

LAB: PROGRAMMING IN PERL
SUBJECT CODE: BINF-553



CENTRE FOR BIOINFORMATICS
PONDICHERRY UNIVERSITY

Name: BABUL PRADHAN
Reg. No.: 18378004
M.Sc 2nd Year(Bioinformatics)
Session: 2018-2020

CERTIFICATE

Certified that this is a bonafide record of the work done by
_____Reg.No. _____of
2nd year M.Sc Bioinformatics, in the Lab- Programming In Perl
during the academic year 2018-2019.

Faculty In Charge

The Head
Centre for Bioinformatics
Pondicherry University

Submitted for the M.Sc Practical Examination held on
_____ at Centre for Bioinformatics, Pondicherry
University.

EXAMINER

1. _____ 2. _____

INDEX

Sl.No.	Date	Title	Page No.	Remarks
1	10/07/2019	Reverse Complement And Count Hydrogen Bonds	1-3	
2	17/07/2019	Counting Number And Percentage Of Aliphatic Amino Acid	4-5	
3	24/07/2019	Counting Molecular Weight Of A Protein Sequence In FASTA Format	6-8	
4	31/07/2019	Convert The Single Letter Amino Acid Code To Three Letter Code	9	
5	07/08/2019	Calculate Di Nucleotide Frequency	10-11	
6	14/08/2019	Use Of ARGV	12	
7	21/08/2019	Regular Expression	13	
8	04/09/2019	Retrieve Parts From A PDB File	14-15	
9	25/09/2019	Count Polar Amino Acid Using File Handling	16-23	
10	09/10/2019	CGI Script For Biodata	24-26	
11	16/10/2019	CGI For Getting And Displaying Protein Details	27-28	
12	19/10/2019	Calculating GRAVY For A Protein	29-30	
13	23/10/2019	Converting Format Using Bioperl	31	
14	30/10/2019	Retrieving Sequence Using Bioperl	32	

Experiment No. 01

Date: 10/07/2019

Reverse Complement And Count Hydrogen Bonds

Aim:

To Learn About Getting The Reverse Complement And Total Number Of Hydrogen Bonds For A Given DNA Sequence

Algorithm:

Input:

```
#!/usr/bin/perl

print "Please Enter The DNA Sequence\n";

$DNA=<STDIN>;

chomp($DNA);

@DNA=split("", $DNA);

$rev=reverse($DNA);

@rev=split("", $rev);

$len=length $DNA;

$revcom=$rev;

@revcom=split("", $revcom);

$i=0;

$hyd=0;

for($i=0;$i<$len;$i++)

{

    if(@DNA[$i] eq "A" || @DNA[$i] eq "T")

    {

        $hyd=$hyd+2;

    }

}
```

```
else{
    if(@DNA[$i] eq "C" || @DNA[$i] eq "G")
    {
        $hyd=$hyd+3;
    }
}
}
for($i=0;$i<$len;$i++)
{
    if(@revcom[$i] eq "A" || @revcom[$i] eq "a")
        {@revcom[$i]="T";}
    else{
        if(@revcom[$i] eq "T" || @revcom[$i] eq "t")
            {@revcom[$i]="A";}
        else{
            if(@revcom[$i] eq "C" || @revcom[$i] eq "c")
                {@revcom[$i]="G";}
            else{
                if(@revcom[$i] eq "G" || @revcom[$i] eq "g")
                    {@revcom[$i]="C";}
            }
        }
    }
}
print "The Reverse Complement Of The Sequence Is\n@revcom\n";
```

print "Total Number Of Hydrogen Bond Involved In This Sequence Is
\$hyd\nAnd The Length Of The DNA Is \$len;";

Result:

```
C:\WINDOWS\system32\cmd.exe - perl 1.pl
C:\Users\18378004\Desktop\Lab Record Perl>perl 1.pl
Please Enter The DNA Sequence
ACATTTGCTTCTGCACAACTGTGTTCACTAGCAACCTCAAAACAGACACCATGGTGCACTGACTCCTGAGGAGAAGTCTGCCGTTACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAGTTGGTGGTGAAGCCCTGGGACAGTTGGTATCAAGGTTACAAGACAGGT
TTAAGGAGACCAATAGAACTGGGCATGTGGAGACAGAGAAGACTCTGGGTTTCTGATAGGCAGTACTCTCTGCTGCTATTTGCTATTTTCCACCCCTTAGGCTGCTGGTGGTCTACCCCTGGACCCAGAGGTTCTTTGAGTCTTTGGGGATCTGCTCACTCCT
GATGCTGTTATGGGCAACCCTAAGGTGAAGGCTCATGGCAAGAAAGTCTCGGTGCCTTAGTGATGGCTGGCTCACCTGGACAACCTCAAGGGCACCTTTGCCACACTGAGTGAGCTGCACTGTGACCAAGCTGCACGTGGATCCTGAGAAGCTTCAGGGTGAGTCTA
TGGGACGCTTGATGTTTTCTTCCCCTCTTTTCTATGGTTAAGTTCATGTCATAGGAAGGGGATAAGTAACAGGGTACAGTTTAAAGTGGGAAACAGACGAATGATTCATCAGTGTGGAGTCTCAGGATCGTTTTAGTTCTTTTATTTGCTGTTCAATAACAAAT
GTTTTCTTTTGTAAATCTTGCTTCTTTTTTCTTCTCGGCAATTTTACTATTATACCTTAATGCCCTTAACATTTGATGATTAACAAAAGGAAATATCTCTGAGATACATTAAGTAACCTAAAAAAGCTTACACAGCTGCGCTAGTACATTAATTTGGAAT
ATATGTGCTGCTTATTCATATTCAATCTCCCTACTTTATTTCTTTTAAATGATACATAATCAATATACATATTTATGGGTTAAAGTGAATGTTTTAAATATGTGTACACATTTGACCAAACTCAGGGTAATTTTGCATTTGTAATTTAAAAAATGCT
TTCTCTTTTAAATACTTTTTTGTATTCTATTCTAATACTTCCCTAATCTCTTTCTTTCAGGGCAATAATGATACAAATGATCATGCCTCTTTGCACCATTTCTAAAGAATAACAGTGATAATTTCTGGGTTAAGGCAATAGCAATATCTCTGCATATAAATAT
TTCTGCATATAAATGTAAGTGTAAAGGTTTCAATTTGCTAATAGCAGCTACAATCCAGCTACCATTTCTGCTTTTATTTATGGTTGGGATAAGGCTGATTATTTCTGAGTCCAAGCTAGGCCCTTTTGCATATCATGTTTCACTCTTATCTCTCCCA
GCTCCTGGGCAACGTGCTGGTCTGTGTGCTGGCCATCACTTTGGCAAGAATTCAACCCACAGTGCAGGCTGCCTATCAGAAAGTGGTGGCTGGTGGCTAATGCCCTGGCCCAAGTATCACTAAGCTCGCTTTCTGCTGCAATTTCTATTAAAGGTTTC
TTTTGTTCCCTAAGTCCAACCTACTAACTGGGGATATTATGAAGGGCTTGAGCATCTGGATTCTGCTAATAAAAAACATTTATTTTCATTGCAA
```

(Input)

```
C:\WINDOWS\system32\cmd.exe
C:\Users\18378004\Desktop\Lab Record Perl>perl 1.pl
Please Enter The DNA Sequence
ACATTTGCTTCTGCACAACTGTGTTCACTAGCAACCTCAAAACAGACACCATGGTGCACTGACTCCTGAGGAGAAGTCTGCCGTTACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAGTTGGTGGTGAAGCCCTGGGACAGTTGGTATCAAGGTTACAAGACAG
TTAAGGAGACCAATAGAACTGGGCATGTGGAGACAGAGAAGACTCTGGGTTTCTGATAGGCAGTACTCTCTGCTGCTATTTGCTATTTTCCACCCCTTAGGCTGCTGGTGGTCTACCCCTGGACCCAGAGGTTCTTTGAGTCTTTGGGGATCTGCTCACTCCT
GATGCTGTTATGGGCAACCCTAAGGTGAAGGCTCATGGCAAGAAAGTCTCGGTGCCTTAGTGATGGCTGGCTCACCTGGACAACCTCAAGGGCACCTTTGCCACACTGAGTGAGCTGCACTGTGACAAGCTGCACGTGGATCCTGAGAAGCTTCAGGGTGAGTC
TGGGACGCTTGATGTTTTCTTCCCCTCTTTTCTATGGTTAAGTTCATGTCATAGGAAGGGGATAAGTAACAGGGTACAGTTTAAAGTGGGAAACAGACGAATGATTCATCAGTGTGGAGTCTCAGGATCGTTTTAGTTCTTTTATTTGCTGTTCAATAACAA
GTTTTCTTTTGTAAATCTTGCTTCTTTTTTCTTCTCGGCAATTTTACTATTATACCTTAATGCCCTTAACATTTGATGATTAACAAAAGGAAATATCTCTGAGATACATTAAGTAACCTAAAAAAGCTTACACAGCTGCGCTAGTACATTAATTTGGA
ATATGTGCTGCTTATTCATATTCAATCTCCCTACTTTATTTCTTTTAAATGATACATAATCAATATACATATTTATGGGTTAAAGTGAATGTTTTAAATATGTGTACACATTTGACCAAACTCAGGGTAATTTTGCATTTGTAATTTAAAAAATG
TTCTCTTTTAAATACTTTTTTGTATTCTATTCTAATACTTCCCTAATCTCTTTCTTTCAGGGCAATAATGATACAAATGATCATGCCTCTTTGCACCATTTCTAAAGAATAACAGTGATAATTTCTGGGTTAAGGCAATAGCAATATCTCTGCATATAAAT
TTCTGCATATAAATGTAAGTGTAAAGGTTTCAATTTGCTAATAGCAGCTACAATCCAGCTACCATTTCTGCTTTTATTTATGGTTGGGATAAGGCTGATTATTTCTGAGTCCAAGCTAGGCCCTTTTGCATATCATGTTTCACTCTTATCTCTCCCA
GCTCCTGGGCAACGTGCTGGTCTGTGTGCTGGCCATCACTTTGGCAAGAATTCAACCCACAGTGCAGGCTGCCTATCAGAAAGTGGTGGCTGGTGGCTAATGCCCTGGCCCAAGTATCACTAAGCTCGCTTTCTGCTGCAATTTCTATTAAAGGTTTC
TTTTGTTCCCTAAGTCCAACCTACTAACTGGGGATATTATGAAGGGCTTGAGCATCTGGATTCTGCTAATAAAAAACATTTATTTTCATTGCAA
The Reverse Complement Of The Sequence Is
T T G C A A T G A A A A T A A A T G T T T T T A T T A G G C A G A A T C C A G A T G C T C A A G G C C C T T C A T A A T A T C C C C C A G T T T A G T A G T T G G A
T T A G G G A C A A A G G A A C C T T T A A T A G A A A T T G G A C A G C A G A A A A G C G A G C T T A G T G A T A C T T G T G G G C C A G G G C A T T A G C C A C
C A G C C A C C A C T T T C T G A T A G G C A G C C T G C A C T G G T G G G G T G A A T T C T T T G C C A A A G T G A T G G G C C A G C A C A G A C C A G C A C
T T G C C C A G G A G C T G T G G G A G G A A G A T A A G A G G T A T G A A C A T T A G C A A A A A G G G C C T A G C T T G G A C T C A G A A T A A T C C A G C C
T A T C C C A A C C A T A A A A T A A A A G C A G A A T G G T A G C T G G A T T G T A G C T G C T A T T A G C A A T A T G A A A C C C T T T A C A T C A G T T A C A A
T A T A T G C A G A A A T A T T A T A T G C A G A G A T A T T G C T A T T A G C C T T A A C C C A G A A A T T A C T A G T T A T T C T T T A G A A A T G G T G C A
A G A G G C A T G A T A C A T T G T A T T A T T G C C C T G A A A A G A G A T A T A G G G A A A T A T A G A A A T A A G A T A A A C A A A A G T A T
T A A A A G A A A A A G C A T T T T T A A A A T T A C A A A T G C A A A A T A C C C T G A T T T G G T C A A A T A G T G T G T A C A C A T A T A A A A C A T T A
A C T T T A A C C C A T A A A T A T G T A A T G A T T A T G T A T C A A T T A A A A T A A A A G A A A A T A A A G T A G G G A G A T T A T G A A T A T G C A A A
A A G C A C A C A T A T T C C A A A T A G T A A T G T A C T A G G C A G A C T G T G T A A A G T T T T T T T A A G T T A C T T A A T G T A T C T C A G A G A T
T T T C C T T T T G T T A T A C A A T G T T A A G G C A T T A A G T A A A T A G T A A A A A T T G C G A G A A G A A A A A A A A A A G A A G C A A G A A T T A
A C A A A G A A A A C A A T T G T A T A G C A A G C A A A T A A A G A A A C T A A A A C A A T C C T G A G A C T T C C A C A C T G A T G C A A T T C G T C
G T T T C C C A T T C T A A A C T G A C C C T G T T A C T T A C C C T C C T A T G A C A C T G A A C T T A C C A T A G A A A A G A A G G G A A A G A A A A C
T C A A G C G T C C C A T A G A C T G A C C C T G A A G T T C T C A G A T C C A C G T G C A G C T T T G T C A C A G T G C A G C T C A C T C A G T T G G C A A A G G
G C C C T T G A G G T T G T C C A G G T G A G C C A G G C C A T C A T A A A G G C A C C G A G C A C T T T C T T G C C A T G A G C C T T C A C C T T A G G G T T G C
C A T A A C A G C A T C A G G A G T G G A C A G A T C C C C A A A G G A C T C A A A G A A C C C T C T G G G T C C A A G G G T A G A C C A C C A G C C A T A A G G G
G G G A A A A T A G A C C A A T A G C C A G A G A G A G T C A G T C A T C A G A A A C C C A A A G A G A G C T T C T C T G T C T C C A C A T G C C A C G T T C T A
T G G T C T C C T T A A A C C T G T C T T G T A A C C T T G A T A C A A C C T G C C C A G G G C C T C A C C A C A C T T C A T C C A C G T T C A C C T T G C C C
A C A G G G C A G T A A C G G C A G A C T T C T C C T C A G G A G T C A G A T G C A C C A T G G T G T C T G T T T G A G G T T G C T A G T G A A C A C A G T T G T G T
A G A A G C A A A T G T
```

