

U.T. 2.- Introducción a la Programación estructurada y modular en los lenguajes de script clientes y servidor

Nombre del Alumno: _____

Calificación Final de la PRUEBA: _____

Condiciones para el Examen:

- El alumno deberá utilizar exclusivamente los recursos del aula para realizar la prueba, por lo que no podrá usar su equipo informático.
- No tendrá acceso a Internet, por lo que deberá tener acceso local a los recursos materiales y digitales que precise para realizar la prueba.

2,25p

Criterio de calificación C1: Saber analizar un algoritmo con el fin de determinar el grado de consecuencia de los elementos de calidad, más concretamente en la corrección de dichos artefactos Sw.

2,25p

Criterio de calificación C2: Tener la capacidad de completar el flujo de control de una porción de un artefacto Sw con el fin de proveer de una solución con factores de calidad.

2,25p

Criterio de calificación C3: Saber interpretar un flujo de control.

3,25p

Criterio de calificación C4: Saber en base a unas especificaciones elaborar el algoritmo/pseudocódigo que resuelva con parámetros de calidad los requerimientos planteados.

2,25p

Criterio de calificación C1: Saber analizar un algoritmo con el fin de determinar el grado de consecuencia de los elementos de calidad, más concretamente en la corrección de dichos artefactos Sw.

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 1.-
	0,0	0,25	0,50	0,75		

Corrige los errores no sintácticos que provoca que el código que se proporciona a continuación no realice la acción indicada.

/* Dados un vector v y un valor x determina si la media de valores del vector es superior al valor x dado por el usuario*/

var

```
secuencia : v[10] numerico;  
numero : numerico  
indice : numerico  
valor: numerico  
suma: numérico=0;
```

inicio

```
/* Suponemos que leemos el vector*/  
imprimir ("\nIntroduce el numero a burca= ")  
leer(valor)  
indice=1  
mientras (alen(v)<=indice)  
{  
    indice=indice+1  
    suma=suma+v[indice]  
}  
si(suma<=valor)  
{  
    imprimir("\nLa media es mayor al valor dada..")  
sino  
    imprimir("\nLa media es mayor al valor dada..")  
}
```

fin

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 2.-
	0,0	0,25	0,50	0,75		

Corrige los errores no sintácticos que provoca que el código que se proporciona a continuación no realice la acción indicada.

/* Dados un vector v determina si está ordenado*/

```
var
    v : vector[10] numerico
    indice: numerico
    end : numerico
inicio
    indice=1
    mientras(indice<=alen(v))
    {
        imprimir("\n Introduce un valor: ")
        leer(v[indice])
        indice=indice+1
    }
    end=0
    mientras(indice<alen(v) and end==1)
    {
        si(v[indice]>v[indice+1])
        {
            end=0
            indice=indice+1
        }
    }
    si(end==0)
    {
        imprimir("\n no está ordenado")
    }
    sino
        imprimir("\n está ordenado")
    }
fin
```



	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 3.-
	0,0	0,25	0,50	0,75		

Corrige los errores no sintácticos que provoca que el código que se proporciona a continuación no realice la acción indicada.

/**/

Un **número abundante** es un número natural que es menor que la suma de sus divisores propios.
Un **número deficiente** es un número natural que es mayor que la suma de sus divisores propios.
Los divisores propios de un número son los que están comprendidos entre (1..N-1)

Ejemplo: 23

Proceso: Se determinará si 23 es un número abundante o deficiente. (23%1?, 23%2?,23%22,)

/**/

var

numero: numerico
indice : numerico;
suma: numerico

inicio

```
imprimir("\n Introduce un valor")
leer(numero)
suma=0
indice=1
mientras(indice<numero)
{
    si(numero%indice<>0)
    {
        suma=suma+índice
        indice=indice+1
    }
}
si(numero>suma)
{
    imprimir("\n el valor es abundante")
}
si(numero<suma)
{
    imprimir("\n el valor es deficiente")
}
```

fin



2,25p

Criterio de calificación C2: Tener la capacidad de completar el flujo de control de una porción de un artefacto Sw con el fin de proveer de una solución con factores de calidad.

<input type="checkbox"/>		Cuestión 4.-				
0,0	0,25	0,50	0,75			

Rellena el espacio para completar el código que se proporciona.

