Guía de Trabajos Prácticos de Pascal

# Parte 1

Cada ejercicio debe ser copiado/escrito en la versión de Pascal que posean y luego compilarlo, probarlo y “decorarlo”

1) Escribir un programa en pascal que dado dos números enteros realice la suma, resta, multiplicación, división entera y resto de los mismos. Los numero son: X = 55 e Y = 30.

program ej1;

Var

x, y : integer;

sum, res, mul, die, moe: integer;

begin

x := 55;

y := 30;

writeln('Valor de X es: ', X);

writeln('Valor de Y es: ', Y);

sum := x+y;

res := x-y;

mul := x\*y;

die := x div y;

moe := x mod y;

writeln('Los resultados son:');

writeln('suma: ', sum);

writeln('resta: ', res);

writeln('multiplicacion: ', mul);

writeln('division entera: ', die, ' y el modulo es: ', moe);

end.

2) Idem al ejercicio anterior, pero ahora tanto x como y deben ser ingresados por el usuario

program ej2;

Var

x, y : integer;

sum, res, mul, die, moe: integer;

begin

writeln('Ingrese valor de x: ');

read(y);

writeln('Ingrese valor de y: ');

read(x);

writeln('Valor de X es: ', X);

writeln('Valor de Y es: ', Y);

sum := x+y;

res := x-y;

mul := x\*y;

die := x div y;

moe := x mod y;

writeln('Los resultados son:');

writeln('suma: ', sum);

writeln('resta: ', res);

writeln('multiplicacion: ', mul);

writeln('division entera: ', die, ' y el modulo es: ' moe);

end.

3) Idem al anterior pero ahora el usuario es el que ingresa que operación quiere hacer.

program ej3;

Var

x, y : integer;

opcion: integer;

result: integer;

begin

writeln('Ingrese valor de x: ');

read(y);

writeln('Ingrese valor de y: ');

read(x);

writeln('Valor de X es: ', X);

writeln('Valor de Y es: ', Y);

writeln('Ahora, qué operación queres hacer:');

readln(opcion);

case opcion do

1: result := x+y;

2: result := x-y;

3: reuslt := x\*y;

4: result := x div y;

5: result := x mod y;

end;

writeln('El resultado de la operación ', opción, ' es: ', result);

end.

4) Ídem al ítem 1 pero ahora con variables REALES.

program ej4;

Var

x, y : real;

sum, res, mul, die: real;

begin

x := 431.43;

y := 131.14;

writeln('Valor de X es: ', X);

writeln('Valor de Y es: ', Y);

sum := x+y;

res := x-y;

mul := x\*y;

die := x / y;

writeln('Los resultados son:');

writeln('suma: ', sum:10:2);

writeln('resta: ', res:10:2);

writeln('multiplicación: ', mul:10:4);

writeln('división: ', die:10:4);

end.

5) Hacer el mismo ejercicio que está en el punto 3 pero ahora con variables reales.

6) En un colegio primario se necesita realizar un algoritmo que en vez de colocar las notas de los alumnos en forma numérica, la misma debe ser con una frase, el algoritmo debe traducir las notas del 0 al 10 en las siguientes frases:

* de 0 a 3: Mal
* de 4 a 5: Insuficiente
* de 6 a 7: Bien
* de 8 a 9: Sobresaliente
* cuando es 10: Perfecto
* Para valores mayores a 10 o menores a 0: ERROR, numero fuera de rango.

program ej6;

Var

nota : integer;

begin

writeln('Ingrese la nota del alumno: ');

read(nota);

case nota of

0..3: writeln('Mal');

4, 5: writeln('Insuficiente');

6..7: writeln('Bien');

8, 9: writeln('Sobresaliente');

10: writeln('Perfecto');

else

writeln('ERROR, numero fuera de rango.');

end;

end.

7) Idem al ejercicio anterior, pero evitar que el programa salga cuando sea un error (numero fuera del rango) y volver a solicitarlo.

program ej7;

Var

nota : integer;

begin

nota := 0;

repeat

if not(nota in [1..10]) then

writeln('ERROR, numero fuera de rango. vuelva a cargar la nota');

writeln('Ingrese la nota del alumno: ');

read(nota);

until nota in [1..10];

case nota of

0..3: writeln('Mal');

4, 5: writeln('Insuficiente');

6..7: writeln('Bien');

8, 9: writeln('Sobresaliente');

10: writeln('Perfecto');

end;

end.

8) Tomando el ejercicio anterior, realizar lo mismo pero para 10 alumnos, aparte calcular el promedio de notas.