Total Number Of Hydrogen Bond Involved In This Sequence Is 3861
And The Length Of The DNA Is 1608;
C:\Users\18378004\Desktop\Lab Record Perl>

(Output)

Experiment No. 02

Date: 17/07/2019

Counting Number And Percentage Of Aliphatic Amino Acid**Aim:**

To Get The Total Number And Percentage Of Aliphatic Amino Acid Residues Of The Uniprot ID Q96AT9

Algorithm:


```
#!/usr/bin/perl
print "Please Enter The File Name Or Path\n";
$seqfile = <STDIN>;
chomp($seqfile);

open (PROTEINFILE, $seqfile);

@protein = <PROTEINFILE>;
close PROTEINFILE;
chomp(@protein);
$len=@protein;
@protein1;
$i=0,$j=0;
for($i=0;$i<$len;$i++){
    if(@protein[$i]=~/>+/{
    }
    else{
        @protein1[$j]=@protein[$i];
        $j=$j+1;
    }
}
$protein1=join("",@protein1);
@protein2=split("", $protein1);
$len2=@protein2;
$A=0;
for($i=0;$i<$len2;$i++){
    if(@protein2[$i] eq "A"){ $A=$A+1;}
    if(@protein2[$i] eq "G"){ $A=$A+1;}
    if(@protein2[$i] eq "I"){ $A=$A+1;}
    if(@protein2[$i] eq "L"){ $A=$A+1;}
```


```
        if(@protein2[$i] eq "V"){ $A=$A+1;}  
    }  
    $value=($A/$len2)*100;  
    print "The Total Length Value Of The Protein Is $len2 And Having $A  
    Aliphatic Amino Acids Having $value Percent Aliphatic Residue.";
```

Result:

 C:\WINDOWS\system32\cmd.exe - perl 2.pl

```
C:\Users\18378004\Desktop\Lab Record Perl\2>perl 2.p  
Please Enter The File Name Or Path  
Q96AT9.txt
```

(Input)

 C:\WINDOWS\system32\cmd.exe

```
C:\Users\18378004\Desktop\Lab Record Perl\2>perl 2.pl  
Please Enter The File Name Or Path  
Q96AT9.txt  
The Total Length Value Of The Protein Is 228 And Having 87 Aliphatic Amino Acids Having 38.1578947368421 Percent Aliphatic Residue.  
C:\Users\18378004\Desktop\Lab Record Perl\2>
```

(Output)

Experiment No. 03

Date: 24/07/2019

Counting Molecular Weight Of A Protein Sequence In FASTA Format

Aim:

To Get The Protein Sequence as FASTA Format And Get The Molecular Weight Of The Protein.

Algorithm:

```
#!/usr/bin/perl
```

```
print "Please Enter The File Name Or Path\n";
```

```
$seqfile = <STDIN>;
```

```
open (PROTEINFILE, $seqfile);
```

```
@protein = <PROTEINFILE>;
```

```
close PROTEINFILE;
```

```
chomp(@protein);
```

```
$len=@protein;
```

```
@protein1;
```

```
$i=0,$j=0;
```

```
for($i=0;$i<$len;$i++){
```

```
    if(@protein[$i]=~/>+/{
```

```
    }
```

```
    else{
```

```
        @protein1[$j]=@protein[$i];
```

```
        $j=$j+1;
```

```
    }
```


```
}
```

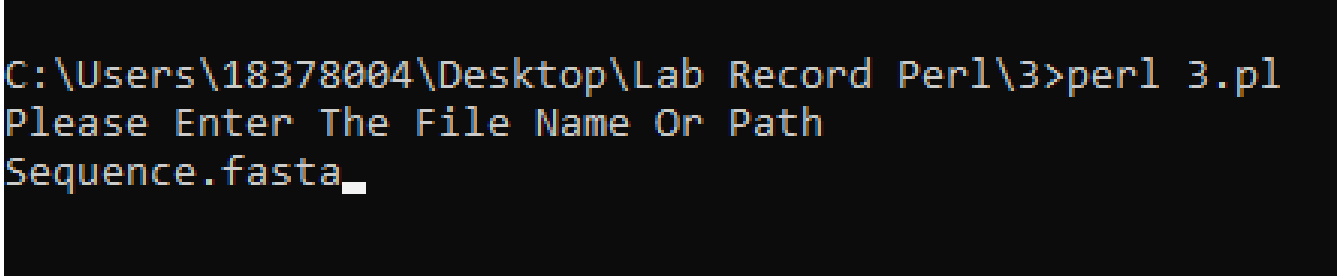
```
$protein1=join(" ",@protein1);
```

```
@protein2=split("", $protein1);  
$len2=@protein2;  
$A=0;  
for($i=0;$i<$len2;$i++){  
    if(@protein2[$i] eq "A"){ $A=$A+89.094;}  
    if(@protein2[$i] eq "R"){ $A=$A+174.203;}  
    if(@protein2[$i] eq "N"){ $A=$A+132.119;}  
    if(@protein2[$i] eq "D"){ $A=$A+133.104;}  
    if(@protein2[$i] eq "C"){ $A=$A+121.154;}  
    if(@protein2[$i] eq "Q"){ $A=$A+146.146;}  
    if(@protein2[$i] eq "E"){ $A=$A+147.131;}  
    if(@protein2[$i] eq "G"){ $A=$A+75.067;}  
    if(@protein2[$i] eq "H"){ $A=$A+155.156;}  
    if(@protein2[$i] eq "I"){ $A=$A+131.175;}  
    if(@protein2[$i] eq "L"){ $A=$A+131.175;}  
    if(@protein2[$i] eq "K"){ $A=$A+146.189;}  
    if(@protein2[$i] eq "M"){ $A=$A+149.108;}  
    if(@protein2[$i] eq "F"){ $A=$A+165.192;}  
    if(@protein2[$i] eq "P"){ $A=$A+115.132;}  
    if(@protein2[$i] eq "S"){ $A=$A+105.093;}  
    if(@protein2[$i] eq "T"){ $A=$A+119.119;}  
    if(@protein2[$i] eq "W"){ $A=$A+204.228;}  
    if(@protein2[$i] eq "Y"){ $A=$A+181.191;}  
    if(@protein2[$i] eq "V"){ $A=$A+117.148;}  
    if(@protein2[$i] eq "U"){ $A=$A+168.064;}
```

```
        if(@protein2[$i] eq "O"){ $A=$A+255.313;}  
    }  
    print "Total Molecular Weight Value Of The Protein Is $A Dalton";
```


Result:

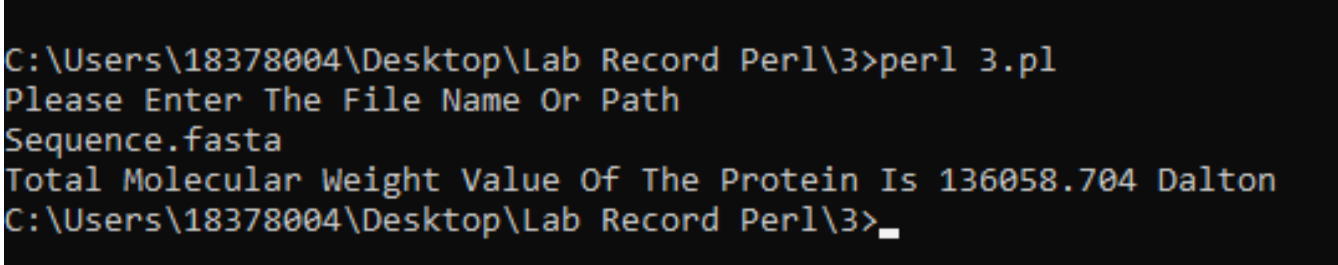
 C:\WINDOWS\system32\cmd.exe - perl 3.pl



```
C:\Users\18378004\Desktop\Lab Record Perl\3>perl 3.pl  
Please Enter The File Name Or Path  
Sequence.fasta_
```

(Input)

 C:\WINDOWS\system32\cmd.exe



```
C:\Users\18378004\Desktop\Lab Record Perl\3>perl 3.pl  
Please Enter The File Name Or Path  
Sequence.fasta  
Total Molecular Weight Value Of The Protein Is 136058.704 Dalton  
C:\Users\18378004\Desktop\Lab Record Perl\3>_
```

