Idea

The genetic code of organisms is stored in DNA molecules as a long string of four nucleotides: A (adenine), C (cytosine), G (guanine), and T (thymine). Short strings of DNA can be “sequenced”—the sequence of letters determined—by various modern biotech methods. Although sequence for a single gene typically has hundreds or thousands of letters, there exist special enzymes that will split a long string into short fragments (which can be sequenced) by breaking the string immediately following each appearance of a particular letter.

Suppose a C-enzyme (which splits after each appearance of C) breaks a 20-letter string into eight fragments, which are identified to be: AC, AC, AAATC, C, C, C, TATA, TGGC. Note that each fragment, except the last one on the string, must end with a C.

This what we will adopt in our project. Our language is the long string of nucleotides that will be split by C-enzyme.

The grammar:

<S>::=<nucleotides-sequence>< nucleotides -sequence-without-C>

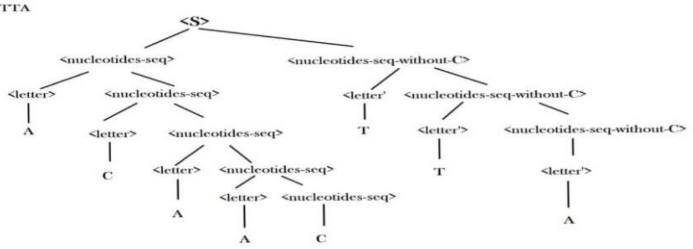
<nucleotides-sequence >::=<letter>< nucleotides-sequence >|C

< nucleotides -sequence-without-C >::=<letter`>< nucleotides -sequence-without-C >|<letter`>

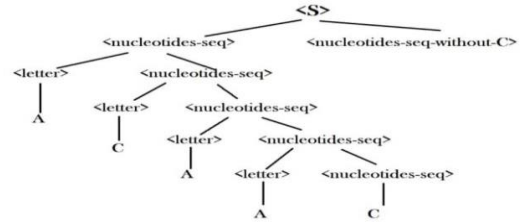
<letter>::=A|C|G|T

<letter`>::=A|G|T #The grammar should accept the string ACAACTTA, but reject ACAAC,AAA.

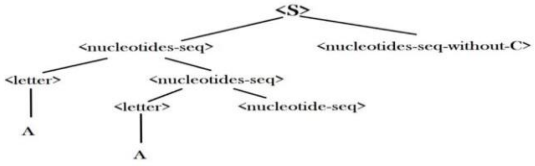
1- ACAACTTA



2- ACAAC



3- AAA



Participating student:

1-Aya Mostafa Farouk

2-Lubna Swelam Mohamed