

PRÁCTICA 3: Estructuras de Control

(Segunda práctica evaluable)

El objetivo de esta práctica es que el alumno se familiarice con las estructuras de control. El alumno utilizará las transparencias de teoría, así como la ayuda integrada en el entorno como guía para resolver la relación de ejercicios propuesta. Los temas cubiertos en este guión son:

- Estructuras Condicionales: 3h. aprox.
- Estructuras Repetitivas: 3h. aprox.

Todos los ejercicios deben realizarse con el editor de Spyder, en el correspondiente fichero-.py con nombre P3_<num_ejercicio>.py.

Instrucciones para la realización de la práctica:

- La práctica durará 6h (aproximadamente), es decir, se trabajará en clase durante las sesiones de las semanas del 17/19 y 24/25 de Octubre.
- Debe entregar a través de PRADO todos los ejercicios que termine correctamente.
- **Fecha límite de entrega de los ejercicios en PRADO: Domingo 29 de Octubre a las 23:59 hrs.**
- Todos los ejercicios deben realizarse con el **editor de Spyder**, en el correspondiente fichero-py con nombre **P3_<num_ejercicio>.py** (Ejemplo: P3_1.py P3_2.py P3_3.py para los ejercicios 1, 2 y 3 respectivamente y así en adelante).
- **Es imprescindible que Ud. haga y entienda bien cada ejercicio antes de entregarlo.** Debe estar preparado/a para que, después de la entrega, **el profesor le pida que le explique algún detalle del ejercicio entregado.**
- **En las clases de prácticas del 31-O y 2-N se realizará una actividad de evaluación continua individual relacionada con estos ejercicios.**
- La nota de la práctica se determinará en función del número de ejercicios puntuables entregados correctamente y la puntuación obtenida en la actividad de evaluación continua individual. La puntuación total de la práctica (sobre 10 puntos) se agregará con la obtenida en las demás prácticas puntuables.
- Si lo desea, también puede entregar los ejercicios adicionales no puntuables, que serán tenidos en cuenta como nota extra de participación en clase.

EJERCICIOS PUNTUABLES PARA LA PRÁCTICA

Bloque 1: Ejercicios Estructuras Condicionales

1. Realice un programa (script) que pida al usuario tres números y los muestre por pantalla de manera ordenada.
2. Implemente un programa que pida al usuario tres letras (indicando para cada lectura que los posibles valores son A, U, G o C), y que identifique (muestre por pantalla) qué aminoácido se correspondería con dicha secuencia. Para ello tenga en cuenta la siguiente tabla:

Adenina

Timina

Guanina

Citosina

20 aminoácidos

Primera base (final 5')	Segunda base								Tercera base (final 3')	
	U		C		A		G			
U	UUU	fenil- alanina	UCU	serina	UAU	tirosina	UGU	cisteína	U	
	UUC	alanina	UCC		UAC		UGC		C	
	UUA	leucina	UCA		UAA	"paro"	UGA	"paro"	A	
	UUG		UCG		UAG	UGG	triptofano	G		
C	CUU	leucina	CCU	prolina	CAU	histidina	CGU		U	
	CUC		CCC		CAC		CGC		C	
	CUA		CCA		CAA	glutamina	CGA	arginina	A	
	CUG		CCG		CAG	CGG		G		
A	AUU	iso- leucina	ACU	treonina	AAU	asparagina	AGU	serina	U	
	AUC		ACC		AAC		AGC		C	
	AUA	metionina	ACA	lisina	AAA		AGA		A	
	AUG		ACG		AAG	arginina	AGG	arginina	G	
G	GUU	valina	GCU	alanina	GAU	ácido aspártico	GGU	glicina	U	
	GUC		GCC		GAC		GGC			C
	GUA		GCA		GAA	ácido glutámico	GGA			A
	GUG		GAG		GAG	GGG			G	

Ayuda: Como aún no hemos visto el tipo `string`, se pueden utilizar tres variables, una para cada una de las bases, de manera que puedan ser utilizadas en las estructuras tipo `if` para mostrar con `print('\naminoacido\n\n')` el texto correspondiente al aminoácido.

3. Escribir un menú con tres opciones para ejecutar la opción que se desee. La forma del menú será la siguiente (no hay que mostrar las líneas):

Menú:

1. Cálculo de la distancia euclídea entre dos puntos (x1, y1) y (x2, y2)
2. Cálculo de distancia d desde un punto (x0, y0) a la línea ax+by+c = 0
3. Área del triángulo definido por los puntos (x1, y1), (x2, y2) y (x3, y3)

Su opción:

Informática Aplicada a la Bioquímica – Grado en Bioquímica

Para todos los puntos se puede utilizar el código de los ejercicios anteriormente resueltos, aunque debe adaptarse en caso de ser necesario y debe estar bien tabulado.

Bloque 2: Estructuras Repetitivas

4. Incluya una quinta opción en el menú anterior con el siguiente texto '4. Salir'. El programa debe permitir al usuario estar realizando cálculos repetitivamente mientras quiera, y sólo terminará cuando el usuario indique la opción de salida. Utilizar una estructura repetitiva para tal tarea. Este ejercicio debe entregarse aparte del ejercicio 3, en un script distinto.
5. Calcular la función “Suma de 0 a n” ($\sum_{i=0}^n i$) para un número entero positivo leído desde teclado, implementando las instrucciones necesarias para el cálculo. Por otro lado, calcular la función “Producto de 1 a n” ($\prod_{i=1}^n i$) para un número entero positivo leído desde teclado, implementando las instrucciones necesarias para el cálculo. Construir un programa que de la opción de elegir entre las siguiente tres opciones:

-
- a. Ir al menú de cálculos geométricos
 - b. Ir al menú de cálculos de funciones matemáticas y series
 - c. Salir
-

Su opción:

Siendo la opción 'a' el acceso al menú del ejercicio 4, y mostrando para la opción 'b' otro menú (similar al de la opción a) que incluya la posibilidad del cálculo de la función Sumatoria, del cálculo de la función Producto y la opción de salida (o más bien de regreso, en este caso... opción - 3. Volver). Este ejercicio también debe incluirse en un script aparte del de los ejercicios anteriores.

