**Motif Finding Problem Project**

**Gryffindor Team**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Team Members:**

1-Islam Hany Heggy.

2-Norhan Khalf Saad El\_din.

3- Menna Tullah Gamal ABD El\_Razik.

4-Menna Tullah Mohamed Mustafa.

5-Maha Mohamed Fathy.

Supervisor/ TA: Ahmed Mehany.

**Project Documentation**

**Brief Description:**

Given a DNA file which consists of a set of sequences t each sequence of length n, find a set of l-mers, one from each sequence that has the minimum number of mutations. The search patterns should come from the first sequence where the number of all possible search patterns of length l equals n-l+1.

t : number of sample DNA sequences.

n: length of each DNA sequence.

l : the length of pattern to find.

m : the maximum number of allowed mutations.

**Project Procedures:**

**1-Read File**

**2-Parse Variables**

**3-Get DNA**

**4-Find Minimum Sequence**

**5-Close File**

**6-Display**

**7-Write To File**

**8-Highlight**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Procedures Hierarchy**

**Main**

writeint

Get DNA

Parse Variables

Read File

Display

Find Minimum Sequence

Close File

closefile

openinputfile

Highlight

Write To File

readfromfile

WriteWindosMsg

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Procedures Functions and Flowcharts**

**1-Read File:**

This Procedure opens the file for reading and read it in a string

, it Receives nothing and Returns the file contents in filedata string which has been declared in the code.

**mov edx, offset file data**

**mov edx, offset file path**

**call read from file**

**call write window msg**

**jmp skip**

**je error**

**call read from file**

**call open input file**

**mov ecx, data range**

**if eax = invalid falue handle**

No Yes

**if eax = 0**

No Yes

**mov file handle ,al**

**call write window msg**

**je error**

**2-Parse Variables**

This Procedure parses the variables from filedata string which has been imported from the file into variables, it Receives the filedata string to work on it and Returns the parsed variables into these variables :

sampleno : refers to t \_\_ number of sample DNA sequences.

sequencelength : refers to n \_\_ length of each DNA sequence.

pattern : refers to l \_\_ the length of pattern to find.

mutations: refers to m \_\_ the maximum number of allowed mutations.

**if input equals**

**' = '**

**read int**

**depending on the counter we save the converted value**

**counter ++, convert to integer**

**3-Get DNA**

This Procedure Gets the DNA full sequence size and put the full sequence in DNA string it Recieves Sampleno variable which refers to t \_\_ number of sample DNA sequences and it Returns

DNA size which contains the size of the DNA, and DNA which contains the DNA full sequence itself.

**\*Getting DNA sequence size:**

**mov edx,0**

**mov eax,0**

**mov eax, sampleno**

**mul sequencelength**

**mov dnasize, eax**

**\* Putting DNA in DNA string:**

**add esi, 5**

**mov ebx , 0**

**mov ecx, dnasize**

**if ecx>0**

No Yes

**mov al, [esi]**

**mov dna[ebx], al**

**inc ebx**

**inc esi**

**4-Find Minimum Sequence**

This Procedure Finds the DNA sub sequence with minimum mutations.

it Recieves the Dna string and work on these variables (sequencelength, pattern, sampleno, mutations), and it Returns an Array which contains the indices of the best sequence found in prevoutput variable .

**Main function which find the minimum sequence**

if current motif = previous motif && mutations count >0

**No**

index ++ and continue check operation

mutation count ++

**5-Close File**

This ProcedureCloses the test case file it Receives filehandle variable and it Returns Nothing.

**movzx eax, filehandle**

**call closefile**

**6-Display**

This Procedure Saves the final output stored in prevoutput variable in a file and display all mutations on the console while highlighting the selected ones, it Receives the filehandle for the output file and Returns the File which contains the Output.

**take the output**

**Call Highlighting**

**Call write to file**

**Display the string with highlighted mutation**

**7-Write To File**

This Procedure Writes the final output to the file ,it Receives the savefilepath and it Returns the File which contains the answer .

**take array of indices**

Create new file to write the output in it, or over write on the existing file

**8-Highlight**

This Procedure Assigns the colors according to the mutations, it Receives the prevoutput variable which is used for highlighting the found motif.

and it Returns the Dna colors which is responsible for coloring motif and different mutations.

Highlight the mutations with different colors

same the mutation in a new array