(Output)

Experiment No. 04

Date: 31/07/2019

Convert The Single Letter Amino Acid Code To Three Letter Code

Aim: To Convert The Single Letter Amino Acid Code To Three Letter Code Using Hashes With Sub-routine For The Protein Sequence

"MRWLLLYYALCFSLSKASAHTVELNNMFGQIQSPGYPSYPSDSEVTWNITVPDGFRIKL"

Algorithm:

```
#!/usr/bin/perl

$protein="MRWLLLYYALCFSLSKASAHTVELNNMFGQIQSPGYPSYPSDSEVTWNITVP
DGFRIKL";

$three, $single; @three=;

@protein=split("", $protein);

$len=length $protein;

for($i=0; $i<$len; $i++){

$single=substr($protein,$i,1);

@three[$i]= three($single);}

print "Three Letter Code Conversion Of $protein Is\n@three\n";

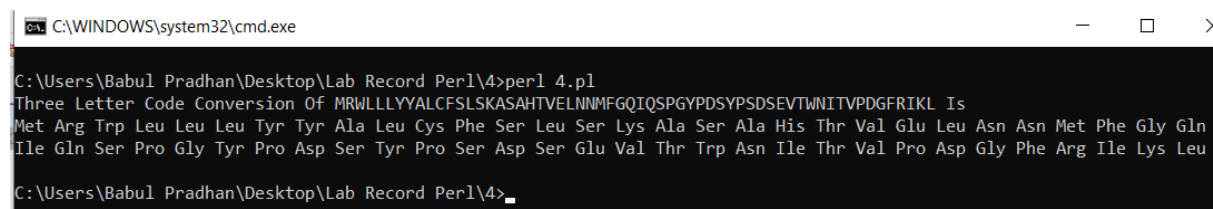
sub three {my ($single)=@_; $single=uc $single;

my (%aminoAcid) =
('A'=>'Ala', 'R'=>'Arg', 'N'=>'Asn', 'D'=>'Asp', 'C'=>'Cys', 'E'=>'Glu', 'Q'=>'Gln', 'G'=>'Gly',
'H'=>'His', 'I'=>'Ile', 'L'=>'Leu', 'K'=>'Lys', 'M'=>'Met', 'F'=>'Phe', 'P'=>'Pro', 'S'=>'Ser', 'T'
=>'Thr', 'W'=>'Trp', 'Y'=>'Tyr', 'V'=>'Val');

if (exists $aminoAcid{$single}){return $aminoAcid{$single};}

else {print STDERR "Not Exist \"$single\" \n";exit;}}
```

Result:



```
C:\WINDOWS\system32\cmd.exe

C:\Users\Babul Pradhan\Desktop\Lab Record Perl\4>perl 4.pl
Three Letter Code Conversion Of MRWLLLYYALCFSLSKASAHTVELNNMFGQIQSPGYPSYPSDSEVTWNITVPDGFRIKL Is
Met Arg Trp Leu Leu Leu Tyr Tyr Ala Leu Cys Phe Ser Leu Ser Lys Ala Ser Ala His Thr Val Glu Leu Asn Asn Met Phe Gly Gln
Ile Gln Ser Pro Gly Tyr Pro Asp Ser Tyr Pro Ser Asp Ser Glu Val Thr Trp Asn Ile Thr Val Pro Asp Gly Phe Arg Ile Lys Leu

C:\Users\Babul Pradhan\Desktop\Lab Record Perl\4>_
```

(Output)

Experiment No. 05

Date:07/08/2019

Calculate Di Nucleotide Frequency

Aim:

To Calculate The Di-Nucleotide Frequencies In Overlapping/Frame Wise Of AA, AT, AG & AC For The DNA Sequence Accession:GU190379.1

Algorithm:

```
#!/usr/bin/perl
print "Please Enter The File Name Or Path\n";
$seqfile = <STDIN>;
chomp($seqfile);

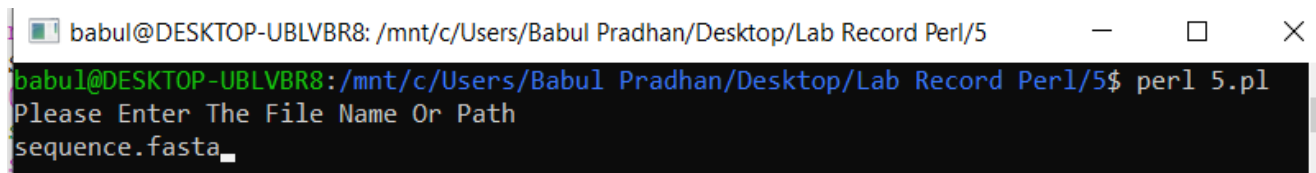
open (SEQFILE, $seqfile);
@sequence = <SEQFILE>;
close SEQFILE;
chomp(@sequence);
#@dna=split("", $sequence);
$len=@sequence;
@sequence1;
$i=0,$j=0;
for($i=0;$i<$len;$i++){
    if(@sequence[$i]=~/>+/{
    }
    else{
        @sequence1[$j]=@sequence[$i];
        $j=$j+1;
    }
}
$sequence1=join("", @sequence1);
@dna=split("", $sequence1);
$len1=@dna;
$AA=0,$AT=0,$AG=0,$AC=0,$i,$j;
print "Total Length Of The Sequence Is $len\n";
for($i=0;$i<$len1;$i++){
    $j=$i+1;
    if(@dna[$i] eq A && @dna[$j] eq A){
```

```

        $AA=$AA+1;
    }
    if(@dna[$i] eq A && @dna[$j] eq T){
        $AT=$AT+1;
    }
    if(@dna[$i] eq A && @dna[$j] eq C){
        $AC=$AC+1;
    }
    if(@dna[$i] eq A && @dna[$j] eq G){
        $AG=$AG+1;
    }
}
$freqAA=($AA/$len1)*100;
$freqAT=($AT/$len1)*100;
$freqAC=($AC/$len1)*100;
$freqAG=($AG/$len1)*100;
print "Total AA is $AA, AT is $AT, AC is $AC, And AG is $AG\n";
print "The Di Nucleotide Frequency Of\nAA is $freqAA %\nAT is $freqAT
%\nAC is $freqAC %\nAG is $freqAG %\n";

```

Result:

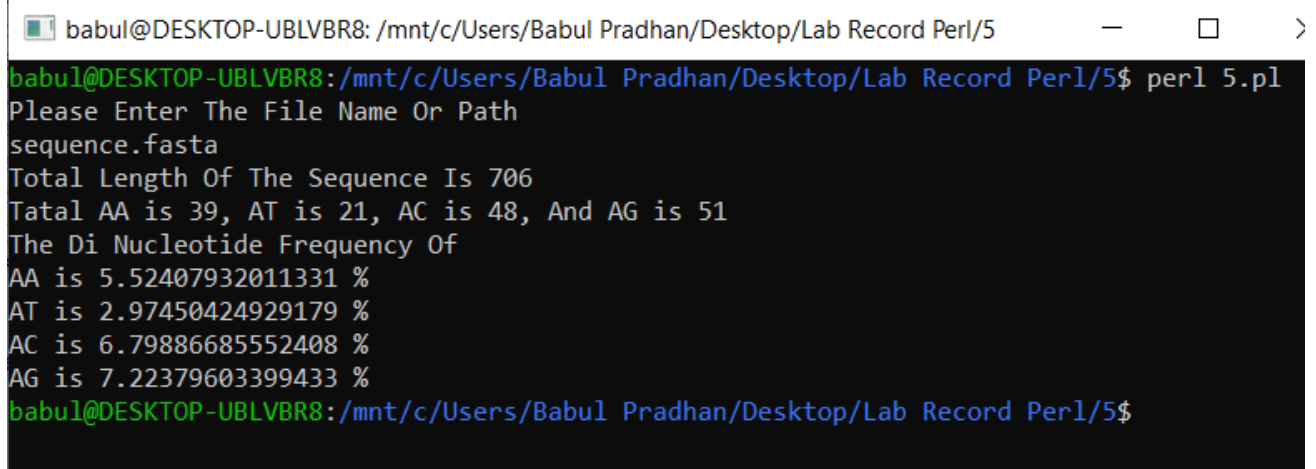


```

babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/5
babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/5$ perl 5.pl
Please Enter The File Name Or Path
sequence.fasta_

```

(Input)



```

babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/5
babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/5$ perl 5.pl
Please Enter The File Name Or Path
sequence.fasta
Total Length Of The Sequence Is 706
Total AA is 39, AT is 21, AC is 48, And AG is 51
The Di Nucleotide Frequency Of
AA is 5.52407932011331 %
AT is 2.97450424929179 %
AC is 6.79886685552408 %
AG is 7.22379603399433 %
babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/5$

```

(Output)

Use Of ARGV

Experiment No. 07

Date: 21/08/2019

Regular Expression

Aim:

To Retrieve The Sequence Portion Only For TheGiven DNA In GenBank Format Using Regular Expression.

Algorithm:

```
#!/usr/bin/perl
@seqfile=<>;@seq,$j,$len1;
$len=@seqfile;
for($i=0;$i<$len;$i++){
    if(@seqfile[$i]=~/ORIGIN/){$j=$i+1;$len1=$len-$j;}
}
for($i=$j;$i<$len;$i++){
    push(@seq,@seqfile[$i]);}
$j=0, @seq4;
chomp(@seq);
$seq=join(" ",@seq);
@seq1=split("",$seq);
$seq2=join("",$seq1);
@seq3=split("",$seq2);
$len2=@seq3;
for($i=0;$i<$len2;$i++){if(@seq3[$i]=~/[atgcn]/ ||
@seq3[$i]=~/[ATGCN]/) {push(@seq4,@seq3[$i]);}
    else{$i=$i+1;}}
$len3=@seq4;
print "The Length Of The Sequence Is $len3 And The Sequ
Is\n@seq4\n";
Result:
```

Result:

[illegible]

(Output)

Experiment No. 08

Date: 04/09/2019

Retrieve Parts From A PDB File**Aim:**

To Retrieve Atom, Heteroatom, Helix And Sheets Parts From A PDB File.

