

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"/>	<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.						
<pre>/* Dados un vector v1 de enteros y un valor x determinar el número de apariciones*/ var secuencia : v[10] numerico; numero : numerico=0; indice : numerico=1; inicio /* Suponemos que leemos el vector*/ imprimir ("\nIntroduce el numero a burca= ") leer(x) mientras (indice<10) { si(v[indice]==x) { numero=numero+2 } indice=indice+1 } imprimir("Aparece",numero,"veces") fin</pre> 						

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 2.-
	0,0	0,25	0,50	0,75		
<p>Corrige los errores no sintácticos que provoca que el código que se proporciona a continuación no realice la acción indicada.</p>						
<pre>/* Dado una secuencia de número leídos y almacenados en un vector A y un número leído determinar si dicho número se encuentra o no en el vector.*/ var secuencia : secuencia[10] numerico; numero : numerico=0; indice : numerico=1; end : numerico=0 inicio mientras (indice<=alen(secuencia)) { secuencia[indice]=random(100) indice=indice+1 } imprimir ("\nIntroduce el numero a burca= ") leer(numero) mientras (indice<=10 and end==0) { si(secuencia[indice]<>numero) { end=1 indice=indice+1 } } si(end==1) { imprimir("EL NUMERO SE ENCUENTRA EN EL VECTOR") } sino imprimir("EL NUMERO NO SE ENCUENTRA EN EL VECTOR") } fin</pre> <div style="border: 2px dashed black; height: 200px; width: 100%;"></div>						

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Cuestión 3.-
0,0	0,25	0,50	0,75		
<p>Corrige los errores no sintácticos que provoca que el código que se proporciona a continuación no realice la acción indicada.</p>					
<pre>/* Leer una secuencia de solo números pares y mostrar la suma de los mismos. (c) lopezquesada@iessanjuanbosco.es */ var numero : numerico suma : numerico indice : numerico inicio imprimir ("\nCuantos valores vas a leer:\n") leer (cantidad) indice=1; mientras (indice<cantidad) { imprimir ("\nIntroduce un valor\n") leer (numero) si (numero%2==0) { suma=suma+numero } indice=indice+1 } imprimir ("\nLa suma realizada es: ", suma) fin</pre> 					

	<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.

```
/* 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 6.-
	0,0	0,25	0,50	0,75		
Rellena el espacio para completar el código que se proporciona.						
<pre>/* * Dado un vector v1 y un valor x determinar cuántos hay tal que v[i]<x, i:1..alen(v). */ var secuencia : vector[10] numerico; indice : numerico=1; cantidad : 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 aparece: ", cantidad) 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"/>	<input type="checkbox"/>		Cuestión 7.-
	0,0	0,25	0,50	0,75		
Determina el enunciado del código que se presenta.						
<pre>/* Enunciado: */ var a : vector[*] numerico; b : numerico=0; c : numerico=0; i : numerico=1; d : numerico; e: numerico inicio imprimir(".....") leer (d) dim(a,d) mientras (i<=d) { a[i]=random(100) i=i+1 } i=1 mientras (i<=d) { c=c+a[i] i=i+1 } e=c/d imprimir(".....", e) fin</pre>						

	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 9.-
	0,0	0,25	0,50	0,75		

Determina el enunciado del código que se presenta.

/*

Enunciado:

*/