9) Tomando el ejercicio 7) realizar lo mismo pero para una cantidad indeterminada de alumnos, aparte calcular el promedio de notas.

10) Determinar si un número es primo

Este ejercicio se puede resolver con pseudocódigo de esta forma:

acción ej10 es

ambiente

num, aux : entero;

esPrimo : logico;

algoritmo

escribir(“Por favor ingrese un numero para saber si es entero”);

leer(num);

aux := 2;

esPrimo := Verdadero;

mientras esPrimo y aux < num hacer

si num mod aux = 0 entonces

esPrimo := falso;

fin si

aux := aux + 1;

fin mientras

si esPrimo entonces

escribir(“El numero ingresado es Primo”);

sino

escribir(“El numero ingresado no es primo”);

fin si

fin acción

Se pide pasar a la sintaxis de pascal

11) Se pide crear un pequeño login para usuarios, el usuario tiene que pasar su nombre y su contraseña, el sistema tiene que comparar contra unos valores previamente cargados y verificar si son o no iguales.

program ej11;

const

user = 'Usuario';

pass = 'password';

var

usuario, contraseña: string;

begin

write('Ingrese el usuario: ');

readln(usuario);

write('Ingrese el password: ');

readln(contraseña);

if user = usuario then

begin

if pass = contraseña then

begin

writeln('Usuario y contraseña valido.');

end

end

else

writeln('Usuario o contraseña invalida.');

end.

12) Dado el ejercicio anterior, realizar el mismo algoritmo, solo que ahora no se muestre el valor de la contraseña cuandos se escribe por el teclado.

program ej12;

uses CRT;

const

user = 'Usuario';

pass = 'password';

var

usuario, contrasena: string;

letra: char;

begin

write('Ingrese el usuario: ');

readln(usuario);

write('Ingrese el password: ');

contraseña :='';

Repeat

letra:=readkey;

If letra<>#13 Then

Begin

contraseña :=contraseña + letra;

Write(chr(42))

End;

until letra=#13;

writeln();

if user = usuario then

begin

if pass = contraseña then

begin

writeln('Usuario y contraseña válido.');

end;

end

else

writeln('Usuario o contraseña inválida.');

end.

13) Realizar el ejercicio 12) pero permitir que el usuario tenga la posibilidad de ingresar la contraseña 3 veces.

## Complementarios

14) Se pide realizar un juego en donde cada jugador ingresa un número entre 1 y 100, el jugador que ingresa el número más grande entre los dos gana la cantidad de punto de (100 - el número elegido).

Program ej14;

uses Crt;

var

juga1punt, juga2punt: integer;

juga1num, juga2num: integer;

i: integer;

begin

writeln('Hola, bienvenido al juego');

writeln('El jugador que ponga el menor número que sea mayor al otro' +

'jugador, va a ganar (100 - el\_numero\_elegido) de puntos');

writeln('Solo se puede elegir un número entre el 1 y el 100.');

writeln('Toque una tecla para comenzar.');

readkey;

juga1punt:=0;

juga2punt:=0;

for i := 1 to 3 do

begin

writeln('Jugador 1 juega, Ingrese un numero');

readln(juga1num);

clrscr;

writeln('Perfecto, el jugador 2.');

readkey;

writeln('Jugador 2 juega, Ingrese un numero');

readln(juga2num);

clrscr;

writeln('Comparando...');

delay(500);

if (0 >= juga1num) and (100 < juga1num) then

juga1num := 100;

if (0 >= juga2num) and (100 < juga2num) then

juga2num := 100;

if juga1num < juga2num then

begin

writeln('Gano el jugador 2');

writeln('Suma: ', 100 - juga2num, ' puntos');

juga2punt := juga2punt + (100 - juga2num);

end

else

begin

writeln('Gano el jugador 1');

writeln('Suma: ', 100 - juga1num, ' puntos');

juga1punt := juga1punt + (100 - juga1num);

end;

end;

writeln('Puntos jugador 1: ', juga1punt);

writeln('Puntos jugador 2: ', juga2punt);

if juga1punt > juga2punt then

writeln('Gano Jugador 1')

else

writeln('Gano Jugador 2');

end.

15) Arreglar el ejercicio anterior para que tenga las nuevas funcionalidades:

* Permita el empate tanto en cada iteración como en el final.
* Se puedan elegir la cantidad de iteraciones.
* Cuando el jugador ingresa un número fuera del rango, el juego deberá informa el error y permitir que vuelva a cargar el valor elegido.