Algorithm:

```
#!/usr/bin/perl
@seqfile = <>;
@ATOM,@HETATM,@HELIX,@SHEET;
$j;
$len1,$l;
$len=@seqfile;
for($i=0;$i<$len;$i++){
    if(@seqfile[$i]=~/REMARK/){$i=$i+1;}
    else{
        if(@seqfile[$i]=~/ATOM/){
            push(@ATOM,@seqfile[$i]);}
        }
    }
for($i=0;$i<$len;$i++){
    if(@seqfile[$i]=~/REMARK/){$i=$i+1;}
    else{
        if(@seqfile[$i]=~/HETATM/){
            push(@HETATM,@seqfile[$i]);}
        }
    }
for($i=0;$i<$len;$i++){
    if(@seqfile[$i]=~/HELIX/){
        push(@HELIX,@seqfile[$i]);}
    }
for($i=0;$i<$len;$i++){
    if(@seqfile[$i]=~/SHEET/){
        push(@SHEET,@seqfile[$i]);}
    }
}
```

```
print "The ATOM Part Is\n @ATOM\nThe HETEROATOM PART Is\n
@HETATM\nThe HELIX Part Is\n @HELIX\nThe SHEET PART Is\n
@SHEET";
```

Result:

```
babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/8
babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/8$ perl
The ATOM Part Is
ATOM 1 N ARG A 126 75.471 24.949 -16.252 1.00 51.01 N
ATOM 2 CA ARG A 126 76.069 25.833 -15.206 1.00 50.94 C
ATOM 3 C ARG A 126 76.858 25.017 -14.193 1.00 50.75 C
ATOM 4 O ARG A 126 76.355 24.024 -13.661 1.00 50.85 O
ATOM 5 CB ARG A 126 74.982 26.654 -14.504 1.00 51.02 C
ATOM 6 N GLN A 127 78.093 25.448 -13.936 1.00 50.40 N
ATOM 7 CA GLN A 127 78.984 24.781 -12.989 1.00 49.94 C
ATOM 8 C GLN A 127 79.252 25.689 -11.782 1.00 49.44 C
ATOM 9 O GLN A 127 80.276 26.383 -11.728 1.00 49.42 O
ATOM 10 CB GLN A 127 80.298 24.377 -13.680 1.00 50.07 C
ATOM 11 CG GLN A 127 81.116 23.335 -12.915 1.00 50.26 C
ATOM 12 CD GLN A 127 82.510 23.115 -13.490 1.00 50.10 C
ATOM 13 OE1 GLN A 127 82.676 22.859 -14.686 1.00 50.14 O
ATOM 14 NE2 GLN A 127 83.520 23.200 -12.628 1.00 50.17 N
ATOM 15 N TRP A 128 78.322 25.669 -10.824 1.00 48.59 N
ATOM 16 CA TRP A 128 78.369 26.519 -9.624 1.00 47.70 C
ATOM 17 C TRP A 128 79.533 26.200 -8.685 1.00 46.89 C
ATOM 18 O TRP A 128 80.050 25.079 -8.680 1.00 46.81 O
ATOM 19 CB TRP A 128 77.068 26.384 -8.826 1.00 48.05 C
ATOM 20 CG TRP A 128 75.817 26.737 -9.565 1.00 48.26 C
ATOM 21 CD1 TRP A 128 74.882 25.869 -10.055 1.00 48.52 C
ATOM 22 CD2 TRP A 128 75.350 28.053 -9.881 1.00 48.47 C
ATOM 23 NE1 TRP A 128 73.861 26.565 -10.663 1.00 48.95 N
ATOM 24 CE2 TRP A 128 74.124 27.907 -10.571 1.00 48.99 C
ATOM 25 CE3 TRP A 128 75.848 29.341 -9.651 1.00 48.63 C
ATOM 26 CZ2 TRP A 128 73.390 29.005 -11.036 1.00 48.91 C
ATOM 27 CZ3 TRP A 128 75.120 30.432 -10.114 1.00 48.59 C
ATOM 28 CH2 TRP A 128 73.905 30.257 -10.798 1.00 48.77 C
```

```
babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/8
ATOM 3845 C LYS B 389 22.723 48.480 10.903 1.00 45.26 C
ATOM 3846 O LYS B 389 21.684 48.796 11.488 1.00 45.39 O
ATOM 3847 CB LYS B 389 23.358 46.068 10.582 1.00 45.16 C
The HETEROATOM PART Is
HETATM 3849 N1 AK8 A 404 64.440 30.479 5.073 1.00 21.69 N
HETATM 3850 N3 AK8 A 404 65.414 26.024 6.132 1.00 24.74 N
HETATM 3851 C4 AK8 A 404 64.766 30.238 2.383 1.00 20.57 C
HETATM 3852 C5 AK8 A 404 65.266 28.312 5.506 1.00 23.39 C
HETATM 3853 C6 AK8 A 404 65.818 28.677 6.732 1.00 24.62 C
HETATM 3854 C7 AK8 A 404 66.157 27.686 7.646 1.00 25.78 C
HETATM 3855 C8 AK8 A 404 65.940 26.353 7.306 1.00 24.87 C
HETATM 3856 C10 AK8 A 404 63.627 32.912 6.186 1.00 21.60 C
HETATM 3857 C13 AK8 A 404 65.478 23.375 2.524 1.00 31.65 C
HETATM 3858 C15 AK8 A 404 63.462 23.061 3.803 1.00 32.08 C
HETATM 3859 C17 AK8 A 404 62.333 24.937 5.070 1.00 31.80 C
HETATM 3860 C20 AK8 A 404 67.123 18.046 3.580 1.00 38.42 C
HETATM 3861 C21 AK8 A 404 67.872 16.893 3.807 1.00 39.45 C
HETATM 3862 C22 AK8 A 404 67.439 15.660 3.319 1.00 39.14 C
HETATM 3863 C24 AK8 A 404 65.502 16.738 2.373 1.00 39.09 C
HETATM 3864 N4 AK8 A 404 63.613 32.624 4.733 1.00 22.49 N
HETATM 3865 C1 AK8 A 404 64.108 31.482 4.241 1.00 21.25 C
HETATM 3866 N2 AK8 A 404 64.275 31.347 2.912 1.00 21.42 N
HETATM 3867 C3 AK8 A 404 65.112 29.185 3.222 1.00 22.62 C
HETATM 3868 C2 AK8 A 404 64.930 29.324 4.603 1.00 23.14 C
HETATM 3869 C9 AK8 A 404 65.074 26.953 5.223 1.00 25.37 C
HETATM 3870 O1 AK8 A 404 64.527 26.581 4.025 1.00 29.65 O
HETATM 3871 C11 AK8 A 404 64.487 25.248 3.709 1.00 31.26 C
HETATM 3872 C16 AK8 A 404 63.460 24.410 4.164 1.00 31.52 C
HETATM 3873 C12 AK8 A 404 65.486 24.724 2.885 1.00 31.52 C
```

(ATOM,HETATOM)

```
babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/8
HETATM 3988 O HOH B 453 50.453 63.110 15.617 1.00 41.57 O
HETATM 3989 O HOH B 454 57.634 43.303 41.212 1.00 42.04 O
HETATM 3990 O HOH B 455 38.446 37.799 14.827 1.00 32.75 O
The HELIX Part Is
HELIX 1 1 ALA A 129 GLU A 131 5 3
HELIX 2 2 LYS A 166 GLY A 173 1 8
HELIX 3 3 VAL A 174 HIS A 187 1 14
HELIX 4 4 THR A 217 SER A 226 1 10
HELIX 5 5 ASP A 229 SER A 249 1 21
HELIX 6 6 LYS A 258 GLU A 260 5 3
HELIX 7 7 PRO A 297 ILE A 301 5 5
HELIX 8 8 LYS A 309 GLY A 325 1 17
HELIX 9 9 THR A 333 VAL A 344 1 12
HELIX 10 10 THR A 353 LEU A 364 1 12
HELIX 11 11 ASN A 367 ARG A 371 5 5
HELIX 12 12 MET A 373 GLU A 379 1 7
HELIX 13 13 HIS A 380 SER A 387 1 8
HELIX 14 14 ALA B 129 GLU B 131 5 3
HELIX 15 15 LYS B 166 GLY B 173 1 8
HELIX 16 16 VAL B 174 SER B 186 1 13
HELIX 17 17 THR B 217 SER B 226 1 10
HELIX 18 18 ASP B 229 SER B 249 1 21
HELIX 19 19 LYS B 258 GLU B 260 5 3
HELIX 20 20 PRO B 297 GLU B 302 1 6
HELIX 21 21 LYS B 309 GLY B 325 1 17
HELIX 22 22 THR B 333 ARG B 343 1 11
HELIX 23 23 THR B 353 LEU B 364 1 12
HELIX 24 24 ASN B 367 ARG B 371 5 5
HELIX 25 25 LEU B 374 GLU B 379 1 6
```

```
babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/8
HELIX 20 20 PRO B 297 GLU B 302 1 6
HELIX 21 21 LYS B 309 GLY B 325 1 17
HELIX 22 22 THR B 333 ARG B 343 1 11
HELIX 23 23 THR B 353 LEU B 364 1 12
HELIX 24 24 ASN B 367 ARG B 371 5 5
HELIX 25 25 LEU B 374 GLU B 379 1 6
HELIX 26 26 HIS B 380 SER B 387 1 8
The SHEET PART Is
SHEET 1 A 5 PHE A 133 GLY A 142 0
SHEET 2 A 5 GLY A 145 GLU A 152 -1 O VAL A 147 N LEU A 139
SHEET 3 A 5 ILE A 158 PHE A 165 -1 O LEU A 159 N ALA A 150
SHEET 4 A 5 ARG A 205 LEU A 210 -1 O VAL A 206 N LEU A 164
SHEET 5 A 5 LEU A 196 HIS A 201 -1 N PHE A 200 O TYR A 207
SHEET 1 B 2 LEU A 262 LEU A 264 0
SHEET 2 B 2 LEU A 270 ILE A 272 -1 O LYS A 271 N LEU A 263
SHEET 1 C 5 PHE B 133 GLY B 142 0
SHEET 2 C 5 GLY B 145 GLU B 152 -1 O VAL B 147 N LEU B 139
SHEET 3 C 5 ILE B 158 PHE B 165 -1 O LEU B 161 N TYR B 148
SHEET 4 C 5 ARG B 205 LEU B 210 -1 O VAL B 206 N LEU B 164
SHEET 5 C 5 LEU B 196 HIS B 201 -1 N PHE B 200 O TYR B 207
SHEET 1 D 2 LEU B 262 LEU B 264 0
SHEET 2 D 2 LEU B 270 ILE B 272 -1 O LYS B 271 N LEU B 263
babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/8$
```

(HELIX,SHEET)

Experiment No. 09

Date: 25/09/2019

Count Polar Amino Acid Using File Handling

Aim:

To Store The 5 Given Protein Sequences In FASTA Format In A Single File And To Find The Length, Number Of Polar Amino Acids For Each Sequence. And Then Display And Write The Results To A File Based On Number Of Polar Amino Acids In Descending Order.

