



IIC1103 – Introducción a la Programación
1 - 2016

Interrogación 1

Instrucciones generales:

- La interrogación consta de cuatro preguntas con distinta ponderación.
- Solo se reciben consultas sobre el enunciado en los primeros 30 minutos.
- La última página es un recordatorio; no está permitido utilizar material de apoyo adicional.
- Las cuatro preguntas están pensadas para ser resueltas en forma sencilla con ciclos y condicionales. No es necesario utilizar otros mecanismos más avanzados.
- Si el enunciado de la pregunta no lo indica explícitamente, no es necesario validar los ingresos del usuario. Puedes asumir que no hay errores por ingreso de datos.
- Responde cada pregunta en una hoja separada.

Pregunta 1 (15 %)

A continuación se presentan tres programas. Señala qué es lo que se muestra en la consola al ejecutar cada uno de los programas.

a) Programa 1 (2 ptos)

```
a = 3
b = 4
a = b
b = 6
b = a
c = a
a = b
b = c
c = a // b
b = a**c
a = c**b
print(a + b)
```

b) Programa 2 (2 ptos)

```
import random
x = random.randint(0,4) * 18
y = x % 5
z = (4-y) + 1
w = (y + z) % 5
print(w)
```

c) Programa 3 (2 ptos)

```
import random
a = 2
b = random.randint(3,4)
c = b-a
a = 3 // b
print(c**a)
```

Pregunta 2 (25 %)

En el curso de Introducción a la Programación del 2015-2, se utilizó el siguiente criterio **general** de evaluación para la Nota Final:

$$NF = 0,30 * I + 0,30 * T + 0,30 * E + 0,10 * P$$

Donde I es el promedio de las Interrogaciones, E es la nota del Examen, T es el promedio de las Tareas y P es la nota de participación.

La nota de Participación fue considerada en forma distinta para cada sección:

- Sección 1: No se considera nota de participación. Con esto, la nota final se calcula como $NF = (0,3 * I + 0,3 * T + 0,3 * E) / 0,9$
- Sección 2: Independiente de su valor, la nota de participación se aplica tal como lo indica el criterio general de evaluación. Es decir, la nota de participación (que vale 10 %) podría subir o bajar el promedio obtenido con las Interrogaciones, Tareas y Examen.
- Sección 3: La nota de participación solo se considera si es que no baja el promedio logrado con las Interrogaciones, Tareas y Examen. Si baja el promedio, se aplica la regla de la Sección 1. Si sube (o iguala) dicho promedio se aplica la regla de la Sección 2, es decir, la nota de participación vale un 10 % de la nota final (tal como lo indica el criterio general de evaluación).
- Sección 4: Igual que sección 1.
- Sección 5: Igual que sección 2.
- Sección 6: Igual que sección 3.

Escribe un programa Python que calcula una nota final en el curso cuando recibe de entrada un número de estudiante, un número de sección, un promedio de Interrogaciones, un promedio de Tareas, una nota de Examen y una nota de participación si corresponde según la sección del alumno. Se debe aplicar el criterio de evaluación antes descrito.

El programa debe imprimir el número del alumno, la Sección y la nota final obtenida.

A continuación se presenta un ejemplo de diálogo:

```
>> Ingrese numero de alumno: 123456
>> Ingrese numero de seccion: 2
>> Ingrese promedio de Interrogaciones: 6.0
>> Ingrese promedio de Tareas: 5.0
>> Ingrese nota de Examen: 6.0
>> Ingrese nota de Participacion: 7.0
>> Nro alumno: 123456 , seccion: 2 , nota final: 58.0
```

Pregunta 3 (30 %)

a) Número curioso (1 pto)

Un número curioso es un número (entero, positivo) que cumple que su último dígito es el mismo que el último dígito de su cuadrado. Por ejemplo:

- 5 es un número curioso, pues $5^2 = 25$, que termina con 5.

- 4 no es un número curioso, pues $4^2 = 16$, que no termina con 4.
- 11 es un número curioso, pues $11^2 = 121$, y ambos terminan con 1.

Escribe un programa que le pregunte una vez al usuario por un número, e imprima si el número es o no curioso. Si el número ingresado es negativo, debe indicar que el número no es válido. A continuación se presenta un ejemplo de diálogo:

```
>> Ingrese numero: 5
>> 5 es numero curioso
```

b) Número curiosísimo (5 ptos)

Un número curiosísimo es un número (entero, positivo) que cumple que el mismo número se encuentra al final de su cuadrado. Por ejemplo:

- 1 es un número curiosísimo, pues $1^2 = 1$, que contiene a 1 como última cifra.
- 25 es un número curiosísimo, pues $25^2 = 625$, que contiene a 25 como última cifra.
- 90625 es un número curiosísimo, pues $90625^2 = 8212890625$, que contiene a 90625 como última cifra.
- 123 no es curiosísimo, pues $123^2 = 15129$

Escribe un programa que imprima los primeros 15 números curiosísimos.

