Escuela politécnica Nacional

Integrantes:

Oscar Pullay

Mario Nuñes

Cuestionario:

**Nivel Bajo**

1. **Que da como resultado la siguiente línea de código 2\*\*3**
   * 6
   * 5
   * 9
   * 8

Justificación: el símbolo \*\* de exponente por los que si 2 elevamos a la 3 da como resultado 8.

1. **Que hace el operador + y el \* en una cadena:**
   * El operador + une mientras que el \* imprime un elemento de una lista
   * El operador + imprime el ultimo carácter mientras que el \* imprime un elemento de una lista
   * El operador + concatena mientras que el \* imprime repeticiones de texto

Justificación: El operador + concatena (une cadenas) y el \* duplica las cadenas.

1. **Tengo declarada la variable texto=”Hola” y ejecuto la siguiente line código texto[-1]que se imprimirá en la pantalla:**
   * Carácter en la posición 1
   * El primer carácter
   * El antepenúltimo carácter
   * Ultimo carácter

Justificación: [] entre estas imprime la posición del texto y si le ponemos -1 comensara desde el final del texto.

1. **Que imprime si ejecutamos las estas dos líneas de de código s=”hola ” y len(s)**
   * Hola
   * 5
   * 4
   * “hola”

Justificación: la función len devuelve el tamaño de la cadena.

1. **Conque función podemos reemplazar en una cadena.**
   * Script
   * Lower
   * Capitalize
   * Replace

Justificación: la función replace permite reemplazar recibiendo dos parámetros el texto a ser reemplazar y el texto que va a reemplazar.

1. **Cuál de las siguientes funciones convierte el texto en minúsculas:**
   * Capitaliza
   * Upper
   * Lower
   * Script

Justificación: función que convierte el texto en minúsculas

1. **Que podemos hacer con la función Split:**
   * Unir
   * Separar
   * Pone la primer letra en mayúsculas
   * Quita espacios en blanco de los lados

Justificación: esta función permite separar una cadena recibiendo como parámetro el carácter que servirá para separar dicha cadena.

1. **Al trabajar con listas cual es el método para añadir al final de la lista:**
   * Replace
   * Len
   * Script
   * Append

Justificación: el append() añade elementos al final de una lista mandándole como parámetro el elemento a añadir.

1. **En Git Hub Los cambios realizados de un proyecto se llama:**
   * Push
   * Add
   * Commits
   * Pull

Justificación: En Git Los cambios realizados se llaman commits, además este permite explicar la razón del cambio realizado.

1. Que es un Branching:
   * Significa descargar los últimos cambios del repositorio
   * Significa hacer efectivo los commints
   * Proceso para inicializar un repositorio
   * Es la mane a de trabajar en diferentes versiones de un repositorio a la vez

Justificación: Es la mane a de trabajar en diferentes versiones de un repositorio a la

**Nivel medio y alto**

1. **Cuál es la librería que debemos importar para trabajar con números aleatorios:**
   * Pygame
   * Time
   * Random
   * Os

Justificación: la librería random permite trabajar con números aleatorios.

1. **Como definimos una clase en Python:** 
   * Class nom\_clase()
   * Class nomCalse:
   * Clase nom \_clase:
   * Clas nomClase()

Justificación: la palabra reservada para crear una clase en python es la palabra class seguido de un nombre y dos puntos.

1. **Como instanciar una clase.**
   * Nomvariable=nomClase
   * Nomvariable=nomClase()
   * Nomvariable=nomClase:

Justificación: se declara una variable de tipos de la clase creada.

1. **Que imprime la siguiente línea de código: print(time.localtime())**
   * Impresión de los segundos del sistema
   * Impresión de los hora del sistema
   * Impresión de una estructura completa de la fecha del sistema

Justificación: imprime la estructura completa de la fecha del sistema

1. **Python es:**
2. Un lenguaje de programación interpretado orientado a objetos de alto nivel
3. Un lenguaje de programación compilado no orientado a objetos de bajo nivel
4. Un lenguaje máquina ejecutado en una máquina virtual java
5. Un lenguaje de programación inventado por el creador de Linux

R) a   
Python es un lenguaje de programación de uso general, orientado a objetos e interpretado.

