

Simulacro Segundo Parcial 2021

1) Los módulos **esNroElegido1** y **esNroElegido2** son equivalentes: Para la misma entrada producen la misma salida. Si vemos cómo están programados, la especificación es distinta, es decir que resuelven el mismo problema de forma diferente. Desarrollar los siguiente incisos

- Realizar la traza de **“esNroElegido1(6)”**. Si la hace en papel, enviar una foto de calidad
- Realizar la traza de **“esNroElegido2(6)”** Si la hace en papel, enviar una foto de calidad
- Traducir ambos módulos a PHP
- contestar:
 - ¿cuántos ciclos se producen al ejecutar la instrucción repetitiva en **“esNroElegido1(6)”** ? **2 ciclos**
 - ¿cuántos ciclos se producen al ejecutar la instrucción repetitiva en **“esNroElegido2(6)”** ? **1 ciclo**
 - ¿cuántos ciclos se producen al ejecutar la instrucción repetitiva en **“esNroElegido2(2)”** ? **0 ciclo**
 - ¿Qué **clase de números eligen estos módulos**? ¿Propondría un nombre más significativo para los módulos? Observación: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31 son número elegidos. **determina si un número es primo**

DIVISORES DE 9: 1, 3
MITAD DE 9 = $9 / 2 = 4.5$

```
MODULO esNroElegido1(ENTERO num) RETORNO BOOLEAN
    ENTERO umbral, i, BOOLEAN bandera
1    umbral <-- ((ENTERO)(num/2))+1
2    bandera <-- true
3    PARA i <-- 2 HASTA umbral PASO 1 HACER
4        SI(num MOD i = 0) ENTONCES
5            bandera <-- false
6        FIN SI
7    FIN PARA
8    RETORNO (bandera)
FIN MODULO
```

```
<?php
/**
 * Determina si el número es primo
 * @param int num
 * @return boolean
 * */
function esPrimo($num) {
    //int $umbral, $i, boolean $bandera
    $umbral = ((int)($num /2))+1;
    $bandera = true;
    for ($i=2; $i<$umbral; $i++) {
        if ($num % $i == 0) {
            $bandera = false;
        }
    }
    return $bandera;
}
```

```

}

echo "Ingrese un nro: ";
$nro = trim(fgets(STDIN));
if (esPrimo($nro)) {
    echo "Es primo";
} else {
    echo "No es primo";
}

```

Traza esNroElegido1(6)

ciclos: 2

°	num	umbral	i	bandera	retorno
0	6				
1		4			
2				true	
3			2		
5				false	
3			3		
5				false	
3			4		
8					false

MODULO esNroElegido2(ENTERO num) **RETORNO** BOOLEAN

```

    ENTERO umbral, i, BOOLEAN bandera
1    umbral <-- ( (ENTERO)(num/2) )+1
2    bandera <-- true
3    i <-- 2
4    MIENTRAS(bandera AND i<umbral) HACER
5        bandera <-- NOT (num MOD i = 0)
6        i <-- i + 1
7    FIN MIENTRAS
8    RETORNO (bandera)
FIN MODULO

```

```

/**
 * Determina si un número es primo
 * @param int $num
 * @return boolean
 */

```

```
function esPrimo2(int $num){
    //int $umbral, $i, boolean $bandera
    $umbral = ( (int)($num /2) )+1;
    $bandera = true;
    $i= 2;
    while ( $bandera && $i< $umbral) {
        $bandera = !($num % $i == 0);
        $i = $i +1;
    }
    return $bandera;
}
```

traza esNroElegido2(2)

CICLO: 0

°	num	umbral	i	bandera	retorno
0	2				
1		2			
2				true	
3			2		
8					true

Traza esNroElegido2(6)

ciclos: 1

°	num	umbral	i	bandera	retorno
0	6				
1		4			
2				true	
3			2		
5				false	
6			3		
8					false

2) Programar **una función** sumaElegidosMenores en pseudocódigo y traducirla a php:

a) **Parámetro:** La función recibe por parámetro un número natural n (mayor a 1). En el cuerpo de la función no debe controlar el valor del parámetro, ya que es una precondition de la función

b) **Retorno:** La función retorna un valor entero

c) **Proceso:** el proceso consiste en **sumar** los **números naturales Elegidos menor o iguales a n** (invoque a alguna de las funciones del ejercicio 1) → ciclo definido: **UTILIZAMOS LA INSTRUCCION PARA**