Pregunta 4 (30 %)

Has sido seleccionado como voluntario para los Juegos Olímpicos 2016, y te han asignado como ayudante de los jueces en la prueba de gimnasia. Se te pide que lleves el registro de los puntajes en el “All Around” masculino, y que al final de la ronda entregues los nombres de los dos competidores con puntajes más altos, el competidor con puntaje más bajo y la cantidad de gimnastas que participaron.

Escribe un programa en Python que permita ingresar, consecutivamente, el nombre y puntaje de cada participante hasta que el usuario ingrese “FIN” en el nombre del competidor. Cuando termine el ingreso de participantes, tu programa deberá mostrar los resultados solicitados.

Un ejemplo de diálogo es:

```
>> Nombre de competidor: Tomas Gonzalez
>> Ingrese puntaje: 9.8
>> Nombre de competidor: Louis Smith
>> Ingrese puntaje: 8.3
>> Nombre de competidor: Zou Kai
>> Ingrese puntaje: 7.7
>> Nombre de competidor: Diego Hypolito
>> Ingrese puntaje: 8.5
>> Nombre de competidor: FIN

>> Puntaje maximo: 9.8, Tomas Gonzalez
>> Segundo puntaje maximo: 8.5, Diego Hypolito
>> Puntaje minimo: 7.7, Zou Kai
>> Numero de competidores: 4
```

Nota: asume que siempre se ingresarán los datos de al menos un competidor, y que un mismo competidor no será ingresado dos veces. Solo se ingresarán los datos de los competidores masculinos.



Recordatorio contenidos I1 - Primer semestre 2016

1. Tipos de datos y operadores

Tipo de dato	Clase	Ejemplo
Números enteros	int	2
Números reales	float	2.5
Números complejos	complex	2 + 3j
Valores booleanos	bool	True/False
Cadenas de texto	str	"hola"

Operación	Descripción	Ejemplo
+	Suma	2.3+5.4
-	Resta	45.45-10.02
-	Negación	-5.4
*	Multiplicación	(2.3+4.2j)*3
**	Potenciación	2**8
/	División	100/99
//	División entera	100//99
%	Módulo	10%3

Prioridad (de mayor a menor): (); **, *, /, // o %;
+ o -.

Operación	Descripción	Ejemplo
== (!=)	Igual (distinto) a	2==2
< (<=)	Menor (o igual)	1<1.1
> (>=)	Mayor (o igual)	3>=1
and	Ambos True	2>1 and 2<3
or	Algún True	2!=2 or 2==2
not	Negación	not True

Prioridad (de mayor a menor): (); or; and; not;
comparadores.

2. Funciones int, float, str

- `int(arg)` convierte `arg` a entero.
- `float(arg)` convierte `arg` a número real.
- `str(arg)` convierte `arg` a cadena de texto.

3. Función print

- Un argumento:
`print(arg)`
- Dos o más argumentos:
`print(arg1, arg2, arg3)`
- Eliminar salto de línea:
`print(arg, end='')`

4. Función input

- `ret = input(texto)` guarda en `ret` un `str` ingresado.
- `ret = int(input(texto))` guarda en `ret` un `int` ingresado.
- `ret = float(input(texto))` guarda en `ret` un `float` ingresado.

5. if/elif/else

```
if <cond 1> :  
    <codigo si se cumple cond 1>  
    if <cond 1.1> :  
        <codigo si se cumple 1.1>  
    else :  
        <codigo si no se cumple 1.1>  
elif <cond 2> :  
    <codigo si se cumple cond 2 pero no cond 1>  
else :  
    <codigo si no se cumple cond 1 ni 2>
```

6. while

```
while <condicion> :  
    <codigo que se ejecuta mientras  
    se cumple condicion>
```

7. `random.randint(i, j)`

Requiere importar módulo: `import random`.

Retorna un número entero entre `i` y `j`, ambos inclusive.

9. `math.pi`

Requiere importar módulo: `import math`.

Retorna el número $\pi = 3,141592\dots$

8. `math.sqrt(x)`

Requiere importar módulo: `import math`.

Retorna la raíz cuadrada de `x`.

10. `math.exp(x)`

Requiere importar módulo: `import math`.

Retorna e^x .