/* Leer una secuencia de números almacenándolos en un vector y mostrar por pantalla sus factoriales.*/

var

```
    secuencia : vector[10] numerico;  
    indice : numerico=1;  
    fact : numerico;  
    indice_fact : numerico
```

inicio

```
    cls()  
    mientras (indice<=10)  
    {  
        secuencia[indice]=random(12)  
        indice=indice+1  
    }
```

```
    indice=1  
    mientras (indice<=10)  
    {
```



```
        indice=indice+1
```

```
    }
```

fin

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 5.-
	0,0	0,25	0,50	0,75		

Rellena el espacio para completar el código que se proporciona.

Dada una frase obtén la misma cambiando minúsculas por mayúsculas:

Ejemplo: Mi correo es juanlop23@um.es

Proceso: Convertir minúsculas por mayúsculas

Resultado: MI CORREO ES JUANLOP23@UM.ES

Nota:

1. Considera en uso de las funciones: strlen() longitud de una cadena, ord() código ascii de un carácter y ascii(e) el carácter asociado a un código ascii
2. El código ascii de la 'a' es 97 y de la 'A' 65.

var

frase: cadena

indice: numerico

longitud: numerico

inicio

imprimir("\n Introduce una frase: ")

leer(frase)

longitud=strlen(frase)

indice=1

mientras(indice<=longitud)

{



 indice=indice+1

}

imprimir("\n el resultado es: ", frase)

fin

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 6.-
	0,0	0,25	0,50	0,75		
Rellena el espacio para completar el código que se proporciona.						
<pre>/* * Dado un vector v1 de solo pares y un valor x determinar cuántos v[i]<x, v[i]>x y v[i]=x, i:1..alen(v). */ var v : vector[10] numerico; indice : numerico=1; cantidad_mayor : numerico; cantidad_menor : numerico; cantidad_igual : numerico; valor : numerico; inicio cls() mientras (indice<=10) { . . . } imprimir("Introduce el valor a buscar: ") leer(valor) indice=1 cantidad=0 mientras (indice<=10) { . . . } imprimir("el valor introducido por el usuario >: ", cantidad_mayor) imprimir("el valor introducido por el usuario <: ", cantidad_menor) imprimir("el valor introducido por el usuario ==: ", cantidad_igual) fin</pre>						

2,25p

Criterio de calificación C3: Saber interpretar un flujo de control.

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cuestión 7.-
0,0	0,25	0,50	0,75	

Determina el enunciado del código que se presenta.

/* Enunciado:



***/**

var

```
a : vector[*] numerico;  
b : numerico=0;  
c : numerico=0;  
i : numerico=1;  
d : numerico;  
e: numerico  
z:numérico=0
```

inicio

```
imprimir(".....")  
leer (d)  
dim(a,d)  
mientras (i<=d)  
{  
    a[i]=random(100)  
    i=i+1  
}  
i=1  
mientras (i<=d)  
{  
    si(a[i]%2==0)  
    {  
        c=c+a[i]  
        z=z+1  
    }  
    i=i+1  
}  
si(z==0)  
{  
    imprimir(".....")  
    sino  
        imprimir(".....", c/z)  
}  
}
```

fin

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 8.-
	0,0	0,25	0,50	0,75		

Determina el enunciado del código que se presenta.

/* Enunciado:



***/**

var

v : vector[5] numerico

t: numerico

q : numerico

inicio

t=1

mientras(t<=alen(v))

{

imprimir("\n Introduce un valor: ")

leer(v[t])

t=t+1

}

t=1

q=0

mientras(t<alen(v))

{

si(v[t]==v[t+1])

{

q=q+1

}

t=t+1

}

imprimir("\n ¡!!!!!!!!!!!! es: ",q)

fin

Nombre:

3,25p

Criterio de calificación C4: Saber en base a unas especificaciones elaborar el algoritmo/pseudocódigo que resuelva con parámetros de calidad los requerimientos planteados.

