

Examen Final

Fundamentos de Informática

18 de Enero de 2021

Parte Escrita I 30 m [20%]

Nombre: _____

Justificar las respuestas.

No se puede usar lápiz. Contestar en el espacio de cada pregunta.

1. [2%] Explica la diferencia de un lenguaje de programación interpretado y otro compilado y pon un ejemplo de cada uno.

En un lenguaje interpretado, al ejecutar un programa, cada línea del mismo se traduce y se ejecuta inmediatamente, no pudiendo separarse el proceso de traducción con el de ejecución.

Sin embargo, en un lenguaje compilado, un programa se traduce por completo, creando un programa objeto una sola vez. Ese programa objeto escrito en código máquina puede ejecutarse cuantas veces se quiera. Aquí, el proceso de traducción está completamente separado del proceso de ejecución.

Python es un lenguaje interpretado y FORTRAN compilado.

2. [2%] Si el reloj de la CPU de un computador es de 2 GHz y se quiere ejecutar un programa de Python de 10 instrucciones, ¿Cuánto tiempo tardará en ejecutarse?

Sea $1ip \approx 10im$ y $1im \approx 10ic$, donde

ip = instrucción python

im = instrucción máquina

ic = instrucción elemental o ciclo

$2\text{ GHz} = 2 \cdot 10^9 \text{ ciclos/s}$. El programa de 10 ip equivale a unos 10^3 ciclos.

Por tanto, tardará $10^3 / 2 \cdot 10^9 = 0.5 \mu\text{s}$ en ejecutarse.

3. [2%] ¿Quién gestiona el tiempo que el procesador dedica a los diferentes programas en ejecución?

El sistema operativo.

4. [2%] ¿Cómo se llama el módulo de Python específico para trabajar con bases de datos? ¿y para química?

sqlite3 y chempy.

5. [2%] En una hoja de cálculo, la celda C3 es la diferencia entre la celda B3 y la F10. Escribe la fórmula adecuada usando referencia relativa a B3 y absoluta a F10.

$C3=B3-\$F\10

Si se usa el cuadro de arrastre de C3 hacia abajo, en la celda C4, ¿qué fórmula se ha copiado?

$C4=B4-\$F\10

6. [10%] ¿Qué tarea realiza la función `son_cuadrones_pares`? ¿qué recibe y qué devuelve? Explica cómo opera con dos ejemplos.

```
def son_cuadrones_pares(x,y):  
    if comprueba(x+y) and comprueba(abs(x-y)):  
        return True  
    return False  
def comprueba(a):  
    if (int(a ** 0.5)) ** 2 == a:  
        return True  
    return False
```

La función `son_cuadrones_pares` comprueba si dos números enteros cumplen que su suma y resta en valor absoluto son cuadrados perfectos.

Recibe dos enteros y devuelve un booleano: True cuando son cuadrones pares y False cuando no lo son. Para ello, la función `son_cuadrones_pares` invoca a su vez, a la función `comprueba`. Esta función recibe un número y devuelve un booleano: True cuando el entero de la raíz cuadrada de ese número al elevarlo al cuadrado es igual que el número recibido y False cuando son distintos.

Ej1: 10 y 26 son cuadrones pares.

En este caso, la función `comprueba` actúa dos veces, la primera recibiendo 36 y devolviendo True y la segunda recibiendo 16 y devolviendo True. Como True and True es True, la función `son_cuadrones_pares` devuelve True. En la primera llamada a `comprueba` se recibe 36, el entero de la raíz cuadrada de 36 es 6 y al elevarlo al cuadrado coincide con 36 y lo mismo ocurre con 16.

Ej2: 5 y 9 no son cuadrones pares.

En este caso, la función `comprueba` solo actúa una vez, recibe 14, se calcula el entero de la raíz cuadrada de 14 que es 3 y se eleva al cuadrado, 9. Como 9 es distinto de 14 se devuelve False. Por la evaluación con cortocircuitos no se invoca a `comprueba` con valor 4, pues la condición en `son_cuadrones_pares` es inevitablemente False y se devuelve False.

Examen Final

Fundamentos de Informática

18 de Enero de 2021

Parte Escrita II 60 m, [30%]

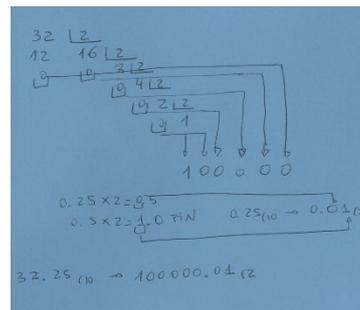
Nombre: _____

Justificar las respuestas.

No se puede usar lápiz. Contestar en el espacio de cada pregunta.

1. [3%] Escribe 32.25 en binario.

100000.01



2. [4%] ¿Cuántos bits por símbolo se necesitan como mínimo para representar un conjunto de 40 símbolos?

6 bits por símbolo, pues con 6 bits se pueden formar $2^6=64$ combinaciones distintas de ceros y unos tomados de 6 en 6, 40 de las cuales se usarían para representar los 40 símbolos del conjunto. Sobran 24 combinaciones.

