

Informática
Nivel medio
Prueba 1

Viernes 2 de noviembre de 2018 (tarde)

1 hora 30 minutos

Instrucciones para los alumnos

- No abra esta prueba hasta que se lo autoricen.
- Sección A: conteste todas las preguntas.
- Sección B: conteste todas las preguntas.
- La puntuación máxima para esta prueba es **[70 puntos]**.

Sección A

Conteste **todas** las preguntas.

1. Resuma lo que significa una red de computadoras. [2]

2. (a) Resuma lo que significa un sistema de gestión de bases de datos. [2]
(b) Resuma **una** ventaja de utilizar pruebas beta antes de lanzar un producto nuevo. [2]

3. La observación directa es una técnica que utiliza un analista de sistemas para determinar los requisitos del usuario para actualizar un sistema informático.
(a) Identifique **una** ventaja de la observación directa. [1]
(b) Identifique **una** desventaja de la observación directa. [1]

4. Elabore un diagrama lógico para la siguiente expresión.
$$\text{NOT } A \text{ OR } (A \text{ AND } B)$$
 [3]

5. Una empresa internacional está en el proceso de mudar su oficina matriz de Europa a Asia.
(a) Identifique **dos** posibles problemas de compatibilidad que podrían surgir en la migración de datos. [2]
(b) Resuma cómo una red privada virtual (VPN – *virtual private network*) permitirá que los empleados que estén en Europa se comuniquen con la oficina matriz de Asia. [2]
(c) Resuma **una** cuestión social asociada con este proceso. [2]

6. Describa cómo se transmiten datos mediante la conmutación de paquetes. [4]

7. Elabore una tabla de seguimiento del siguiente algoritmo.

```
A = 3  
B = 7  
loop while B >= A  
  A = A + 1  
  output(B - A)  
  B = B - 1  
end loop
```

 [4]

Sección B

Conteste **todas** las preguntas.

8. (a) (i) Distinga entre la memoria de acceso aleatorio (RAM – *random access memory*) y la memoria de solo lectura (ROM – *read only memory*). [3]
- (ii) Resuma la función de un sistema operativo en la gestión de la memoria principal. [2]
- (b) Explique las funciones del bus de datos y el bus de direcciones en el ciclo de instrucción de la máquina. [4]
- (c) (i) Indique cómo los datos almacenados en el siguiente byte se representarán en hexadecimal. [1]

0	1	0	1	1	1	1	0
---	---	---	---	---	---	---	---

- (ii) Indique cuántos enteros se podrían representar en este byte. [1]
- (iii) Resuma por qué no se podría utilizar este byte para representar caracteres tales como los usados en chino. [2]
- (d) Elabore una tabla de verdad con dos variables de entrada. Si las variables de entrada son iguales, el valor de la variable de salida deberá ser Verdadero. Caso contrario, deberá ser Falso. [2]
9. (a) Resuma la necesidad de los lenguajes de alto nivel. [2]
- (b) Explique **dos** beneficios de utilizar subprocesos en un programa informático. [4]
- (c) Identifique **tres** características de una colección. [3]

La colección `NUMEROS` ya existe y almacena números reales.

- (d) Elabore en pseudocódigo un algoritmo, usando los métodos de acceso de una colección, que revisará cada elemento de la colección `NUMEROS`, y contará cuántos elementos almacenados en la colección están en el intervalo $[-1,1]$.

La respuesta final deberá indicarse como salida. [6]

Véase al dorso

10. El siguiente método `calcBMI()` acepta la altura (H) de una persona en metros (m) y el peso (W) en kilogramos (kg) y produce como salida el índice de masa corporal (BMI – *Body Mass Index*).

```
calcBMI(H, W)
  X = H * H
  B = W / X
  return B
endcalcBMI
```

Boris pesa 104 kg y su altura es 2,00 m. Su BMI puede calcularse al llamar al método `calcBMI()` de la siguiente manera

```
BorisBMI = calcBMI(2.00, 104).
```

- (a) Indique el valor de la variable `BorisBMI`. [1]

Una persona puede pertenecer a una de las siguientes cuatro categorías de peso:

BMI	Categoría de peso
menos de 18,5	menos del peso normal
desde 18,5 a menos de 25,0	peso normal
desde 25,0 a menos de 30,0	sobrepeso
mayor o igual a 30,0	obeso

- (b) Utilice pseudocódigo para elaborar un algoritmo que acepte el BMI de una persona y produzca como salida la categoría de peso a la que pertenezca. [4]

(Esta pregunta continúa en la página siguiente)

(Pregunta 10: continuación)

Los datos de un grupo de adultos, su altura (en metros) y su peso (en kg) se almacenan en tres matrices unidimensionales.

	NOMBRE		PESO (kg)		ALTURA (m)
[0]	Annie	[0]	52,40	[0]	1,56
[1]	Boris	[1]	100,00	[1]	2,00
[2]	Hugh	[2]	105,00	[2]	2,03
[3]	Paul	[3]	61,00	[3]	1,75
[4]	Robby	[4]	88,00	[4]	1,80
...
...
[29]	Zara	[29]	68,00	[29]	1,71

En donde:

NOMBRE es una matriz unidimensional que contiene nombres (que están actualmente ordenados en orden alfabético).

PESO es una matriz unidimensional que contiene los pesos en kilogramos.

ALTURA es una matriz unidimensional que contiene las alturas en metros.

Por ejemplo,

NOMBRE[0] es Annie.

Pesa 52,40kg y este valor se incluye en PESO[0].

ALTURA[0] es 1,56 que representa la altura de Annie en metros.

- (c) Indique el nombre de la persona cuya altura se incluye en ALTURA[3]. [1]
 - (d) (i) Identifique **un** motivo por el cual no se puede usar un algoritmo de búsqueda binaria para encontrar el nombre de la persona cuya altura se da. [1]
 - (ii) Describa cómo se podría obtener como salida el nombre de la persona cuya altura se da. [2]
 - (e) Elabore un algoritmo que producirá como salida los nombres de todas las personas cuyo BMI sea mayor que el promedio del BMI de este grupo. [6]
- Debe llamar al método `calcBMI()` en su respuesta.