

PRÁCTICA 3

Esta práctica tendrá dos bloques. Cada semana se presentará los contenidos a desarrollar en ella.

Entrega viernes 7 de mayo antes de las 15:00 horas en la zona habilitada en el Moodle de la asignatura.

La práctica se realizará en parejas y será un único entregable (zip) por pareja, nombrado como Practica3_NombreApellido1Apellido2_NombreApellido1Apellido2.zip, por ejemplo: [Practica3_PedroJimenezOrtega_AlejandraSuarezPerez.zip](#). En el zip se incluirán todos los programas implementados.

SEMANA 1- FICHEROS DE TEXTO

1. Crea un programa que a partir de un fichero de texto con nombres cree un nuevo fichero donde se guarden los nombres que contenía el anterior pero sin repeticiones. Como ejemplo de fichero de trabajo se adjunta el fichero names.txt.

El programa tiene que contemplar:

- Solicitar al usuario el nombre del fichero con nombres y el nombre del fichero donde se incluirán los nombres sin repeticiones.
- Apertura de los ficheros en el modo que le corresponde.
- Lectura desde un fichero y escritura en el otro.
- Tratamiento de cómo no incluir los nombres repetidos en el fichero de escritura.

2. Crea un programa que a partir del fichero students.csv cree un nuevo fichero donde se guarden los registros de aquellos estudiantes que tienen el correo electrónico. La cabecera del fichero students.csv (primera línea) debe mantenerse en el fichero final.

El programa tiene que contemplar:

- Solicitar al usuario el nombre del fichero a abrir en modo lectura y el nombre del fichero donde se incluirán los registros. Comprueba que se pueden abrir los ficheros, utilizando la corrección de errores con try: except:
- Lectura desde un fichero y escritura en el otro.
- Tratamiento de cómo incluir los registros correctos en el nuevo fichero.

3. El fichero heart_attack.csv contiene variables clínicas de pacientes que llegaron a un hospital con síntomas de un infarto de corazón. Las variables son: edad en años, sexo (1=male, 0=female), tipo de dolor en el pecho, presión sanguínea medida en mmHg en el momento de admisión en el hospital, colesterol en sangre medido en mg/dl, azúcar en sangre >120mg/dl (1=True, 0=False), resultado del electrocardiograma, máximo número de pulsaciones, ¿ejercicio indujo a la angina? (1=yes, 0=no), oldpeak, target (1=sufrió infarto, 0= no sufrió infarto).

Implementa una función que calcule la media de una lista y utilízala para dar los siguientes resultados por pantalla:

- Media de edad global de los que sufren infarto y los que no.
- Media de edad desglosado por hombres y mujeres.
- Haz lo mismo que en los dos anteriores pero con la variable colesterol.

SEMANA 2- DICCIONARIOS

4. Una secuencia de ADN está formado por una cadena compuesta por las cuatro letras del alfabeto {A, T, G, C}, por ejemplo ATGCATTAC. Cada elemento de esta cadena es un nucleótido.

Cada tres letras forman un codón (tripleto de nucleótidos). En el código genético cada aminoácido está codificado por uno o varios codones. Las posibilidades de formar distintos codones a partir de las 4 nucleótidos es $64=4^3$ y codifican 20 aminoácidos y 3 señales de parada útiles para la traducción del código genético, por lo que hay redundancia un mismo aminoácido está representado por varios codones. Una proteína es una secuencia de aminoácidos.

Tabla 1 Representación Codones-Aminoácidos

		Second Letter						
		T	C	A	G			
First Letter	T	TTT } Phe TTC } TTA } Leu TTG }	TCT } TCC } Ser TCA } TCG }	TAT } Tyr TAC } TAA } Stop TAG } Stop	TGT } Cys TGC } TGA } Stop TGG } Trp	Third Letter	T	C A G
	C	CTT } CTC } Leu CTA } CTG }	CCT } CCC } Pro CCA } CCG }	CAT } His CAC } CAA } Gln CAG }	CGT } CGC } Arg CGA } CGG }			
	A	ATT } ATC } Ile ATA } ATG } Met	ACT } ACC } Thr ACA } ACG }	AAT } Asn AAC } AAA } Lys AAG }	AGT } Ser AGC } AGA } Arg AGG }			
	G	GTT } GTC } Val GTA } GTG }	GCT } GCC } Ala GCA } GCG }	GAT } Asp GAC } GAA } Glu GAG }	GGT } GGC } Gly GGA } GGG }			

En esta práctica vamos a construir un traductor de ADN (secuencia de nucleótidos) a proteínas (secuencia de aminoácidos).

Para ello contamos con el módulo incompleto `genoma.py` que tenemos que completar, en él están definidas las funciones a desarrollar y la estructura de datos tipo diccionario que se corresponde con la Tabla 1 y un diccionario adicional que relaciona la notación corta (1 letra) con el aminoácido correspondiente.

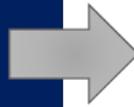
Se proporciona el fichero del genoma del SARS coronavirus en formato fasta. En este formato una secuencia comienza con una descripción en una única línea (línea de cabecera), seguida por líneas de datos de secuencia.

Se aporta el programa principal (incompleto, faltan funciones por implementar) que hace uso del módulo anterior y un fichero con secuencia de nucleótidos en formato fasta para crear un nuevo fichero con la secuencia de aminoácidos correspondiente a la secuencia de ADN leída y como información adicional muestra la frecuencia de los aminoácidos que componen la secuencia estudiada.

IMPORTANTE: Recuerda lo que **no debes hacer** para completar el código del módulo. Cada función, como unidad de código, debe ser correcta y cumplir con todas las especificaciones dadas. No pienses en grande, ve paso a paso y comprueba/testea cada función con simulaciones desde la consola.

DON'T

- Write entire program
- Test entire program
- Debug entire program



DO

- Write a function
- Test the function, debug the function
- Write a function
- Test the function, debug the function
- *** Do integration testing ***