5 bits por símbolo es insuficiente, pues $2^5=32 < 40$.

3. [4%] ¿Cuánto espacio (en MB) ocupa en un computador, sin comprimir, 1 minuto de un programa de radio? Tomar como frecuencia de muestreo 22.05 KHz y precisión de 8 bits/muestra

1m = 60 s

22.05 KHz = 22.05 10^3 muestras/s

22.05 10^3 muestras/s 60s * 8 bits/muestra = 6*0.2205 10^6 8 bits = 1.3 10^6 1B/(2^{20} B)*MB (\approx 1.3 MB)

4. [4%] Escribe una función documentada que reciba el radio y altura de un cono y calcule y devuelva su volumen. $V_{\text{cono}} = \frac{1}{3}\pi r^2 h$

>>>vol_cono(0.5,0.8)

0.21

```
from math import pi
```

```
def vol_cono(radio,altura):
```

```
    ''' recibe radio y altura y calcula el volumen del cono '''
```

```
    volu=1/3*pi*radio**2*altura
```

```
    return volu
```

5. [5%] Completa los huecos `._._._` en el programa siguiente para evaluar un polinomio en un valor de x y explica un ejemplo de ejecución.

```
grado=int(input("grado del polinomio "))
lc=[]
for i in range(grado+1):
    coe=float(input("coeficiente del término en x**"+str(i)+" : "))
    lc.append(coe)
x=float(input("x "))
px= 0
for i in range(grado,-1,-1):
    px+= lc[i]*x**i
print("Valor del polinomio para x=",x," es ",px)
```

Ejemplo:

Sea el polinomio $-2x^3+4x^2+7$ y $x=-1$, el programa pide el grado, se teclea 3. En un bucle se piden los coeficientes del polinomio, desde el término independiente hasta el coeficiente que acompaña a x^3 , añadiéndolos a una lista `lc`. `lc = [7,0,4,-2]`. Se pide x, se teclea -1 y se escribe en pantalla: Valor del polinomio para $x=-1.0$ es 13.0

6. [5%] ¿Qué tarea realiza este programa?

Ordenar ascendentemente una lista de números, usando el algoritmo de la burbuja.

¿Para qué sirve cada bucle?

El bucle externo, con índice `i`, informa de la cantidad de parejas a tratar para ordenarse.

El bucle interno, con índice `j`, informa de la pareja concreta a ordenar, intercambiando sus valores, si el elemento de la izquierda es mayor que el de la derecha.

```
v=[45,26,9,3,1]
np=len(v)-1
for i in range(np,0,-1):
    for j in range(0,i):
        if v[j] > v[j+1]:
            v[j+1],v[j]=v[j],v[j+1]
    print(i,v)
```

¿Qué se escribe por pantalla?

- 4 [26, 9, 3, 1, 45] En esta iteración, el número más grande se coloca al final (posición 4).
3 [9, 3, 1, 26, 45] En ésta, el segundo número más grande se coloca (posición 3 de la lista)
2 [3, 1, 9, 26, 45] Aquí, el tercer número más grande de la lista se coloca (posición 2).
1 [1, 3, 9, 26, 45] Finalmente, el cuarto número más grande se coloca (posición 1).

7. [5%] Sea el archivo *GasesNobles.txt*:

```
He;Helio;2;4.0026;2
Ne;Neón;10;20.1797;2,8
```

Y el programa:

```
fe=open('GasesNobles.txt','r')
d_gn={}
Lista=[]
for linea in fe:
    Lista=linea[:-1].split(";")
    d_gn[Lista[0]]=Lista[1:]
for elem in d_gn:
    print(elem,d_gn[elem])
```

Explica cada sentencia del programa y qué se escribe por pantalla al ejecutar.

1. Se abre el archivo *GasesNobles.txt* para leerlo volcando en la variable *fe* la “conexión” entre el programa y el archivo de texto.
2. Se inicializa *d_gn* como un diccionario vacío.
3. Se inicializa *Lista* como una lista vacía.
4. Se ejecuta un bucle con dos iteraciones. En cada iteración la variable *linea* almacena en un string cada línea del archivo abierto. En la primera iteración:

linea= 'He;Helio;2;4.0026;2\n' donde \n es el carácter retorno de carro o fin de línea

linea[:-1]= 'He;Helio;2;4.0026;2' elimina el último carácter de la cadena

'He;Helio;2;4.0026;2'.split(";") el método *split* recibe la cadena de la izquierda y usando como separador ; construye una lista con ello.

Lista=['He', 'Helio', '2', '4.0026', '2']

Se construye un elemento del diccionario *d_gn* con clave el símbolo del elemento químico y valor la lista restante de 4 elementos.

5. Se ejecuta un bucle con dos iteraciones que recorre el diccionario y lo escribe por pantalla, cada elemento en una línea distinta.

Por pantalla se escribe:

```
He ['Helio', '2', '4.0026', '2']
```

```
Ne ['Neón', '10', '20.1797', '2,8']
```