1. **Después de una declaración try para capturar una excepción usaremos:**
2. Throw
3. raise
4. catch
5. except

R) d  
Primero, se ejecuta el bloque try (el código entre las declaración try y except).  
Si no ocurre ninguna excepción, el bloque except se saltea y termina la ejecución de la declaración try.

1. **¿Cómo se define una variable asignándole un valor?**
2. v = 0
3. number v = 0
4. int v = 0
5. var v = 0

R) a  
Para definir una variable en Python solo debemos asignarle el valor, en dicho momento se define el tipo de variable según el valor asignado (Python es un lenguaje sensible a mayúsculas y minúsculas, por lo que debemos respetar esta característica)  
**¿Cómo se abre un archivo para leerlo?**

1. f = open("archivo.txt", "read")
2. f = open("archivo.txt", "r")
3. f = open("archivo.txt", "w")
4. f = open("read", "archivo.txt")

R) b  
Si el archivo de texto se encuentra en la misma carpeta que nuestro programa no necesitamos indicar el path (camino). Los modos de apertura del archivo de texto pueden ser:

'r' Abre el archivo para lectura (debe existir el archivo) open(nombre del archivo,modo)

1. **¿Cuál es la forma correcta de escribir un bucle for?**
2. for a in range(0, 3):
3. for a in range(0..3)
4. for(a in range[0..3])
5. for(a=0; a<3; a++)

R)a a  
En general, un bucle es una estructura de control que repite un bloque de instrucciones. Un bucle for es un bucle que repite el bloque de instrucciones un número prederminado de veces. El bloque de instrucciones que se repite se suele llamar cuerpo del bucle y cada repetición se suele llamar iteración.

La sintaxis de un bucle for es la siguiente:

for variable in elemento iterable (lista, cadena, range, etc.):

cuerpo del bucle

1. **La forma correcta de escribir una función es:**
2. define nombrefuncion()
3. nombrefuncion: function()
4. def nombrefuncion():
5. function nombrefuncion()

R)c c  
En Python, la definición de funciones se realiza mediante la instrucción def más un nombre de función descriptivo -para el cuál, aplican las mismas reglas que para el nombre de las variables- seguido de paréntesis de apertura y cierre. Como toda estructura de control en Python, la definición de la función finaliza con dos puntos (:) y el algoritmo que la compone, irá identado con 4 espacios:

def mi\_funcion():

# aquí el algoritmo

1. **¿Qué es un Lenguaje de alto nivel?**
2. Es un compilador estable y potente para la solución de problemas humanos y los entienda físicamente la PC.
3. Es el sistema de códigos directamente interpretable (0 y 1) por un circuito microprogramable, como el microprocesador de una computadora.
4. Es un es un lenguaje de programación para los computadores, microprocesadores, microcontroladores y otros circuitos integrados programables.
5. Se caracteriza por expresar los algoritmos de una manera adecuada a la capacidad cognitiva humana, en lugar de la capacidad ejecutora de las máquinas

R)d d  
Un lenguaje de alto nivel permite al programador escribir las instrucciones de un programa utilizando palabras o expresiones sintácticas muy similares al inglés. Por ejemplo, en C se pueden usar palabras tales como: case, if, for, while, etc.

