using System.Security.Cryptography.X509Certificates;

class Genomics

{

public string findGene(string dna){

//Start codon: ATG

//Stop codon: TAA

//find ATG index

int index\_atg = dna.IndexOf("ATG");

if(index\_atg == -1){

return "No gene found";

}

//Console.WriteLine("ATG index: " + index\_atg);

//find TAA index after ATC

int index\_taa = dna.IndexOf("TAA", index\_atg);

if(index\_taa == -1){

return "No gene found";

}

Console.WriteLine("TAA index: " + index\_taa);

//report substring between ATG and TAA (codon included)

return dna[index\_atg..(index\_taa + 3)];

}

public string findValidGene(string dna){

int startIndex = dna.IndexOf("ATG");

if(startIndex == -1){

return "No gene found";

}

int stopIndex = dna.IndexOf("TAA", startIndex + 3);

while (stopIndex != -1)

{

if((stopIndex - startIndex) % 3 == 0){

//...

string gene = dna[startIndex..(stopIndex + 3)];

return gene;

}

else {

//stopIndex++;

stopIndex = dna.IndexOf("TAA", stopIndex++);

}

}

return "No gene found";

//in valid gene stopIndex - startIndex should be multiple of 3

//if stopIndex - startIndex is not multiple of 3, find the next TAA (new stopCodon)

//and repeat the process - Do this until you find a valid gene or stopIndex is -1

}

}

class Program

{

public static void Main(string[] args)

{

// start codeon: ATG

// stop codon: TAA

string dna = "ACGATGCGTAAGC";

// ^ ^

var gene = new Genomics();

string sub\_gene = gene.findGene(dna);

Console.WriteLine("The gene section: " + sub\_gene);

string dna2 = "ACGACTGCGTAAAGC";

sub\_gene = gene.findGene(dna2);

Console.WriteLine("The gene section: " + sub\_gene);

string dna3 = "ATGxxxyyyzzzTAA";

sub\_gene = gene.findValidGene(dna3);

Console.WriteLine("The gene section: " + sub\_gene);

string dna4 = "ATGxxxyyyzzzyyTAAxxxyyyzzzCTAA";

sub\_gene = gene.findValidGene(dna4);

Console.WriteLine("The gene section: " + sub\_gene);

}

}