Sample:

The Given Protein Sequence Accession Are

O23533

P11150

P77044

Q63HQ2

Q71U36

Algorithm:

```
#!/usr/bin/perl
print "Please Enter The File Name Or Path\n";
$seqfile =<STDIN>;

open (PROTEINFILE, $seqfile);

@protein = <PROTEINFILE>;
close PROTEINFILE;
$a=join("",@protein);
$lenna=@protein;
$j=0;
@protein1,@protein2,@protein3,@protein4,@protein5,$protein5;

@proteina,@proteina,@proteina,@proteina,@proteina;
$a=0,$b=0,$c=0,$d=0;$e=0;

for($i=1;$i<$lenna;$i++){
    push(@protein1, @protein[$i]);
```

```

        if(@protein[$i+1]=~/>+/{
            $j=$i+2;
            $i=$lena;
        }
    }
    for($i=$j;$i<$lena;$i++){
        push(@protein2, @protein[$i]);
        if(@protein[$i+1]=~/>+/{
            $j=$i+2;
            $i=$lena;}
    }

    for($i=$j;$i<$lena;$i++){
        push(@protein3, @protein[$i]);
        if(@protein[$i+1]=~/>+/{
            $j=$i+2;
            $i=$lena;}
    }

    for($i=$j;$i<$lena;$i++){
        push(@protein4, @protein[$i]);
        if(@protein[$i+1]=~/>+/{
            $j=$i+2;
            $i=$lena;}
    }

    for($i=$j;$i<$lena;$i++){
        push(@protein5, @protein[$i]);
        if(@protein[$i+1]=~/>+/{
            $j=$i+2;
            $i=$lena;}
    }

    chomp(@protein1);
    chomp(@protein2);
    chomp(@protein3);
    chomp(@protein4);
    chomp(@protein5);

    $protein1=join("",@protein1);
    @protein1=split("", $protein1);

```

```

$len1=@protein1;
for($i=0;$i<$len1;$i++){
    if(@protein1[$i]=~/\s/){$i=$i+1;}
    else{
        push(@proteina, @protein1[$i]);
    }
}
$lenna=@proteina;

```

```

$protein2=join("",@protein2);
@protein2=split("", $protein2);
$len2=@protein1;
for($i=0;$i<$len2;$i++){
    if(@protein2[$i]=~/\s/){$i=$i+1;}
    else{
        push(@proteinb, @protein2[$i]);
    }
}
$lenb=@proteinb;

```

```

$protein3=join("",@protein3);
@protein3=split("", $protein3);
$len3=@protein3;
for($i=0;$i<$len3;$i++){
    if(@protein3[$i]=~/\s/){$i=$i+1;}
    else{
        push(@proteinc, @protein3[$i]);
    }
}
$lenc=@proteinc;

```

```

$protein4=join("",@protein4);
@protein4=split("", $protein4);
$len4=@protein4;
for($i=0;$i<$len4;$i++){
    if(@protein4[$i]=~/\s/){$i=$i+1;}
    else{

```

```

        push(@proteind, @protein4[$i]);
    }
}
$lend=@proteind;

$protein5=join("",@protein5);
@protein5=split("", $protein5);
$len5=@protein5;
for($i=0;$i<$len5;$i++){
    if(@protein5[$i]=~/\s/){$i=$i+1;}
    else{
        push(@proteine, @protein5[$i]);
    }
}
$lene=@proteine;
chomp(@proteina);
chomp(@proteinb);
chomp(@proteinc);
chomp(@proteind);
chomp(@proteine);
for($i=0;$i<$len1;$i++){
    if(@proteina[$i] eq "Q" || @proteina[$i] eq "N" || @proteina[$i] eq
"H" || @proteina[$i] eq "S" || @proteina[$i] eq "T" || @proteina[$i] eq
"Y" || @proteina[$i] eq "C"){ $a=$a+1;}

}

for($i=0;$i<$len2;$i++){
    if(@proteinb[$i] eq "Q" || @proteinb[$i] eq "N" || @proteinb[$i]
eq "H" || @proteinb[$i] eq "S" || @proteinb[$i] eq "T" || @proteinb[$i] eq
"Y" || @proteinb[$i] eq "C"){ $b=$b+1;}

}

for($i=0;$i<$len3;$i++){

```

```

        if(@proteinc[$i] eq "Q" || @proteinc[$i] eq "N" || @proteinc[$i] eq
"H" || @proteinc[$i] eq "S" || @proteinc[$i] eq "T" || @proteinc[$i] eq
"Y" || @proteinc[$i] eq "C"){ $c=$c+1;}

```

```

}

```

```

for($i=0;$i<$len4;$i++){
    if(@proteind[$i] eq "Q" || @proteind[$i] eq "N" || @proteind[$i] eq
"H" || @proteind[$i] eq "S" || @proteind[$i] eq "T" || @proteind[$i] eq
"Y" || @proteind[$i] eq "C"){ $d=$d+1;}

```

```

}

```

```

for($i=0;$i<$len5;$i++){
    if(@proteine[$i] eq "Q" || @proteine[$i] eq "N" || @proteine[$i] eq
"H" || @proteine[$i] eq "S" || @proteine[$i] eq "T" || @proteine[$i] eq
"Y" || @proteine[$i] eq "C"){ $e=$e+1;}

```

```

}

```

```

%polar=('a'=>"@proteina\nLength Is $lena\nNumber Of Polar Amino Acid
Is $a\n",'b'=>"@proteind\nLength Is $lenb\nNumber Of Polar Amino
Acid Is $b\n",'c'=>"@proteinc\nLength Is $lenc\nNumber Of Polar
Amino Acid Is $c\n",'d'=>"@proteind\nLength Is $lend\nNumber Of
Polar Amino Acid Is $d\n",'e'=>"@proteine\nLength Is $lene\nNumber
Of Polar Amino Acid Is $e\n");

```

```

@polarlen=($a,$b,$c,$d,$e);

```

```

@rev=sort{$b <=> $a} @polarlen;

```

```

$seqout= "protein.txt";

```

```

unless ( open(PROTEIN, ">$seqout" ) ) {
    print "cannot open file \"$seqout\" to write to !!!\n\n";
    exit;
}

```

```

for($i=0;$i<5;$i++){

```

```

    if(@rev[$i] eq $a){print PROTEIN "$polar{a}\n"; print
"$polar{a}\n";}

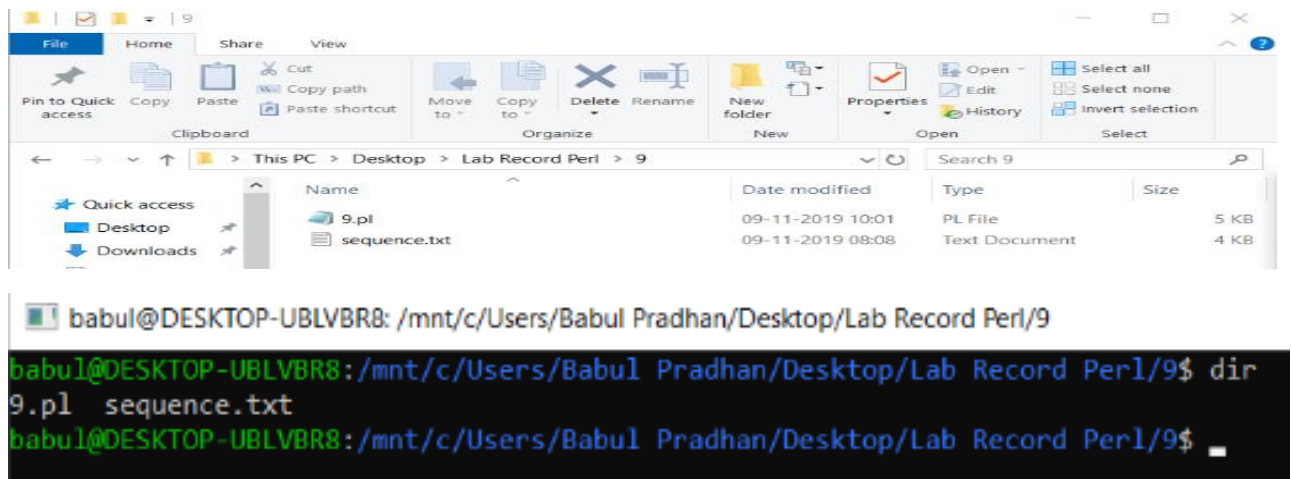
```

```

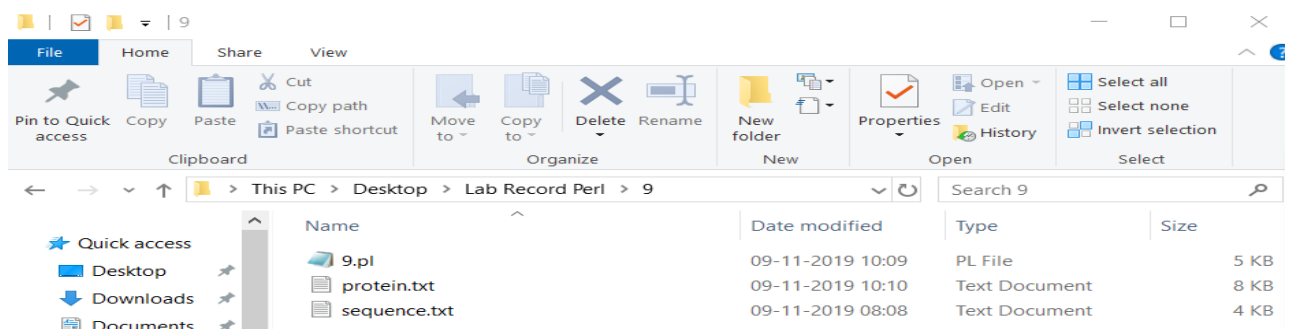
if(@rev[$i] eq $b){print PROTEIN "$polar{b}\n"; print
"$polar{b}\n";}
if(@rev[$i] eq $c){print PROTEIN "$polar{c}\n"; print
"$polar{c}\n";}
if(@rev[$i] eq $d){print PROTEIN "$polar{d}\n"; print
"$polar{d}\n";}
if(@rev[$i] eq $e){print PROTEIN "$polar{e}\n"; print
"$polar{e}\n";}
}
close (PROTEIN);
print "The New Output File Has Been Generated In The Current Working
Directory.\nHave A Great Day Ahead.\n";

```

Result:



