

Informática Examen Teoría y Práctica (julio de 2020)

El examen se debe resolver exclusivamente con medios propios. Está permitido consultar la ayuda oficial de Python <http://www.python.org>, pero no valerse de ningún otro tipo de web o de documentación, ni apuntes, ni guiones, ni compañeros, etc.

Se debe subir al final de la prueba las respuestas, preferiblemente en un solo fichero editado por ordenador, a FaiTIC en el apartado habilitado para tal fin. (El formato o nombre del fichero no importa.)

El tiempo para la prueba son solamente **2 horas**, controla el tiempo, no habrá prólogo ya que el diseño del examen lo tiene en cuenta.

Apartado I (más teórico)

Argumenta brevemente con palabras propias sobre las siguientes dos aspectos (1 punto):

- (a) ¿Cómo funcionan principalmente los algoritmos de ordenación Mergesort y Bubblesort visto en clase? ¿Cuáles son sus principales diferencias? ¿Cuándo usarías cuál en una aplicación que tiene datos por ordenar?
- (b) Razona brevemente sobre las diferencias y semejanzas entre listas, tuplas, conjuntos, y diccionarios de Python (piensa como tomarías la decisión cuál escoger en una situación concreta...).

A continuación se incluyen 10 preguntas tipo test. Cada pregunta tiene una sola respuesta correcta. La puntuación de las preguntas tipo test se obtendrá sobre 3 puntos aplicando la fórmula:

$$(\text{aciertos} - \text{fallos}/4) \cdot 3/10$$

Tipo test 1: Señala lo correcto. Los nombres de variables en python

- (a) pueden empezar con un dígito si sigue una barra baja.
- (b) pueden coincidir con palabras reservados ya que se deducen desde el contexto en un lenguaje interpretado.
- (c) pueden contener una mezcla de letras en minúsculas y mayúsculas.
- (d) tienen que empezar con una letra.

Tipo test 2: Para los prefijos con i (como gibi, Gi, o exbi, Ei) es correcto:

- (a) Indican que se mide en bits en vez de bytes.
- (b) Avisan que se trata de prefijos internacionales para simplificar el comercio de productos tecnológicos entre países.
- (c) Indican potencias de 2 en vez de potencias de 10.
- (d) Es una jerga informática para distinguir capacidades de almacenamiento lógica (sistema operativo) del físico (disco duro).

Tipo test 3: ¿Cuál es el orden cronológico correcto (de más antiguo a más moderno) de estas personas históricas relacionadas con la computación?

- (a) John von Neumann, Alan Turing, Konrad Zuse, Charles Babbage
- (b) Konrad Zuse, Charles Babbage, John von Neumann, Alan Turing
- (c) Charles Babbage, Alan Turing, Konrad Zuse, John von Neumann
- (d) Charles Babbage, Konrad Zuse, Alan Turing, John von Neumann

Tipo test 4: Un sistema de encriptación asimétrico, es decir, con claves privados y públicos, ofrece las siguientes propiedades (señala la **correcta**):

- (a) la autenticidad del remitente, como en firmas electrónicas
- (b) un mensaje encriptado con la clave pública solo puede desencriptar la propietaria de la clave privada, como en estándares de encriptación
- (c) la clave privada se puede deducir a partir de la clave pública por eso hay que mantenerla en secreto, como en comunicación con ssh
- (d) la anonimidad de los participantes, como en el darknet

Tipo test 5: ¿Cuál de las siguientes afirmaciones es correcta?

- (a) Un algoritmo/método Monte Carlo siempre calcula un resultado correcto.
- (b) Un algoritmo/método Monte Carlo tiene un tiempo de cálculo probabilístico.
- (c) Un algoritmo/método Monte Carlo aproxima un resultado correcto con cierta probabilidad.
- (d) Un algoritmo/método Monte Carlo nunca calcula un resultado correcto.

Tipo test 6: Un inventor dice que con su tecnología se puede empaquetar 100 GB de memoria con re-escritura en un área de 1 mm^2 . ¿Qué piensas?