(** *determina la suma de números elegidos hasta num**)
1 MODULO sumaElegidosMenores(ENTERO num) RETORNO ENTERO

ENTERO cont (* contador *), sumaNros (*acumulador de números*)

2 sumaNros ← 0

3 PARA cont ← 2 **HASTA** num+1 **PASO 1 HACER**

4 SI (esNroElegido2(cont)) ENTONCES

5 sumaNros ← sumaNros + cont

FIN SI

FIN PARA

6 RETORNO(sumaNros)

FIN MODULO

function sumaElegidosMenores(\$num){
// declarar las variables **COMPLETAR..**

\$sumaNros = 0;

for(\$cont = 2 ; \$cont <= \$num ; \$cont++) {

if(esNroElegido2(\$cont){

\$sumaNros = \$sumaNros + \$cont;

}

}

return \$sumaNros;

}

traza sumaElegidosMenores(2)

°	\$num	\$cont	\$sumaNros	retorno
1	2			
2			0	
3		2		
5			2	
3		3		
6				2

traza esNroElegido2(2)

°	num	umbral	i	bandera	retorno
0	2				
1		2			
2				true	
3			2		
8					true

*(** determina la suma de números elegidos hasta num*)*

1 MODULO sumaElegidosMenores(ENTERO num) RETORNO ENTERO

ENTERO cont (* contador *), sumaNros (*acumulador de números*)

2 sumaNros ← 0

3 PARA cont ← 1 HASTA num+1 PASO 1 HACER

4 SI (esNroElegido2(cont)) ENTONCES

5 sumaNros ← sumaNros + cont

FIN SI

FIN PARA

6 RETORNO(sumaNros)

FIN MODULO

traza sumaelegidosMenores(2)

°	num	cont	sumaNros	retorno
1	2			
2			0	
3		4		
5			4	
3		2		
5			3	
3		3		
6				3

traza esNroElegido2(2)

o	num	umbral	i	bandera	retorno
0	2				
1		2			
2				true	
3			2		
8					true

traza esNroElegido2(1)

o	num	umbral	i	bandera	retorno
0	1				
1		1			
2				true	
3			2		
8					true

*(** determina la suma de números elegidos hasta num*)*
MODULO sumaElegidosMenores(ENTERO num) **RETORNO** ENTERO
 ENTERO cont (* contador *), sumaNros (*acumulador de números*)
 sumaNros ← 0
 PARA cont ← 0 **HASTA** num **PASO** 1 **HACER**
 SI (esNroElegido2(cont+1)) **ENTONCES**
 sumaNros ← sumaNros + (cont+1)
 FIN SI
 FIN PARA
 RETORNO(sumaNros)
FIN MODULO

Atención: Busque la solución más sencilla. Se evaluará el tipo de ciclo elegido (definido o indefinido).

Ejemplos:

a) Entrada: 2 Salida: 2

Proceso: $0 + 2 = 2$

b) Entrada: 6 Salida: 10

Proceso: $0 + 2 + 3 + 0 + 5 + 0 = 10$

2, 3, 5 son números elegidos

1, 4 y 6 no son números elegidos

c) Entrada: 12 Salida: 28

Proceso: $0 + 2 + 3 + 0 + 5 + 0 + 7 + 0 + 0 + 0 + 11 + 0 = 28$

2, 3, 5, 7 y 11 son números elegidos

1, 4, 6, 8, 9 y 10 no son números elegidos

d) Entrada: 17 Salida: 58

Proceso: $0 + 2 + 3 + 0 + 5 + 0 + 7 + 0 + 0 + 11 + 0 + 13 + 0 + 0 + 0 + 17 = 58$

3) Diseñar en Pseudocódigo y implementar en PHP

Se quiere relevar cierta información de la empresa IPLearning, para lo cual se necesita un **programa principal** que solicite como **entrada**:

i) nombre del empleado

ii) edad

iii) puesto: "a" = administrativo, "t" = técnico, "p" = profesional

Para obtener la siguiente **salida**:

a) Nombre de la persona con menor edad.

b) Qué porcentaje representan los profesionales en la empresa

c) Promedio de edad del personal técnico

Para la resolución considere utilizar un ciclo **interactivo** que le permite al usuario ingresar **cero o más empleados**.

