

Del ADN a la Proteína

...

Elisa Breeze y Néstor Ortega

Del ADN a la Proteína sin Biopython

```
with open("gene.fna", "r") as f:
    lineas = f.readlines() # leer todas las líneas

adn = ''.join([l.strip().upper() for l in lineas if not l.startswith('>')]) # descartamos encabezados FASTA (empiezan con >)

print(f"Secuencia original:")
print(adn)
print("\n")

# Replicación (generación de hebras complementarias)
pares = {'A': 'T', 'T': 'A', 'C': 'G', 'G': 'C'} # Diccionario con bases complementarias
complementario = ''.join([pares[b] for b in adn]) # Hebra complementaria
molde = complementario[::-1] # Hebra molde (Invirtiendo hebra complementaria)

print("Paso 1: Replicación")
print(f"Hebra complementaria (5'→3'): {complementario}")
print(f"Hebra molde (3'→5'): {molde}")
print("\n")

# Transcripción (ADN a ARNm)
arnm = molde.replace('T', 'U') # Convertir a Secuencia ARNm
print("Paso 2: Transcripción")
print(f"ARNm: {arnm}")
print("\n")
```

Traducción (ARNm a proteína)

```
codones = {  
    'UUU': 'F', 'UUC': 'F', 'UUA': 'L', 'UUG': 'L',  
    'UCU': 'S', 'UCC': 'S', 'UCA': 'S', 'UCG': 'S',  
    'UAU': 'Y', 'UAC': 'Y', 'UAA': 'STOP', 'UAG': 'STOP',  
    'UGU': 'C', 'UGC': 'C', 'UGA': 'STOP', 'UGG': 'W',  
  
    'CUU': 'L', 'CUC': 'L', 'CUA': 'L', 'CUG': 'L',  
    'CCU': 'P', 'CCC': 'P', 'CCA': 'P', 'CCG': 'P',  
    'CAU': 'H', 'CAC': 'H', 'CAA': 'Q', 'CAG': 'Q',  
    'CGU': 'R', 'CGC': 'R', 'CGA': 'R', 'CGG': 'R',  
  
    'AUU': 'I', 'AUC': 'I', 'AUA': 'I', 'AUG': 'M',  
    'ACU': 'T', 'ACC': 'T', 'ACA': 'T', 'ACG': 'T',  
    'AAU': 'N', 'AAC': 'N', 'AAA': 'K', 'AAG': 'K',  
    'AGU': 'S', 'AGC': 'S', 'AGA': 'R', 'AGG': 'R',  
  
    'GUU': 'V', 'GUC': 'V', 'GUA': 'V', 'GUG': 'V',  
    'GCU': 'A', 'GCC': 'A', 'GCA': 'A', 'GCG': 'A',  
    'GAU': 'D', 'GAC': 'D', 'GAA': 'E', 'GAG': 'E',  
    'GGU': 'G', 'GGC': 'G', 'GGA': 'G', 'GGG': 'G'  
}
```

```
arnm = arnm.upper()  
proteina = ""  
  
for i in range(0, len(arnm), 3): # Recorrer ARNm de 3 en 3 (codones)  
    codon = arnm[i:i+3] # Extraer el codón  
    if len(codon) < 3: # Si queda menos de 1 codón entero, se para  
        break  
    aa = codones.get(codon, '') # Obtenemos aminoácido y lo añadimos a la proteína  
    if aa == 'STOP': # si se trata de un codón de parada, se para  
        break  
    proteina += aa  
  
print("Paso 3: Traducción")  
print(f"Proteína traducida: {proteina}")
```

Secuencia original:

CTCAAAAGTCTAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGACACTTTGCGTTCGGGCT

Paso 1: Replicación

Hebra complementaria (5'→3'): GAGTTTTAGATCTCGGTGGCAGGTCCCTCGTCCATCGAC

Hebra molde (3'→5'): TGGCAGCAAAGTTTTATTGTAAAATAAGAGATCGATATAAA

Paso 2: Transcripción

ARNm: UGGCAGCAAAGUUUUUAUUGUAAAAUAAGAGAUCAUAUAAAAUUGGGAUAUAAAAAGGGAGAAGG

Paso 3: Traducción

Proteína traducida: WQSFIVK

Del ADN a la proteína con Biopython

```
from Bio.Seq import Seq
from Bio import SeqIO

for secuencia in SeqIO.parse("gene.fna", "fasta"):
    dna = secuencia.seq
    print(f"Secuencia original ({secuencia.id}):")
    print(dna)
    print("\n")

    # Replicación (generación de hebras complementarias)
    complementario = dna.complement()
    molde = dna.reverse_complement()
    print("Paso 1: Replicación")
    print(f"Hebra complementaria (5'→3'): {complementario}")
    print(f"Hebra molde (3'→5'): {molde}")
    print("\n")

    # Transcripción (ADN a ARNm)
    arnm = molde.transcribe()
    print("Paso 2: Transcripción")
    print(f"ARNm: {arnm}")
    print("\n")

    # Traducción (ARNm a proteína)
    proteina = arnm.translate(to_stop=True) # Solo hasta el codón Stop
    print("Paso 3: Traducción")
    print(f"Proteína traducida: {proteina}")
```

Secuencia original (NC_000017.11:c7687490-7668421):

CTCAAAAGTCTAGAGCCACCGTCCAGGGAGCAGGTAGCTGCTGGGCTCCGGGGACACTTTGCGTTCGGGCTGGGAGCGTGCTTCCACGACGGT

Paso 1: Replicación

Hebra complementaria (5'→3'): GAGTTTTTCAGATCTCGGTGGCAGGTCCCTCGTCCATCGACGACCCGAGGCCCTGTGAAACGC

Hebra molde (3'→5'): TGGCAGCAAAGTTTTATTGTAAATAAGAGATCGATATAAAATGGGATATAAAAGGGAGAAG

Paso 2: Transcripción

ARNm: UGGCAGCAAAGUUUUAUUGUAAAAUAAGAGAUUGAUUAAAAUUGGGAUUAAAAAGGGAGAAGGAGGGGAAGGGUGGGUGAAAAUG

Paso 3: Traducción

Proteína traducida: WQSFIVK