- (a) Es tecnología punta para el año 2020 respecto a densidad.
- (b) Es un avance importante y ya disponible a nivel tecnológico comparado actualmente en el año 2020 respecto a densidad.
- (c) Tal densidad ya fue superado hace años con algunas tecnologías experimentales.
- (d) Es una falsa noticia ya que es físicamente imposible de alcanzar.

Tipo test 7: ¿Cuál de las siguientes afirmaciones es correcta?

- (a) El protocolo IPv6 usa más servidores raíz DNS que el protocolo IPv4.
- (b) El protocolo IPv6 solamente se usa para dispositivos móviles.
- (c) La implantación de IPv6 proporciona más identificadores para dispositivos conectados al internet comparado con el IPv4.
- (d) El protocolo IPv6 sirve para redes privadas y no para redes públicas.

Tipo test 8: Señalar la opción que ordena de menor a mayor amplitud las siguientes redes de computadoras:

- (a) LAN < CAN < MAN < WAN
- (b) LAN < MAN < CAN < WAN
- (c) LAN < WAN < CAN < MAN
- (d) CAN < LAN < MAN < WAN

Tipo test 9: Suponer un programa en python3 que contiene la siguiente línea de código que se ejecuta en algún momento

```
a[0]="hola"
```

El programa no da ningún error cuando se ejecuta. Señala lo correcto:

- (a) a puede ser una tupla.
- (b) a puede ser un diccionario.
- (c) a puede ser un conjunto.
- (d) ninguna de las tres anteriores.

Tipo test 10: ¿En qué condiciones es aconsejable la implementación recursiva de un algoritmo antes que la no recursiva?

- (a) Cuando la naturaleza del algoritmo es recursiva.
- (b) Siempre: la versión recursiva es más eficiente.
- (c) Cuando el sistema tiene poca memoria.
- (d) Nunca.

Apartado II (más práctico)

Problema de programación 1: [1 Punto(s)] Realizar un programa en python que lea un DNI (con la letra, por ejemplo 12345678X) y muestre por pantalla los dígitos de las dos últimas posiciones.

Problema de programación 2: [1.5 Punto(s)] Realizar un programa en python que lea un número entero entre 0 y 30. Si es menor o igual que 10, muestra por pantalla *es pequeño*, si está entre 11 y 20, muestra *es mediano*. Y si es mayor que 20, muestra *es grande*.

Problema de programación 3: [1.5 Punto(s)] Escribe una función en python que reciba dos argumentos, una lista de números (posiblemente con elementos repetidos) y un número. La función debe devolver la cantidad de veces que aparece dicho número en la lista. Se debe visualizar dicho resultado por pantalla. Por ejemplo, dada la lista `[1, 1, 2, 2, 3, 5, 5, 0, 0, 0, 2, 6, 1, 1, 1, 1]` y el valor 1. La función debe devolver: 6

Problema de programación 4: [2 Punto(s)] Escribir un programa que, utilizando funciones, permita calcular una sucesión numérica hasta cierta cantidad de elementos. El comportamiento es el siguiente:

- Se debe solicitar por teclado un número comprobando que sea positivo. Este número indicará la cantidad de términos de la sucesión que se van a calcular y mostrar.
- Si se introduce un valor entre 0 y 3 (ambos incluidos), se mostrarán siempre los tres primeros elementos de la sucesión.
- La sucesión se construye de la siguiente forma:
 - El primer elemento de la sucesión es aleatorio entre 0 y 9 (ambos incluidos)
 - El segundo elemento siempre es 1
 - El tercer elemento siempre es el primer elemento + 1
 - A partir de tercer elemento, las siguientes posiciones se calculan de la siguiente forma:
 - Si el término anterior es impar, se obtiene sumando los dos términos anteriores;
 - Si es término anterior es par, sumando los tres términos anteriores.

Por ejemplo:

```
Se introduce el valor 10.  
[8, 1, 9, 10, 20, 39, 59, 98, 196, 353]
```

```
Se introduce el valor 7.  
[2, 1, 3, 4, 8, 15, 23]
```

```
Se introduce el valor 0.  
[3, 1, 4]
```

```
Se introduce el valor 4  
[1, 1, 2, 4]
```