

Interview Questions

1. If

M represents either A or C

R represents either A or G

W represents either A or T

S represents either C or G

Y represents either C or T

K represents either G or T

Write a program to pick all the sequences (string) matching to CSGGWTGGYGGCGTCRCTGTC from the below list

CCGGATGGCGGCGTCACTGTC

CCGGATGGCGGCGTCGCTGTC

CGGGTTGGCGGCGTCGCTGTC

CAGGTTGGCGGCGTCACTGTC

CTGGTTGGTGGCGTCACTGTC

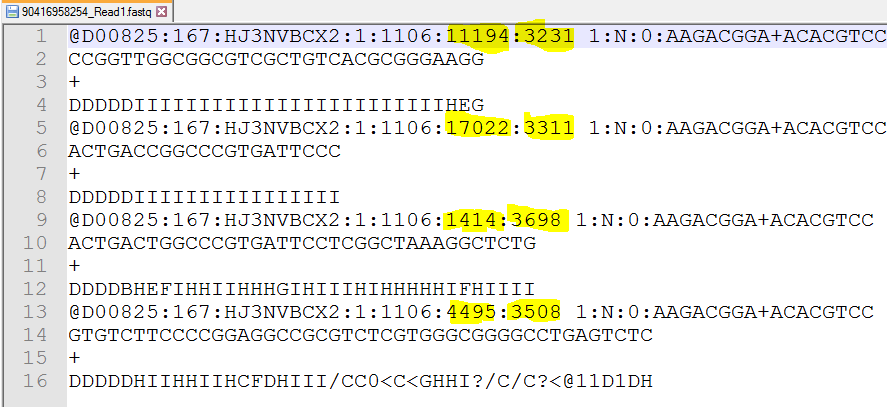
CGGGTTGGTGGCGTCGCTGTC

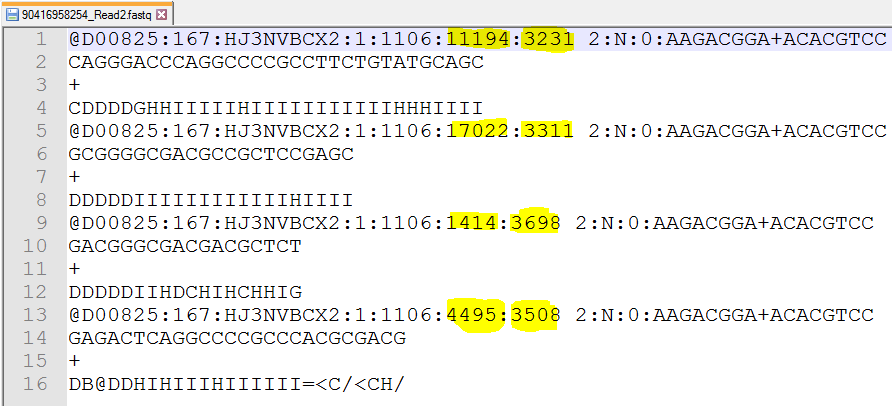
CGGGGTGGTGGCGTCGCTGTC

1. There are 2 files (available at C:\FastQLocation), names ending with Read1 and Read2

File 1 : 90416958254\_Read1.fastq

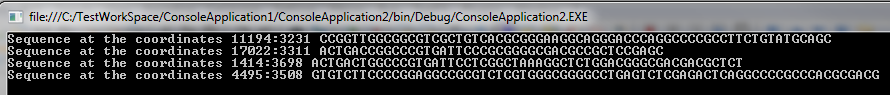
File 2 : 90416958254\_Read2.fastq





1. 4 Consecutive lines in any file represent unique DNA content. And the content coming from the same piece of DNA (from both the files) is mapped using X & Y coordinate (Highlighted).
2. A <SPACE> (ASCII Value 32) will be followed after X:Y coordinate always.
3. X:Y coordinates are available at 1, 5, 9, 13 lines in both the files
4. DNA sequences are available at 2, 6, 10, 14 lines in both the files

Write a simple program to read both the files, concatenate and print the DNA sequences coming from the same piece of DNA [from both the files] (as shown in the picture below)



1. There are 4 lines of Comma (,) delimited numbers (as shown below) and each number does not exceed 3 characters including decimal point

0,11,100,125,596,985,44,1,633,789,99,582,999,859,962

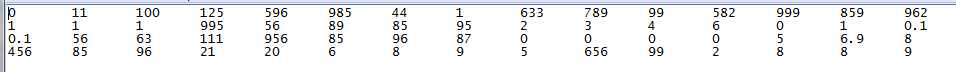
1,1,1,995,56,89,85,95,2,3,4,6,0,1,0.1

0.1,56,63,111,956,85,96,87,0,0,0,0,5,6.9,8

456,85,96,21,20,6,8,9,5,656,99,2,8,8,9

Each line contains 15 numbers to represent 15 positions.