1. **¿Qué diferencia hay entre una clase y un objeto?**
2. Una clase es una instancia de un objeto
3. Un objeto es una instancia de una clase
4. Un objeto no tiene tipo
5. Ninguna

R)b  
Un objeto se conoce como la instancia de una clase. ¿Esto qué significa? Que hasta el momento que creemos un objeto es que podremos interactuar con lo que se definió en la clase.  
**¿Cuál es la forma correcta de escribir un bucle while?**

1. while a foreach[0..4]
2. while (a < 5)
3. while a in range(0..4)
4. while a < 5:

R)d  
Un bucle while permite repetir la ejecución de un grupo de instrucciones mientras se cumpla una condición (es decir, mientras la condición tenga el valor True) .

La sintaxis del bucle while es la siguiente:

while condicion:

cuerpo del bucle

1. **Para mostrar texto en la consola usamos el comando**
2. print
3. p
4. console.log
5. echo

R) a

1. **¿Qué símbolo se utiliza para comentar una línea de código?**
2. \*
3. --
4. //
5. #

R) d  
La primera será tener una línea con el comentario. Para ello lo que hacemos es anteponer el símbolo de la almohadilla al principio de la línea.

# Esto es un comentario en Python de una linea

1. **Para mostrar el valor de la posición 2 de un array llamada micoleccion utilizamos**
2. puts(micoleccion[2])
3. print(micoleccion[2])
4. pp(micoleccion[2])
5. print(micoleccion[1])

R) d  
Para acceder a los distintos elementos de la lista se utilizará la misma notación de índices de cadenas y tuplas, con valores que van de 0 a la longitud de la lista −1.

1. **¿Cuál de los siguientes no es un operador de asignación válido?**
2. |=1
3. +=1
4. -=1
5. @=1

R) d  
Los operadores de asignación se utilizan para basicamente asignar un valor a una variable, así como cuando utilizamos el “=”. Los operadores de asignación son “=,+=,-=,\*=,/=,\*\*=,//=”,

1. **¿Cuáles son los tipos de lenguaje de programación?**
2. Lenguaje Estructural, Máquina, Ensambladores, Fisico-Humano, lenguajes de bajo nivel y de lenguajes de alto nivel
3. Únicamente de bajo nivel y lenguajes de alto nivel.
4. El lenguaje máquina, Lenguajes ensambladores y de lenguajes de alto nivel
5. C, C++, Java, Visual Basic, SQL, SQL Server.

R) d  
Los tipos de lenguajes de programación son los siguientes:   
Lenguaje de Programación "HTML":   
Lenguaje de Programación "SQL":   
Lenguaje de Programación "Java":   
Lenguaje de Programación "C":   
Lenguaje de Programación "XML":   
Lenguaje de Programación "PHP":

1. **Son los tipos de datos que se manejan en Programación.**
2. Tipo Java, C++, Smalltalk, Python, Object Pascal, Visual .net, Visual Basic, Delphi, Perl, entre otros.
3. Cadena, Boleano, Carácter, Numericos, alfanuemrico, Entero.
4. String, Boolean, Char, Integer, int, etc.
5. Simbólicos, de estructura, de cadena, de complemento, generales, particulares, entre otros.

R) b  
El propósito general de toda computadora es manejar datos. Un dato es una colección de resultados compuesta por caracteres, números y símbolos almacenados en el procesador de manera que puedan ser procesados.

Los tipos de datos simples son: Numéricos (enteros, reales), Lógicos (booleanos) Carácter, (caracteres, cadenas de caracteres)

1. **Son los tipos de variables que se manejan en Programación.**
2. Tipo Java, C++, Smalltalk, Python, Object Pascal, Visual .net, Visual Basic, Delphi, Perl, entre otros.
3. Cadena, Boleano, Carácter, Numeros, Entero, entre otros.
4. De Actionscript, COBOL, C#, .NET, PHP, Simula, PowerBuilder, Maya, entre otros.
5. String, Boolean, Char, Integer, int, Double, Float, etc

R) d  
Una variable es un espacio de la memoria del ordenador a la que asignamos un contenido que puede ser un valor numérico (sólo números, con su valor de cálculo) o alfanumérico (sólo texto o texto con números)