**Desafío III: BioCorp Ltda.**

**Monserrat Torres.**

**Taller de Ingeniería II.**

**Gabriel Núñez.**

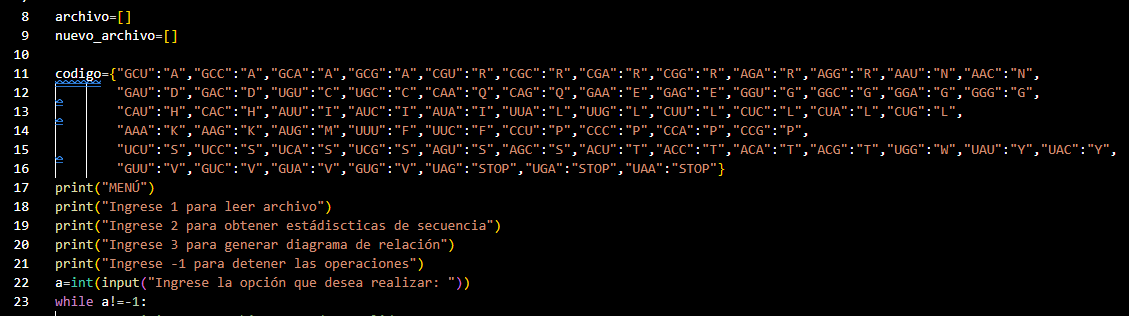
**19.12.2022.**

**Resumen del desafío:**

Se nos solicita realizar un programa que lea un archivo en formato Fasta, el cual contiene una secuencia de nucleótidos de un gen la cual debemos transformar a una secuencia de proteínas. La secuencia inicial está conformada por Adenina (A), Guanina (G), Citocina (C), y Timina (T), además de contener intrones, los cuales son partes de la secuencia que no codifican nada, estas están delimitadas por el inicio (I) y el final (F). Luego de eliminar los intrones, la Timina es transformada en Uracilo (U). Luego se empieza a buscar el codón de inicio que es la Metionina (AUG), y el codón de Stop que puede ser (UAG; UGA; UAA). Todo lo que esté entre esos dos codones es la secuencia válida, el resto será descartado, pues no tiene ningún valor en la secuencia.

Para que una proteína sea considerada como tal, tiene que contar con una cantidad de nucleótidos mayor a 15, después de la Metionina de inicio. El codón de Stop no es contado.

Antes de realizar el desafió

Te pide que crees un menú con cada requerimiento, además se crea un diccionario con los aminoácidos.

**Requerimientos del desafió:**

1. Leer un archivo en formato Fasta que contiene la secuencia de nucleótidos, la cual debe ser transformada a proteína e impresa en un nuevo archivo Fasta:

Primero abrimos el archivo y leemos la primera línea de este, que solo nos entrega el nombre. Luego leemos el resto del archivo, guardando la secuencia en una lista llamada Archivo. La cual se salta la sección (intrón) desde I a F.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteLuego cambiamos la Timina por Uracilo. El “”.join(archivo) hace que las letras se guarden en un formato de texto.

Texto

Descripción generada automáticamenteDespués empezamos a leer la lista para encontrar la Metionina de inicio (AUG) y empezar a guardar los codones en una lista llamada nuevo\_archivo. Los codones van de a tres letras e ignora todas las que estén antes del Start y después del Stop.

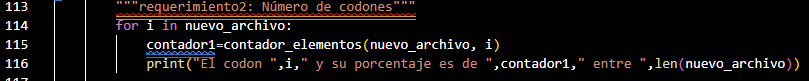
Texto

Descripción generada automáticamentePor último, empiezo a cambiar los codones por la letra de su aminoácido, y los guardo en el archivo final. Al comienzo puse un ultima.append(“M”), que corresponde a la Metionina de inicio.

1. Obtener estadísticas de la secuencia:

Texto

Descripción generada automáticamenteEmpieza a contar la Adenina, Guanina, Citocina y el Uracilo en la secuencia válida. Además de sacar su porcentaje.

Texto

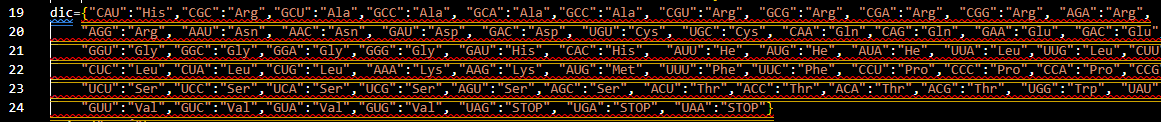
Descripción generada automáticamenteCuenta la cantidad de codones en la secuencia válida y saca su porcentaje. Para esto ocupé una función que retorna el contador del elemento ingresado previamente.

Texto

Descripción generada automáticamentePor último, hace un conteo de los tipos de aminoácido que hay en la secuencia. Polar sin carga, polar positivo, polar negativo y apolar.

1. Generar un diagrama:

Se genera un diccionario con las abreviaciones de las proteínas y se importa la librería.



La línea 151 permite que se genere el gráfico en formato png. Luego en el mismo gráfico genera un nodo donde guarda el AUG en un círculo, el cual ocupa el primer lugar de la lista nuevo\_archivo. Luego con un ciclo empieza a tomar cada codón para unirlo con el anterior, todos rodeados por un círculo. La línea 159 crea un ciclo que lee cada posición hasta el largo del archivo y los guarda en un nodo dentro de un rectángulo. Solo que, en este caso, en vez de guardar el codón, se guarda su aminoácido correspondiente al que está relacionado en el diccionario. Por último, crea un nodo circular que guarda en la última posición el Stop.