Now align all the numbers (without commas) to easily visualize position wise values as shown below. (No need to write a program. Just write the approach how it can be done (by using any program))



Answers

1.Answer for the Question1

static void Main(string[] args)

{

List<string> lstSeq = new List<string>();

lstSeq.Add("CCGGATGGCGGCGTCACTGTC");

lstSeq.Add("CCGGATGGCGGCGTCGCTGTC");

lstSeq.Add("CGGGTTGGCGGCGTCGCTGTC");

lstSeq.Add("CAGGTTGGCGGCGTCACTGTC");

lstSeq.Add("CTGGTTGGTGGCGTCACTGTC");

lstSeq.Add("CGGGTTGGTGGCGTCGCTGTC");

lstSeq.Add("CGGGGTGGTGGCGTCGCTGTC");

string StringToMatch = "CSGGWTGGYGGCGTCRCTGTC";

StringToMatch=StringToMatch.Replace("M","[AC]");

StringToMatch=StringToMatch.Replace("R","[AG]");

StringToMatch=StringToMatch.Replace("W","[AT]");

StringToMatch=StringToMatch.Replace("S","[CG]");

StringToMatch=StringToMatch.Replace("Y","[CT]");

StringToMatch=StringToMatch.Replace("K","[GT]");

foreach(string obj in lstSeq)

{

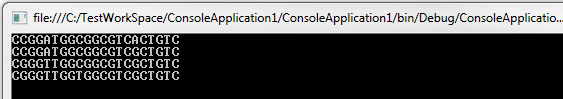
if (Regex.IsMatch(obj, StringToMatch))

Console.WriteLine(obj);

}

Console.ReadKey();

}



2.Answer for the Question2

class Program

{

static Dictionary<string, string> dictRead1Values = new Dictionary<string, string>();

static Dictionary<string, string> dictRead2Values = new Dictionary<string, string>();

static void Main(string[] args)

{

string directory = "C:\\FastQLocation";

string[] fastQFiles = Directory.GetFiles(directory);

foreach (string fastQFile in fastQFiles)

{

if (fastQFile.Contains("\_Read1"))

{

CaptureContent(fastQFile,1 );

}

else if (fastQFile.Contains("\_Read2"))

{

CaptureContent(fastQFile,2);

}

}

foreach (string key in dictRead1Values.Keys)

{

Console.WriteLine("Sequence at the coordinates "+key+" " + dictRead1Values[key] + dictRead2Values[key]);

}

Console.ReadKey();

}

static void CaptureContent(string txtFile, int readNum)

{

try

{

string line1 = string.Empty;

string line2 = string.Empty;

string yCood = string.Empty;

string xCood = string.Empty;

int i = 1;

foreach (string line in File.ReadLines(txtFile))

{

if ((i % 4) == 1)

{

line1 = line;

}

else if ((i % 4) == 2)

{

line2 = line;

}

else if ((i % 4) == 0)

{

string seqIdentifier = line1;

seqIdentifier = seqIdentifier.Substring(0, seqIdentifier.IndexOf(" "));

yCood = seqIdentifier.Substring(seqIdentifier.LastIndexOf(':') + 1);

seqIdentifier = seqIdentifier.Substring(0, seqIdentifier.LastIndexOf(':'));

xCood = seqIdentifier.Substring(seqIdentifier.LastIndexOf(':') + 1);

if (readNum == 1)

{

if (!dictRead1Values.ContainsKey(xCood + ":" + yCood))

{

dictRead1Values.Add(xCood + ":" + yCood, line2);

}

}

else if (readNum == 2)

{

if (!dictRead2Values.ContainsKey(xCood + ":" + yCood))

{

dictRead2Values.Add(xCood + ":" + yCood, line2);

}

}

}

i = i + 1;

}

}

catch (Exception ex)

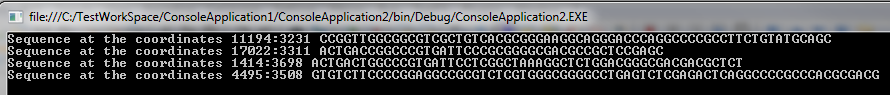
{

throw ex;

}

}

}



3. Answer for the Question 3

Replace Comma (,) with TAB character. That’s it. And the numbers will be aligned.

To achieve this in notepad, a tab character has to be explicitly copied from somewhere else.