# Project 1: Fun with DNA (REGEX Lookaround)!

In this project we find Opening reading frame or ORF from DNA sequences with the help of Python regex.

**DNA** is a sequence of bases, `A`, `C`, `G`, or `T`. They are translated into proteins 3-bases where each sequence is called a **codon**. There is a special start codon `ATG`, and three stop codons, `TGA`, `TAG`, and `TAA`. Example:

```
cgcgcATGcATGcgTGAcTAAcgTAGcgcgcgc
```

An opening reading frame or **ORF** consists of a **start codon**, followed by some more codons, and ending with a **stop codon**. The above example has overlapping ORFs.

- `ATGcATGcgTGA` and
- `ATGcgTGAcTAA`.

The following pattern only finds the first ORF (`atgcatgcgtga'`). Since it consumes the first ORF, it also consumes the beginning of the second ORF.

```
from re import *

dna = 'cgcgcATGcATGcgTGAcTAAcgTAGcgcgcgc'
dna = dna.lower()
orfpat = r'(?x) ( atg  (?: (?!tga|tag|taa) ... )*  (?:tga|tag|taa) )'
print findall(orfpat,dna)
```

We want to find an ORF without consuming it, we can use a **positive lookahead** assertion (`(?= ( atg`). We put the whole ORF pattern inside the lookahead and find the two `atgcatgcgtga` and `atgcgtgactaa`.

```
from re import *

dna = 'cgcgcATGcATGcgTGAcTAAcgTAGcgcgcgc'
dna = dna.lower()
dna = dna.lower()
orfpat = r'(?x) (?= ( atg  (?: (?!tga|tag|taa) ... )*  (?:tga|tag|taa) ))'
s = findall(orfpat,dna)

if s:
     print ', '.join(s)
```

This project **adopts** and **simplifies** the Splitsvile examples (DNA) from Rex Dwyer’s ipython notebook.