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 10.-
	0	0,25	R=0,5	1		
<p>Opción 1.-</p> <p>Enunciado: Calcular el valor de Pi utilizando series infinitas. El usuario introducirá la precisión como el número de fracciones que se van a acumular. Utiliza la serie Nilakantha. Esta es una serie infinita que sirve para calcular Pi y que además es bastante fácil de entender. Aunque es más complicada que la fórmula de Gregory-Leibniz, converge en los valores de Pi mucho más rápido. $\pi = 3 + 4/(2*3*4) - 4/(4*5*6) + 4/(6*7*8) - 4/(8*9*10) + 4/(10*11*12) - (4/(12*13*14))$</p> <p>Ejemplo: Introduce la precisión: <u>3</u> Proceso: calcular <u>$3 + 4/(2*3*4) - 4/(4*5*6) + 4/(6*7*8)$</u> Resultado: El valor de PI con precisión 3 es 3,261904</p> <ul style="list-style-type: none"> • Considera utilizar dos índice: <ul style="list-style-type: none"> ○ índice=2 y en cada iteración índice=índice+2 ○ índice_signo=1 y en cada iteración índice_signo=índice_signo*-1 <p>Opción 2.-</p> <p>Enunciado: Dado una frase que sólo contiene caracteres dígitos {'0'..'9'} calcula en el valor entero correspondiente: Ejemplo: Introduce la frase numérica: "2456" Proceso: Hay que extraer cada frase[i] para construir el numérico asociado. frase[1]='2' entero parcial 2 frase[2]='4' entero parcial 24 frase[3]='5' entero parcial 245 frase[4]='6' entero parcial 2456</p> <p>Resultado: El entero correspondiente a la frase es 2456</p> <ul style="list-style-type: none"> • Considera en uso de las funciones: strlen() longitud de una cadena, ord() código ascii de un carácter y ascii(e) el carácter asociado a un código ascii • El código ascii del '0' es 48 y del '9' es 57. <p>Opción 3.-</p> <p>Enunciado: Dado un <u>vector v</u> de enteros determina la media de los múltiplos de un <u>valor x</u> proporcionado por el usuario. Ejemplo: v=[5,6,124,50,45] y x=5 Proceso: obtener a media de 5,50,45 dado que son múltiplos de 5 Resultado: La media de los múltiplos de 5 es: 33.33</p> <p>Opción 4.-</p> <p>Enunciado: Dado dos años proporcionados por el usuario x,y , determina cuantos bisiestos hay entre ellos Ejemplo: Introduce el comienzo del intervalo: 1945 - (x=1945) Introduce el comienzo del intervalo: 2300 - (y=2300) Proceso: acumular cuantos años bisiestos hay entre 1945 y 2300 Resultado: "El número de años bisiestos entre los proporcionados es: w"</p>						

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cuestión 11.-
0	0,25	R=0,5	1		
<p>Dado un <u>vector</u> v de enteros y un <u>valor</u> x proporcionado por el usuario determina si existe un amigo del proporcionado.</p>					
<p>Ejemplo: $v=[12,284,45,2,4,4,7,12]$ $x=220$ Proceso: Determinar si hay un v[índice] que se amigo con 220 Resultado: En la posición 2 hay un amigo de 220</p> <p>Dos números amigos son dos enteros positivos a y b tales que a es la suma de los divisores de b y b es la suma de los divisores de a. (la unidad se considera divisor, pero no lo es el mismo número). Un ejemplo es el par (220, 284), ya que:</p> <ul style="list-style-type: none"> • los divisores propios de 220 son 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 y 110, que suman 284 • los divisores propios de 284 son 1, 2, 4, 71 y 142, que suman 220 					

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cuestión 12.-
0	0,25	0,75	R=1	1,25	
<p>Opción 1.- Enunciado: Dada una frase realiza su encriptación de forma que cada letra mayúscula o minúscula se sustituya por la correspondiente 2 posiciones siguientes en el alfabeto. Ejemplo: frase="juanlop@um.es" Proceso: $a \rightarrow c$, $b \rightarrow d$... $x \rightarrow z$, $y \rightarrow a$, $z \rightarrow b$ $a \rightarrow c$, $b \rightarrow d$... $x \rightarrow z$, $y \rightarrow a$, $z \rightarrow b$ Resultado: [Se leen una cadena] La frase encriptada es: lwcpnqr@wo.gu</p> <ol style="list-style-type: none"> 1. Considera en uso de las funciones: strlen() longitud de una cadena, ord() código ascii de un carácter y ascii(e) el carácter asociado a un código ascii 2. El código ascii de la 'a' es 97 y de la 'z' es 122. 3. El código ascii de la 'A' es 65 y de la 'Z' es 90. 					
<p>Opción 2.- Enunciado: Dada una frase realiza su encriptación utilizando la frase de encriptación. Dado un carácter 'x' si aparece en la frase de encriptación en la posición fencrip[i] se sustituye por el carácter fencrip[i+1] (si la posición es strlen(frase) se sustituirá por fencrip[1]), si no aparece se deja el mismo. Ejemplo: [Se leen dos cadenas] frase=juanlop@um.es y fencrip="jkuhet" Proceso: El carácter 'j' si está en fencrip por lo que se sustituye por k .. juanlop@um.es Resultado: La frase encriptada es: khanlop@um.ts</p>					