(Before Running The Code)




```

babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/9
babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/9$ perl 9.pl
Please Enter The File Name Or Path
sequence.txt
MDLIRGVLLRLLLLASSLGGPAGVSLRAAIRKPGKVGPPDLIKLGLALNCTAFSIIQWKMPRHGSPILGTYTVFYSEVGADKSLQEQH
SVPLSRDIPPTTEEVIGDLKPGTEYRVSIAAYSQAAGRLSSPRHVTTLSQDSCLPAPAAPQPHVIVVSDSEVALSWKPGASEGSAP
QYYSVEFRPDDFKKWTSTIHERIQMDSMVIKGLDPTDNYQFAVRAMNSHGSPSRSWPSDIIRTLCEAGSGRYGPRYITDMGAGE
DEGFEDDDLDLDSFEVEVKPLPATKGGNKKFLVESKKMSISPKTISRILIPPTSSASLPVTTVAPQPIPIQRKGGKNGVAIMSRLFDM
PCDETLCSADSFVCVNYTWGGSRCQCTLGKGGESCSSEDIQYVQFFGHSHSVTFEPLKMSYQAFQITLFEFRAEAEGLLLYCGENEH
RGDFMSLAITIRRSLOFRFNCCTGVAIIIVSETKIKLGGWHTVMLYRDNGLLQILNMGTPVTGQSQGQYSKITFRTPLYLGGAPSA
YWLVRATGTNRGFGQCVQSLAVGRRIDMRPWLGLKALSAGDVGECSSGICDEASCIHGGTCTAIKADSVICLCPGLGFKGRHEDAF
TLTIPOFRESLSRYAATPWLEPQHLYLSFMEFEITFRPDSGDGVLLYSYDTGSKDLSINLAGGHVEFRFDGSGTGLVLRSEDPLT
LGNWHELVRVSRITAKNGILQVQKQKIVEGMEGGFTQIKCNTDIFIGGVPNYDDVKKNSGVLPKPSGSIQKIIILNDRTIHKHDFTS
GVNENAAHPCVRAPCAHGGSCRPKEGYDCDCPLGFEGHLHCQKECGNYCLNTTIEAIEIPQFGRSYLTYNPDILKRVSGSRSNV
MRFKTTAKDGLLLWRGDSMPRPNDSFISLGLRDGAVFSYNLGSVVASIMVNGSFNDGRWHRVKAVERDQSGKITVDDYGGARTGK
SGMMRQLNIGALYVGGMKIEIALHTNRQYMRGLVGCISHFTLSTDYHISLVEDAVDGGKNINTCGAK
length Is 1001
Number Of Polar Amino Acid Is 295
MAAASSSSCRRRYDVFPSSFSGVDVRKTFLSNLLAEAFDRRSINTFMDHGIERSRTIAPELISIREARISIVIFSKNYASSTWCLDEL
VEIHNRLNDWGQLVISVFYDVPSEVRKQTGEFGVFKKTCEDKEEDQKQRWMOALVDITNIAGEDLNRNGPSEAAAMVVKKIANDVSN
KLISPSNSGDFVGTAEHLEAMNSILCLESKEARMVGIWGPSSGIGKSTIGKALYSOLFQCFHFHAFVVPVYSMKSEWEEIFLSKILG
KDIKIGGKLGVEQMLNOKKVLIVLDDVDDPEFLKTLVGETWFGPGSRIIVITQDMOLLKAHDIDLLEYVKFSPSLDLALMLCRS
AFGENSPDDDFKALAEVAVLAVAGMLPLGLSVLSSLKRRTEKEEMEMHPRFRNGLNGDIMKTLRVSYDRLLHQKQDMFLYIACLNG
FVSYVNDLLEDNVGVTVMLVEKSLIRITPDGDIEMHNLLLEKLGIEIDRKSGKNPGKRRFLTDFTDLRKLTVLGRFCTAIFRSKEL
LPIDEKSFQGMRLNLQCLSVTGMDMLPQSLVLPKLRLLLDWDRCLKCLPYFSFKADYLIQLTMMGSKLEKLWEGTVPLGSLRMMN
HGSRYLREISDLSNARNLEELNLSECRSLVTLSSSIQNAIKLTYLDMRGCTKLESPTHLNLESLEYLGLLYDNLNRNFPVFKMET
STTSPHGIEIRVENCINWKNLPGLDYLACVRCPMCEFRPNDLVRLIVRGNQMLEKLWEGVQSLASLVEMDMSECGNLTIEIPDLK
ATNWNLYLSNCKSLVTPSTIGNLQKLVRLMEKCTGLEVLPTDVLNLSLKMLDLSGCSSLRFPLISKSIKWLYLENTAIEEVPC
CIENFSLVTLVMMYCKRLKNISPNIIFRLTILKLVDTECRGVNVAMSDASVEDHSSYIPLYENIEYTRHHLPSKLTFTNDVEFNFS
CSKRIKECGILLNMTSGTSEEYINLPCVQIVADTSLTALNMELSLGQGEASSVSSYPSLEGESLCVDSMSEQQDEEIPILNSGSG
length Is 1021
Number Of Polar Amino Acid Is 289
CSKRIKECGILLNMTSGTSEEYINLPCVQIVADTSLTALNMELSLGQGEASSVSSYPSLEGESLCVDSMSEQQDEEIPILNSGSG
length Is 1021
Number Of Polar Amino Acid Is 289
MDTSPLCFSIILLVLCIFIOSSALGQSLKPEPFGRRAAQAVETNKTLEHMKTRFLLFGETNQCOIRINHPDTLQECGFNSSLPLVMI
IHGWSVDGVLENWIWQMVAAALKSQPAQPVNVGLVWITLAHDHYTIAVRNTRLVGKEVAALLRWLEESVQLSRSHVHLIGYSLGHA
VSGFAGSSGGGTHKIGRITGLDAAGPLFEGSAPSNNRSLPDDANFVDAIHTFTREHMGSLVGIKQPIGHDFYPNGGGSFQPGCHFLEL
YRHHIAQHGfNAITQTIKCSHERSVHLFIDSLHAGTQSMAYCGDMNSFSQGLCLSCCKKGRCNTLGYHVRQEPKRSKRLFLVTRA
QSPFKVYHYQFKIQFNQTETPIQTTFTHSLGLTKEKMQKIPITLGKGIASNKTYSFLLITLDVDIGELIMIKFKWNSAVWANVWD
VQTIIPWSTGPRHSGVLVKTIRVKAGETQQRMTFCSENTDDLRLRPTQKIFVKCEIKSKTSKRKIR
length Is 1039
Number Of Polar Amino Acid Is 163
MRECISIHVGQAGVQIGNACWELCYCLEHGIQPDGQMPSDKTIGGGDDSFNTFFSETGAGKVPRAVFVDLEPTVIDEVRTGTYRQL
FHPEQLITGKEDAANNYARGHYTIGKEIIDLVLDIRKLADQCTGLQGFLVFHSFGGGTSGGFTSLLMERLSVDYGKKSKLEFSIY
PAPQVSTAVEPYNSILTTHTTLEHSDCAFMVDNEAIYDICRRNLDIERPTYTNLNRLLIGQIVSSITALRFDGALNVDLTEFQTNL
VPYPRIHFPLATYAPVISAEKAYHEQLSVAEITNACFEPANMVKCDPRHGKYMACCLLYRGDVVPKDVNAAIATIKTKRTIQFVD
WCPTGFKVGINYQPPVPPGGDLAKVQRAVCMLSNNTAIAEAWARLDHKFDLMYAKRAFVHWYVVGEGMEEGEFSEREDMAALEKDY
E EVGVDSVEGE GEEGEEY
length Is 444
Number Of Polar Amino Acid Is 123
MSYQPQTEAATSRLNVEEAGKTLRIHFNDCCGQGETVVLHSGSGPGATGWANFSRNIDPVEAGYRVILLDCPGWGKSDSVVNSG
SRSDLNARILKSVDQLDIAKIHLLGNSMGGHSSAFTLKWPERVGGKLVLMGGGTGGMSLFTPMPTGEGIKRLNLQLYRQPTIENLKL
MMDIFVFDSDLTDLFAEARLNNMLSRRDLHENFVKSLEANPKQFPDFGPRLAIEIKAQTLIVWGRNDRVPM DAGLRLLSGIAGSEL
HHIFRDCGHWAQWEHADAFNLVLNFLARP
length Is 284
Number Of Polar Amino Acid Is 71
The New Output File Has Been Generated In The Current Working Directory.
Have A Great Day Ahead.
babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/9$

```

(After Running The Code)

sequence.txt - Notepad

File Edit Format View Help

tr|023532|O23532_ARATH Disease resistance RPP5 like protein OS=Arabidopsis thaliana OX=3702 GN=d14480c PE=4 SV=1

```
MAAASSCSRRYDVDPFSGVQVVRKTFSLNLEAFDRRSINTFMDHGIERSRTIAPELIS
ATREARISIVISKNYASTWCLDELVEIHNRLDWQGLVISVYVDVDPSEVRKQTGEFG
DVFKKTCEDEKQQRWQALVDITNAGEDLNGPSEAMVVKIANDVSNKLISPSNS
FGDFVGIEAHLEAMNSILCLESEARMVGWGPSPGSGIKSTIGKALYSOLFQCFHFHAFVP
HVYSMKSEWEEIFLSKILGDKIKGGKLGVEQMLNQKVLIVLDVDDVDFELKTLVGET
KWFPGSRRIIVITQDMLKAHDIDLVEYKFPSLDLAKMLCRSAFGENSPDDFKALA
FEVAVALGNLPLGLSVLGSGLKRRTEKEWEMMPFRNGLNGOIMKTLRVSYDRHLHQKQD
DMFLYIACLFNGFEVSYVNDLLEDNVGVTMLVEKSLIRITPGDIEHNNLEKLGIEIDR
AKSKGNPGKRRFLTFDFTLRKTVLGRFCTAFRSKELLPIDEKSFQGMRLNQCLSVTGD
YNDLPQSLVYLPPLRLLDWRCPKCLPYSPKADYLIQLTWMSKLEKLWIEGTVPLGSL
KRMNMHGRYLREISDLNARNLEELNLSECRSLVTLSSSIQWAIKLIYLDNRGCKLES
FPTHNLNLESLYGLLYYDNLNRFNPKMETSTTSPHGIEIRVENCIWNKNLPGDLAYLC
LVRCMPCEFRPNLRLVIRGNQMLEKLWEGVQSLASLVEHDMSECGNLTEIPDLKATN
LVNLYLSNCKSLVTVPTSTIGNLQKLVRLMEKCTGLEVLPTDNLSSLKMLDLSGCSSLR
TFPLISKIKWLYLENTAIEEVPCCIEFNSWLTVMYCKRLKNISPNIFRLTILKLVLD
FTECRGVNVAHSDASVEDHSSYIPLYENIEYTRHHLPSKLTNDVEFNFSCKRIKECGI
RLNMTSGTSEYINLPCVQIVADTSLTALNMLSLGQGEASSVSSYPSLEGESLVDMSI
ISEQQDEEIPILNLSGSN
```

>sp|P11150|LIPC_HUMAN Hepatic triacylglycerol lipase OS=Homo sapiens OX=9606 GN=LIPC PE=1 SV=3

```
MDTSPLCFSILLVLCIFIQSSALGQSLKEPFGRRQAQAVETNKTLMHEKTRFLFFGETNQ
GCQIRINHPDTLQECGFNSLLPLVMIHGHVSVDGLENNWIQWVAALKSQAPQVNVGLV
DMITLAHDHYTIAVRNTRLVGKEVAALLRWLEESVQLSRSHVHLIGYSLGAHVSFGAGSS
IGGTHKIGRITGLDAAGPLFEGSAPSNRLSPDDANFVDAIHTFTRHMGSLVGKQPIGH
YDFYPNGSSFPQGHFLELYRHAHQGFNAITQIKCSHERSVHLFIDSLHAGTQSMAY
PCGDMNSFSQGLCLCKKGRNCLGYHVQRPGRSKRLFLVTRAQSPFKVYHYQKIQF
INQETETPIQTTFMSLGTGKMKQKIPITLKGKIASNKYTSFLITLDVDIGELIMIKFKW
ENSAMWAWMDTVQTIIPWSTGPRHSGVLKTRIRKAGETQQRMTFCSENTDDLRLRPTQ
EKIFVKEIKSKTSKRKIR
```

>sp|P77044|MHPc_ECOLI 2-hydroxy-6-oxononadienedioate/2-hydroxy-6-oxononatrienedioate hydrolase OS=Escherichia coli (strain K12) OX=83333 GN=mhpc PE=1 SV=4

```
MSYQPQTEAATSRFLNVEEAGKTLRIHFNDCCGQDETIVLLHSGSGPATGWANFSRNIDP
LVEAGYRVLIDCPGWGKSDSVVNSGSRDLNARILKSVVDQLDIAKIHLLGNSMGHSS
VAFTLKWPERVGLKVLMMGGTGGMSLFTPMTEGKRLNQLYRQPTIENLKLMMDFVDF
TSDLTALFEARLNNMLSRRDHLNFKVLEANKQFPDFGRLAEIKAQTLIVWGRNDR
FVPMAGLRLLSGIAGSELHIFRDCGHWAEHDAFNQLVLNFLARP
```