```
var
a : numerico;
b : numerico=0;
c : numerico=1;
inicio
    imprimir(".....")
    leer (a)
    mientras (c<a)
    {
        si (a%c==0)
        {
            b=b+c
        }
        c=c+1
    }
    si (a==b)
    {
        imprimir ("????????????????????????????????????")
    }
    sino
    {
        imprimir ("????????????????????????????????????")
    }
fin
```

--	--	--

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"/>	
	0	0,25	R=0,5	1	Cuestión 10.-
<p>Opción 1.- Dado dos enteros y las funciones $\min(x,y)$ y $\max(x,y)$ –que devuelven el menor y mayor respectivamente de x,y-, muestra la suma de los enteros comprendidos entre los proporcionados:</p> <p>Ejemplo: 25 52 Proceso: $25+26+27\dots+50+51+52$ Resultado: la suma de esa secuencia</p> <p>Opción 2.- Dado un número entero visualizar su reverso: Nota: <i>La potencia se calcula con la función $\text{potencia}(x,y)$ que recibe dos numéricos y devuelve x^y</i> <i>Dado un entero el número de dígitos que tiene la proporcionaría la función $\text{cdigit}(x)$ que recibe un numérico y devuelve dicha información.</i></p> <p>Ejemplo: 25893 Proceso: invertir los dígitos Resultado: 39852</p> <p>Opción 3.- Dada una secuencia de números mostrar la mayor longitud de la sub-secuencia de números negativos:</p> <p>Ejemplo: [1,2,-5,-10,20,-50,-100,-200,500] Proceso: La mayor sub-secuencia de números negativos es -50,-100 y -200 Resultado: La mayor longitud de negativos es 3</p> <p>Opción 4.- Considérense las proposiciones o enunciados lógicos siguientes:</p> <ul style="list-style-type: none"> • p: Es divisible entre 4 • q: Es divisible entre 100 • r: Es divisible entre 400 <p>La fórmula lógica que se suele usar para establecer si un año es bisiesto es cuando $(p \vee \neg q) \wedge r$ es verdadera</p> <p>Ejemplo: El usuario teclea 2012 Proceso: Determinar si es bisiesto Resultado: 2012 es un año bisiesto</p>					

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Cuestión 11.-
0	0,25	R=0,5	1		
<p>Opción 1.- 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. <i>Los divisores propios de un número son los que están comprendidos entre (1..N-1)</i></p> <p>Ejemplo: $v1=[23,5,67,5]$ Proceso: Se determinará de $v[i]$ es un numero abundante o deficiente.</p> <p>Opción 2.- Dado una cantidad monetaria y los tipos de monedas del país determinar como fraccionarías dicha cantidad</p> <p>Ejemplo: 545 euros y $m=[1,2,5,10,20,50,100,200,500]$ Resultado: 1 de 500, 2 de 20, 1 de 5,</p> <p>Opción 3.- Dado un vector de enteros de termina cuantas parejas (vector[i],vector[i+1]) cumplen que $vector[i]==vector[i+1]$</p> <p>Ejemplo: $v=[12,45,45,2,4,4,7,12]$ Proceso Encontradas (45,45), (4,4) Resultado: Hay 2 parejas</p> <p>Opción 4.- Dado un vector v1 y dos valores que representan dos índices, determinar cuántos de los elementos de entre estos dos índices o posiciones del vector son menores que el primero $v[0]$. <i>Debe considerarse problemas relativos a posicionamiento fuera del rango del vector.</i></p> <p>Ejemplo: $v=[12,45,450,2,70,56,-9,12]$ $i=2,j=4$ Proceso $v[2]=45,v[3]=450,v[4]=2$ Resultado: Hay 1 elemento menor que el primero $v[0]=12$</p>					

<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		
Dado dos vectores v1 y v2 de tamaño [*] crear un tercero v3[alen(v1)+alen(v2)] con el contenidos de v1 y v2						
Ejemplo: v1=[23,3,7,5,10] v2=[2,5,67,5] Resultado: v3=[23,3,7,5,10, 2,5,67,5]						
Opcional (3 puntos): Dado dos vectores v1 y v2 de tamaño [*] ordenados crear un tercero v3[alen(v1)+alen(v2)] con el contenidos de v1 y v2 ordenado. Suponemos que el usuario introduce v1 y v2 ya ordenados. Ejemplo: v1=[3,7,15,100] v2=[3,5,67] Resultado: v3=[3,3,7,5,15,67,100]						