la clase Persona y además los métodos getters y setters. Finalmente, generar una instancia en el método *main*.
2. Definir una clase Punto. Generar dos instancias P1 y P2. Comprobar la distancia que existe desde P1 a P2.
3. Diseñar una clase Sumatoria que tenga: una variable vector de tipo entero y dos métodos, uno que permita mostrar la tabla de multiplicar para cada variable y otro que permita devolver la suma. Debe tener un constructor por defecto que les asigne valores entre 1 y 10 o bien un constructor para ingresar los dos valores. En el main de prueba deberá generar dos instancias de la clase Sumatoria, mostrar las tablas de multiplicar y finalmente mostrar la suma de ambas variables por pantalla.
4. Crear una clase *Libro* que contenga los siguientes atributos: ISBN, Título, Autor, Número de páginas. Definir 5 instancias y para cada una de ellas informar mediante un método el número de ISBN, el título y el autor.
5. Crear una clase *Fraccion* con métodos para sumar, restar, multiplicar y dividir fracciones. En cada método se debe mostrar el resultado de dicha operación.