>sp|Q63H02|EGFLA_HUMAN Pikachurin OS=Homo sapiens OX=9606 GN=EGFLAM PE=1 SV=2

```
MDLIRGVLLRLLLASSLPGGAVSLRAAIRKPKGVGPPLDIKLGALNCTAFSIQWKNPRH
PGSILGYTVFVSEVGADKSLQEQLHSLVPLSRDIPTEEVIGDLKPGTEYRVSIAAYSQA
KGKRLSSPRHVTTLSDQSLPPAAPQPHVIVVSDSEVALSWKPGASEGSAPIQYVSVEF
IRPDFDKWITSIHRIQMSVMKGLDPTNYQFAVRAMNSHGSPSRMPSDIIRTLCEPE
EAGSGRYGPRIYITDMGAGEDDEQFEDDLDISFEVVKPLPATKGGNKKFLVESKKMSIS
NPKTISRLIPPTSASLPVTTVAPQPIPIQRKGNGVAIMSRLFDMPCEDTLCSDSFCVN
DYTWGSGRCQCTLGKGGESCEDIVIQPQFFGHYSYVTFEPLKNSYQAFQITLFRFAEE
DGLLLYCGENEHGRDGFMSLAIRRSLLQFRNCGTGVAIIVSETIKILGGWHTVMILYRGG
LWGLLLQNNGTPTVGGSGQYKTFRTPLVLGGAPSAYWLVRATGTNRGFGQCVQSLAV
NGRRIDWRPPLGKALSGADVGECSSGICDEASCIHGCTAIKADSYICLPLGFKGRH
CEDAFTLTIPQFRESLSYAATPHLEPQHYLSFMEFEITFRPDSGDGVLLYSYDTGSKD
FLSINLAGGHVEFRFDCSGGTGLVRSDEPLTLGNMHELRSVSTAKNGILQVDQKQIVEGM
AEGGFTQIKCNTDIFIGVPPYNDVCKNSGVLKPFSGSIQKILNDRITHVKHDTSGVN
VENAAHPCVAPCAHGGSCRPKEGVDCCPLGFEGLHCQCEGNYCLNTIIEAIEIPQF
IGRSYLTYNPDILKRVSGSRNSVHFRKTTAKDGLLLWRGDSPHRPNDSFISLGLRDA
LVFSYNLGSGVASIMVNGSFNDGRWHRVAVRDGQSGKITVDDYGARTGKSPGMRLNI
NGALYVGGMKIALHTNRQYMRGLVGCSHSFTLSTDYHISLVEDAVDGKNINTCGAK
```

>sp|Q7IU36|TBA1A_HUMAN Tubulin alpha-1A chain OS=Homo sapiens OX=9606 GN=TUBA1A PE=1 SV=1

```
MRECI SIHVGQAGVQIGNACHLYCLEHGIQPDGQMPSDKTI GGGDDSFNTFFSETGAGK
HVPRAVFDLEPTVIDEVRTGTyrQLFHPEQLITGKEDAAANNYARGHYITGKEIIDLVLD
RRKLADQCTGLQGLFVHFSFGGGTGGFTSLMERLSVDYGGKSKLEFSIYPAPQVSTA
VVEPYNSILTHTTLEHSDCAFMVDNEAIYICRRNLDIERPTYTNLNLIGQIVSSITA
SLRFDGALNVLDTEFQTNLVYPRIHFPLATYAPVISAEKAYHEQLSVAEITNACFEPAI
QMVKCDPRHGKYNACCLLYRGDVPKDVNAATIAIKTKRTIQFVDCPTGFKVGINYQPP
TVVPGDGLAKVQRAVCMLNNTTAAEAARLHDHFDLMYAKRAFVHMYVVEGMEEGEFSE
AREDMAALEKDYEEVGVDSVEGEGEEGEEY
```

(Original File)

protein.txt - Notepad

File Edit Format View Help

```
MDLIRGVLLRLLLASSLPGGAVSLRAAIRKPKGVGPPLDIKLGALNCTA
LVRATGTNRGFQGCVCVQSLAVGRRIDMRPWPLGKALSGADVGECCSSGICDE
Length Is 1001
```

Number Of Polar Amino Acid Is 295

```
MAAASSSCSRRYDVFPSSFSGVDVRKTFLSNLLAEAFDRRSINTFMDHGIER
IDEKSFQGMRLNQCLSVTGDMDLPQSLVYLPKRLRLLDWDRCLPKCLPYS
Length Is 1021
```

Number Of Polar Amino Acid Is 289

```
MDTSPLCFSILLVLCIFIQSSALGQSLKPEPFGRRQAQAVETNKTLMHEKKT
```

Length Is 1039

Number Of Polar Amino Acid Is 163

```
MRECI SIHVGQAGVQIGNACWELCYCLEHGIQPDGQMPSDKTI GGGDDSFN
Length Is 444
```

Number Of Polar Amino Acid Is 123

```
MSYQPQTEAATSRFLNVEEAGKTLRIHFNDCCGQDETIVLLHSGSGPATG
Length Is 284
```

Number Of Polar Amino Acid Is 71

Activate Windows

Go to Settings to activate Windows.

Windows taskbar showing search bar, task view, and various application icons. The system tray shows the date and time as 10:12 on 09-11-2019.

(Generated File)

Experiment No. 10

Date: 09/10/2019

CGI Script For Biodata

Aim:

To Write CGI Script To Display Biodata Based On Details Entered In HTML Form

Algorithm:

Perl CGI Code:

```
#!/C:/xampp/perl/bin/perl.exe"
print "Content-type: text/html\n\n";
use CGI qw(:standard);
$fn = param("fname");
$ln = param("lname");
$age = param("age");
$phn = param("phone");
$city = param("city");
$state = param("state");
$cls = param("class");
$sub = param("sub");
$res = param("res");
$sex = param("gender");
print "Name: $fn $ln <br>";
print "Age: $age<br>";
print "Sex: $sex<br>";
print "Phone No: $phn<br>";
print "City: $city<br>";
```

```
print "State: $state <br>";
print "Class: $cls $sub<br>";
print "Qualification: $qual<br>";
```

HTML Code:

```
<html><head><title>Test Form</title></head><body>
<form action = "http://localhost/cgi-bin/10.pl" method =
"POST">
First Name: <input type = "text" name = "fname" ><br>
Last Name: <input type = "text" name = "lname" ><br>
Phone No: <input type = "text" name = "phone" ><br>
Age: <input type = "text" name = "age" ><br>
Gender: <input type = "radio" name = "gender" value=male>Male
<input type = "radio" name = "gender" value="female">Female<br>
City: <input type = "text" name = "city" ><br>
State: <select name="state">
<option value = "Andhra Pradesh">Andhra Pradesh</option>
<option value = "Arunachal Pradesh">Arunachal Pradesh</option>
<option value = "Assam">Assam</option>
<option value = "Bihar">Bihar</option>
<option value = "Chhattisgarh">Chhattisgarh</option>
<option value = "Delhi">Delhi</option>
<option value = "Odisha">Odisha</option>
<option value = "West Bengal">West Bengal</option>
```

```
</select><br>
```

```
Class: <input type = "text" name = "class" ><br>
```

```
Subject: <input type = "text" name = "sub" ><br>
```

```
Qualification: <select name="qual">
```

```
<option value = "BSc"> BSc</option>
```

```
<option value = "MSc"> MSc</option>
```

```
<option value = "PhD"> PhD</option>
```

```
</select>
```

```
<input type = "submit">
```

```
</form></body></html>
```

Result:

Test Form

file:///C:/xampp/htdocs/10.html

First Name: BABUL

Last Name: PRADHAN

Phone No: 7438889317

Age: 23

Gender: ☒ Male ☐ Female

City: ANGUL

State: Odisha

Class: Msc

Subject: BIOINFORMATICS

Qualification: MSc

Submit Query

localhost

localhost/cgi-bin/10.pl

Name: BABUL PRADHAN

Age: 23

Sex: male

Phone No: [7438889317](tel:7438889317)

City: ANGUL

State: Odisha

Class: Msc BIOINFORMATICS

Qualification: MSc

Experiment No. 11

Date: 16/10/2019

CGI For Getting And Displaying Protein Details

Aim:

To Get Protein Details Like ID, Name, Function, Organism, And Sequence Using Input Tags From User. And Output The Details And Length Of The Protein Having UniProt ID: Q96AT9.

Algorithm:

Perl CGI Code:

```
#!/C:/xampp/perl/bin/perl.exe"
print "Content-type: text/html\n\n";
use CGI qw(:standard);
$ID = param("ID");
$Name = param("Name");
$Function = param("Function");
$Organism = param("Organism");
$Sequence= param("Sequence");
chomp($Sequence);
@seq=split("",$Sequence);
$len=@seq;
print "ID: $ID <br>"
print "Name: $Name <br>";
print "Function: $Function<br>";
print "Organism: $Organism<br>";
print "Sequence: $Sequence <br>";
print "Length Of The Sequence Is : $len <br>";
```

HTML:

```
<html><head><title>Test Form</title></head>
```

```
<body>
```

```
<form action = "http://localhost/cgi-bin/11.pl" method =  
"POST">
```

```
ID: <input type = "text" name = "ID" >
```

```
Name: <input type = "text" name = "Name" >
```

```
Function: <input type = "text" name = "Function" >
```

```
Organism: <input type = "text" name = "Organism" >
```

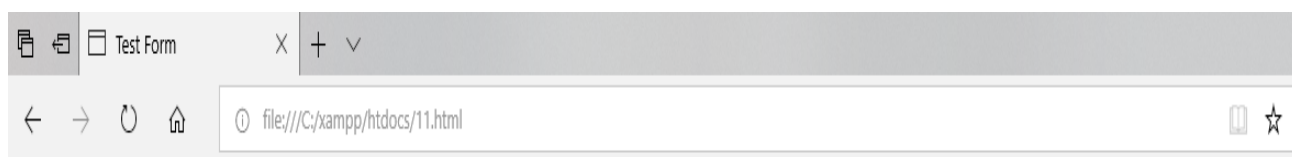
```
Sequence: <input type = "text" name = "Sequence" >
```

```
<input type = "submit">
```

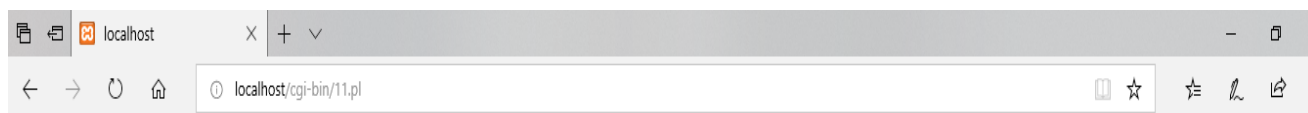
```
</form>
```

```
</body>
```

```
</html>
```

Result:

ID: Q96AT9 Name: phosphate 3-epimerase Function: bohydrate Degradation Organism: homo sapiens (Human) Sequence: MASGCKIGPSILNSDL Submit Query



ID: Q96AT9
Name: Ribulose-phosphate 3-epimerase
Function: Carbohydrate Degradation
Organism: Homo sapiens (Human)
Sequence: MASGCKIGPSILNSDLANLGAELRMLDSGADYLHLDVMDGHFVFNITFGHPVVESLRKQ
LGQDPFFDMHMMVSKPEQWVKPMVAVAGANQYTFHLEATENPGALIKDIRENGMKVGLAIK PGTSV EYLAPWANQIDMAL VMTVEPGFGGQKF MEDMMPKVHWLRTQFPSLDIEVDGGVGP
DTVHKAEAGANMIVSGAIMRSEDPRSVINLLRNV CSEAAQKRS LDR
Length Of The Sequence Is : 231

Experiment No. 12

Date: 19/10/2019

Calculating GRAVY For A Protein.