Atención: Antes de especificar el programa, revise detenidamente cada uno de los siguientes ejemplos para comprender mejor la entrada y salida esperada en distintos casos. Además para la salida se muestran valores enteros. Se evaluará el tipo de ciclo elegido.

ANÁLISIS:

ENTRADA (LECTURA - teclado): STRING nombre, ENTERO edad, STRING puesto

SALIDA (ESCRIBIR -pantalla):

a) Nombre de la persona con **menor** edad

ENTERO edadMenor (variable **selectora**)

STRING nombreDeEdadMenor

b) Qué porcentaje representan los profesionales en la empresa

5 --- 100%
3 ---- $x = 3 \cdot 100 / 6$

cantEmpleados --- 100%
cantProfesionales --- $x = \text{cantProfesionales} \cdot 100 / \text{cantEmpleados}$

ENTERO cantProfesionales (variable **contadora de profesionales**)
ENTERO cantEmpleados (variable **contadora de empleados**)

FLOAT porcentajeProfesionales

c) Promedio de edad del personal técnico

ENTERO sumaEdadesTecnicos (variable **Acumuladora** de Edades)
ENTERO cantTecnicos (variable **contadora** de técnicos)

FLOAT promedioEdadesTecnicos

TIPO DE CICLO: Indefinido INTERACTIVO, cero o más empleados (MIENTRAS)
STRING respuesta (“sí” o “no”) ,

PROGRAMA Principal

(* realiza cálculos para la empresa IPLearning *)

(* Declarar variables *)

STRING nombre, puesto, ENTERO edad

ENTERO edadMenor (*variable **selectora***)

STRING nombreDeEdadMenor

ENTERO cantProfesionales (*variable **contadora de profesionales***)

ENTERO cantEmpleados (*variable **contadora de empleados***)

FLOAT porcentajeProfesionales

ENTERO sumaEdadesTecnicos (*variable **Acumuladora** de Edades*)

ENTERO cantTecnicos (*variable **contadora** de técnicos*)

FLOAT promedioEdadesTecnicos

STRING respuesta

(* Inicializar variables *)

edadMenor ← 9999

cantProfesionales ← 0

cantEmpleados ← 0

sumaEdadesTecnicos ← 0

cantTecnicos ← 0

(*proceso*)

ESCRIBIR(“Desea ingresar un empleado: ”)

LEER(respuesta)

MIENTRAS (respuesta = “sí”) HACER

(*entrada:*)

ESCRIBIR(“Nombre:”)

LEER(nombre)


```

ESCRIBIR("Edad:")
LEER(edad)
ESCRIBIR("Puesto (a/p/t):")
LEER(puesto)

```

```

cantEmpleados ← cantEmpleados + 1
SI(puesto="p" )ENTONCES
    cantProfesionales ← cantProfesionales + 1
OTRO-SI(puesto="t" )ENTONCES
    sumaEdadesTecnicos ← sumaEdadesTecnicos + edad
    cantTecnicos ← cantTecnicos + 1
FIN SI

```

```

SI(edad < edadMenor)ENTONCES
    edadMenor ← edad
    nombreDeEdadMenor ← nombre
FIN SI

```

```

ESCRIBIR("Desea ingresar otro empleado: ")
LEER(respuesta)
FIN MIENTRAS

```

```

(* resultados *)
SI( cantEmpleados >0 )ENTONCES
    ESCRIBIR("El empleado de menor edad es:", nombreDeEdadMenor)

    porcentajeProfesionales ← cantProfesionales *100 / cantEmpleados
    ESCRIBIR("El porc. de profesionales es:", porcentajeProfesionales,"%")

    promedioEdadesTecnicos ← sumaEdadesTecnicos / cantTecnicos
    ESCRIBIR("El promedio de edad de Tecnicos es:", promedioEdadesTecnicos )
SINO
    ESCRIBIR("Error, no se ingresaron datos")
FIN SI

```

FIN PROGRAMA

Ejemplo 1:

Entrada	Salida
Pedro 30 p Anna 26 a Mirta 45 p Hector 31 p Juan 30 t	a) Anna b) 60 c) 30

Ejemplo 2:

Entrada	Salida
---------	--------

Franco	51	p	a)	Carla
Marco	26	t	b)	25
Brenda	45	a	c)	25
Carla	24	t		

Ejemplo 3:

Entrada	Salida
	Error, no se ingresaron datos