6. Implemente un programa que calcule la evolución de dos poblaciones dados un **tamaño inicial** de población, el factor de crecimiento r_i y la capacidad del medio K_i para cada una de ellas, el factor α indicando el efecto que tiene la especie 2 sobre la especie 1, β indicando el efecto que tiene la especie 1 sobre la especie 2, y dados un **año inicial** y un **año final**, todo ello leído desde teclado. Utilice las siguientes funciones:

$$P_1(t + 1) = P_1(t) + r_1 \times P_1(t) \times (K_1 - P_1(t) - \alpha \times P_2(t))/K_1,$$
$$P_2(t + 1) = P_2(t) + r_2 \times P_2(t) \times (K_2 - P_2(t) - \beta \times P_1(t))/K_2,$$

que representan el cálculo de los tamaños de cada población tras un año, y que permitiría ir recorriendo los años a modo de bucle hasta llegar a los cálculos del año final. Finalmente, incluya en el menú 'b' del ejercicio anterior la siguiente opción 'Evolución de poblaciones en un periodo de tiempo', que realizará los cálculos anteriormente indicados. **Por lo tanto, este ejercicio también incluirá todas las opciones implementadas para los ejercicios anteriores, pero se implementará en un script distinto al de los ejercicios anteriores.**

Interpretación de los valores α y β

Valor del parámetro	Interpretación biológica (S1 = Especie 1 y S2 = Especie 2)
$\alpha > 1$	S2 inhibe a S1 más que lo que S1 se inhibe a si mismo
$\alpha = 1$	S2 inhibe a S1 lo mismo que S1 se inhibe a si mismo
$0 < \alpha < 1$	S2 inhibe a S1, pero menos que lo que S1 se inhibe a si mismo
$\alpha = 0$	S2 no afecta a S1
$-1 < \alpha < 0$	S2 potencia a S1 pero menos que lo que S1 se inhibe a si mismo
$\alpha = -1$	S2 potencia a S1 lo mismo que S1 se inhibe a si mismo
$\alpha < -1$	S2 potencia a S1 más que lo que S1 se inhibe a si mismo

La interpretación biológica del valor del parámetro β se puede obtener substituyendo S1 por S2 y S2 por S1 en la tabla anterior.

EJERCICIOS NO PUNTUABLES PARA LA PRÁCTICA (refuerzo mediante trabajo autónomo o mediante resolución por el profesor en pizarra)

1. Hacer un script que pida al usuario el límite inferior y superior de un intervalo de números enteros [inf, sup] y que imprima por pantalla todos los números pares comprendidos entre ellos utilizando sentencias repetitivas.
2. Implemente un script que le permita al usuario mostrar la tabla de multiplicar de un número que elija. Cuando el número elegido sea cero, el script finalizará. Ampliar para que muestre las tablas de multiplicar desde el 1 hasta el 10 con una pausa entre cada tabla.
3. Hacer un script que pida al usuario un valor entero N y que muestre por pantalla todos los números primos que sean menores o iguales que N.
Consejo: Hacer primero un script para demostrar si un número es primo y después hacer el ejercicio.
4. Implemente un script que permita, dado un valor entero n, calcular la sumatoria $1+2+3+....+n$ utilizando sentencias repetitivas.
5. Realizar un script que dé al usuario la posibilidad de realizar una de las siguientes operaciones: raíz cuadrada, logaritmo neperiano y cuadrado sobre un número introducido por teclado. Se mostrará un menú al usuario con las opciones (r, l, c), y se controlará que la operación sea válida para el número introducido (en caso de ser errónea, se mostrará un mensaje de error al finalizar la ejecución y no se operará). Posteriormente, incorpore una estructura repetitiva que finalizará cuando el usuario elija la opción 's' (salir).
6. Hacer un script que pida al usuario números. Si el número introducido es impar, lo imprimirá; en caso contrario los irá teniendo en cuenta para, al final, imprimir su media. Si el usuario introduce

el 0, el script finalizará con el mensaje “Fin (0) = X”, donde X será la media de los números pares introducidos. Los números reales introducidos por teclado también serán válidos, pero se redondearán al entero más cercano.

7. Implemente un script que, dado un valor n , muestre n líneas, de forma tal que cada línea contenga 10 asteriscos (*).
8. La función `random.random()` devuelve un número aleatorio con distribución uniforme en el rango [0.0, 1.0). Teniendo en cuenta la posibilidad de ciertas catástrofes meteorológicas que afectan negativamente a la capacidad del medio y que en media se producen cada 4 años, la siguiente función serviría para calcular el crecimiento de una población de individuos en dicho medio tras un año:

$$P(t + 1) = P(t) + 0.35 \times P(t) \times \frac{(15 - P(t))}{15}, \quad \text{si } \text{random}() < 0.25$$

$$P(t + 1) = P(t) + 0.35 \times P(t) \times (200 - P(t))/200, \quad \text{en otro caso}$$

Escribir un programa que lea un tamaño de población para un instante t y calcule la población para los momentos $t+1$ hasta $t+5$ indicando si en algún año se produjo una catástrofe.