Aim: Write a Perl program to calculate Grand Average of Hydropathy (GRAVY) for the given protein Having UniProt ID: Q96AT9.

Algorithm:

```
#!/usr/bin/perl
print "Please Enter The File Name Or Path\n";
$seqfile = <STDIN>;

open (PROTEINFILE, $seqfile);

@protein = <PROTEINFILE>;

close PROTEINFILE;
chomp(@protein);
$len=@protein;
@protein1;
$i=0,$j=0;
for($i=0;$i<$len;$i++){
    if(@protein[$i]=~/>+/{
    }
    else{
        @protein1[$j]=@protein[$i];
        $j=$j+1;
    }
}
$protein1=join("",@protein1);
@protein2=split("", $protein1);
$len2=@protein2;
$A=0;
for($i=0;$i<$len2;$i++){
    if(@protein2[$i] eq "A"){ $A=$A+1.8;}
    if(@protein2[$i] eq "R"){ $A=$A+(-4.5);}
    if(@protein2[$i] eq "N"){ $A=$A+(-3.5);}
}
```



```

if(@protein2[$i] eq "D"){ $A=$A+(-3.5);}
if(@protein2[$i] eq "C"){ $A=$A+2.5;}
if(@protein2[$i] eq "Q"){ $A=$A+(-3.5);}
if(@protein2[$i] eq "E"){ $A=$A+(-3.5);}
if(@protein2[$i] eq "G"){ $A=$A+(-0.4);}
if(@protein2[$i] eq "H"){ $A=$A+(-3.2);}
if(@protein2[$i] eq "I"){ $A=$A+4.5;}
if(@protein2[$i] eq "L"){ $A=$A+3.8;}
if(@protein2[$i] eq "K"){ $A=$A+(-3.9);}
if(@protein2[$i] eq "M"){ $A=$A+1.9;}
if(@protein2[$i] eq "F"){ $A=$A+2.8;}
if(@protein2[$i] eq "P"){ $A=$A+(-1.6);}
if(@protein2[$i] eq "S"){ $A=$A+(-0.8);}
if(@protein2[$i] eq "T"){ $A=$A+(-0.7);}
if(@protein2[$i] eq "W"){ $A=$A+(-0.9);}
if(@protein2[$i] eq "Y"){ $A=$A+(-1.3);}
if(@protein2[$i] eq "V"){ $A=$A+4.2;}
}
$value=$A/$len2;
print "The Grand Average Of Hydropathy Value For The Given Protein Is
$value\n";

```

Result:

Select babul@DESKTOP-UBLVBR8: /mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/12

```

babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/12$ perl 12.pl
Please Enter The File Name Or Path
sequence.fasta
The Grand Average Of Hydropathy Value For The Given Protein Is -0.119047619047619
babul@DESKTOP-UBLVBR8:/mnt/c/Users/Babul Pradhan/Desktop/Lab Record Perl/12$

```

(Output)

Experiment No. 13

Date: 23/10/2019

Converting Format Using Bioperl

Aim: To Convert A FASTA File To GenBank Format Using Bioperl

Algorithm:

#!/usr/local/bin/perl

use Bio::Seq;

use Bio::SeqIO;

\$seq = "sequence.fasta";

\$in = Bio::SeqIO->newFh('-file' => "\$seq", '-format' => 'Fasta');

\$out = Bio::SeqIO->newFh('-format' => 'Genbank');

print \$out \$_ while <\$in>;

Result:

```

C:\WINDOWS\system32\cmd.exe
C:\Users\18378004\Desktop\Lab Record Perl>perl 13.pl
LOCUS       NC_000011.10:c5227071-5225464                1608 bp    dna        linear    UNK
DEFINITION  Homo sapiens chromosome 11, GRCh38.p13 Primary Assembly
ACCESSION   unknown
FEATURES             Location/Qualifiers
ORIGIN
  1  acatttgctt  ctgacacaac  tgtgttctact  agcaacctca  aacagacacc  atgggtgcatc
 61  tgactcctga  ggagaagtct  gccgttactg  ccctgtgggg  caaggtgaac  gtggatgaag
121  ttggttggtga  ggccctgggc  aggttggtat  caaggttaca  agacaggttt  aaggagacca
181  atagaaactg  ggcatgtgga  gacagagaag  actcttgggt  ttctgatagg  cactgactct
241  ctctgcctat  tggctctatt  tcccaccctt  aggctgctgg  tgggtctacc  ttggaccag
301  aggttctttg  agtcctttgg  ggatctgtcc  actcctgatg  ctggtatggg  caaccctaag
361  gtgaaggctc  atggcaagaa  agtgcctggg  gccttttagt  atggcctggc  tcacctggac
421  aacctcaagg  gcacctttgc  cacactgagt  gagctgcact  gtgacaagct  gcacgtggat
481  cctgagaact  tcagggtgag  tctatgggac  gcttgatgtt  ttctttcccc  ttcttttcta
541  tgggttaagtt  catgtcatag  gaaggggata  agtaacaggg  tacagtttag  aatgggaaac
601  agacgaatga  ttgcatcagt  gtggaaagtt  caggatcgtt  ttagtttctt  ttatttgcgt
661  ttcataacaa  ttgttttctt  ttgtttaact  ctgtctttct  ttttttttct  tctccgcaat
721  ttttactatt  ataactaatg  ccttaacatt  gtgtataaca  aaaggaaata  tctctgagat
781  acatttaagta  acttaaaaaa  aaactttaca  cagtctgcct  agtacattac  tatttggaat
841  atatgtgtgc  ttatttgcac  attcataatc  tccctacttt  attttctttt  atttttaatt
901  gatacataat  cattatacat  atttatgggt  taaagtgtaa  tgttttaata  tgtgtacaca
961  tattgaccaa  atcagggtaa  ttttgcattt  gtaattttaa  aaaatgcttt  cttcttttaa
1021  tatacttttt  tgtttatctt  atttctaata  ctttccctaa  tctcttttct  tcagggcaat
1081  aatgatacaa  tgtatcatgc  ctctttgcac  cattctaaag  aataacagtg  ataatttctg
1141  gggttaaggca  atagcaaatat  ctctgcatat  aaatatttct  gcatataaat  tgtaactgat
1201  gtaagagggt  tcatattgct  aatagcagct  acaatccagc  taccattctg  cttttatttt
1261  atgggttgga  taaggctgga  ttattctgag  tccaagctag  gcccttttgc  taatcatgtt
1321  catactctct  atcttctctc  cacagctcct  gggcaacgtg  ctggtctgtg  tgctggccca
1381  tcactttggc  aaagaattca  cccaccaggt  gcaggctgcc  tatcagaaag  tgggtggctgg
1441  tgtggctaata  gccctggccc  acaagtatca  ctaagctgcg  tttcttgctg  tccaattttc
1501  attaaagggt  cctttgttcc  ctaagtccaa  ctactaaact  gggggatatt  atgaaggggc
1561  ttgagcatct  ggattctgcc  taataaaaaa  catttatttt  cattgcaa
//
C:\Users\18378004\Desktop\Lab Record Perl>_

```

(Output)

Experiment No. 14

Date: 31/10/2019

Retrieving Sequence Using Bioperl**Aim:** To Retrieve A Sequence Using Accession Number Using Bioperl**Algorithm:**

#!/usr/local/bin/perl

use Bio::Seq;

use Bio::SeqIO;

print "Please Enter The Accession Number ";

\$accession=<STDIN>;

chomp(\$accession);

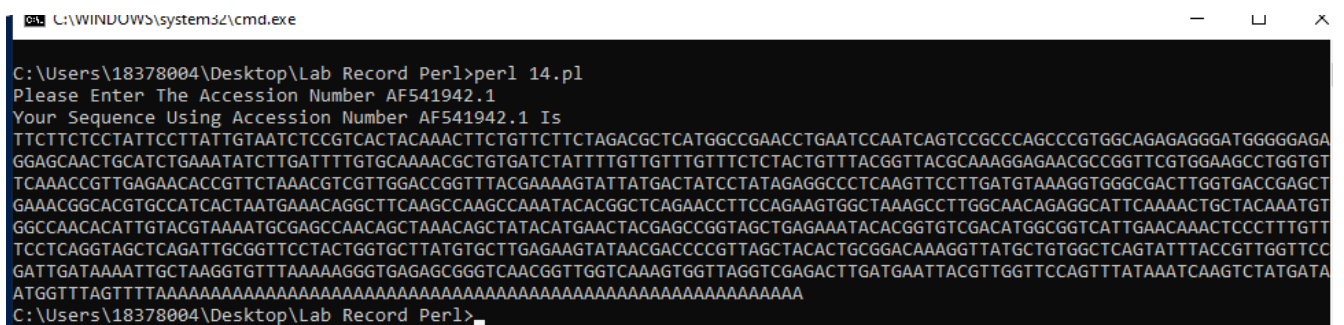
use Bio::DB::GenBank;

\$genBank = new Bio::DB::GenBank;

my \$seq = \$genBank->get_Seq_by_acc(\$accession);

my \$seqout = \$seq->seq();

print "Your Sequence Using Accession Number \$accession Is\n\$seqout";

Result:


```

C:\WINDOWS\system32\cmd.exe
C:\Users\18378004\Desktop\Lab Record Perl>perl 14.pl
Please Enter The Accession Number AF541942.1
Your Sequence Using Accession Number AF541942.1 Is
TTCTTCTCCTATTCTTATTGTAATCTCCGTCACTACAACTTCTGTTCTTAGACGCTCATGGCCGAACCTGAATCCAATCAGTCCGCCAGCCCGTGGCAGAGAGGGATGGGGGAGA
GGAGCAACTGCATCTGAAATATCTTGATTTTGTGCAAAACGCTGTGATCTATTTTGTGTTGTTTCTCTACTGTTTACGGTTACGCAAGGAGAAACGCCGGTTCGTGGAAGCCTGGTGT
TCAAACCGTTGAGAACACCGTTCTAAACGTCGTTGGACCGGTTACGAAAAGTATTATGACTATCCTATAGAGGCCCTCAAGTTCCTTGATGTAAGGTGGGCGACTTGGTGACCGAGCT
GAAACGGCACGTGCCATCACTAATGAAACAGGCTTCAAGCCAAGCCAAATACACGGCTCAGAACCTTCCAGAAGTGGCTAAAGCCTTGGCAACAGAGGCATTCAAAACTGCTACAAATGT
GGCCAACACATTGTACGTAATAATGCGAGCCAACAGCTAAACAGCTATACATGAACACTACGAGCCGGTAGCTGAGAAATACACGGTGTGACATGGCGGTCATTGAACAACTCCCTTTGTT
TCCTCAGGTAGCTCAGATTGCGGTTCTACTGGTGCTTATGTGCTTGAGAAGTATAACGACCCCGTTAGCTACACTGCGGACAAAGGTTATGCTGTGGCTCAGTATTTACCGTTGGTTCC
GATTGATAAAATTGCTAAGGTGTTTAAAAAGGGTGAGAGCGGGTCAACGGTTGGTCAAAGTGGTTAGGTCGAGACTTGATGAATTACGTTGGTCCAGTTTATAAATCAAGTCTATGATA
ATGGTTTATGTTTAAAAA
C:\Users\18378004\Desktop\Lab Record Perl>_

```

(Output)