課程名稱:最佳化演算法

作業名稱:尋找基因碼

授課老師:蕭瑛東

班級:資四甲

學號:110916015

姓名:林翡

日期:2024/3/2

* 題目:請以matlab語言撰寫一程式，提示使用者輸入一染色體序列，可以輸出該染色體的基因序列。若輸入的染色體序列找不到任何基因，則顯示「no gene is found」。
* 系統分析:
* 演算流程

開始

使用者輸入

染色體序列

head = strfind(chromosome,'ATG');

head不為空?

是

否

輸出no gene is found.

tail = {'TAA', 'TAG', 'TGA'};

genes = {};

for j = 1:length(head)

gene\_found = false;

for i = 1:length(tail)

stop = strfind(chromosome, tail{i});

stop = stop(stop > head(j));

gene\_three不為空且是3的倍數?

是

否

genes{end+1} = strjoin(gene\_three, '');

gene\_found = true;

if沒有找到基因序列

是

genes不為空?

否

是

輸出no gene is found.

stop不為空?

否

是

gene = chromosome(head(j) + 3:stop(1) - 1);

gene是否為3的倍數?

否

輸出no gene is found.

是

gene\_three = regexp(gene, '.{3}', 'match');

輸出no gene is found.

輸出no gene is found.

輸出no gene is found.

輸出no gene is found.

結束

* 虛擬碼:

輸入:染色體序列;

head = 找出染色體序列中有’ATG’的位置;

if head為空:

輸出 ’no gene is found.’;

return;

tail = {'TAA', 'TAG', 'TGA'};

genes = {};

for ( j from 1 to length (head)) do

gene\_found = false;

for ( i from 1 to length (tail)) do

stop = 找出有終止字串的位置;

stop = 保留stop大於head的位置;

if stop不為空:

gene = 提取從 head + 3 到 stop – 1位置的基因序列;

if gene的長度是三的倍數:

gene\_three = 以三個為一組的基因序列;

if gene\_three不為空且gene\_three的長度是三的倍數:

將gene\_three加到genes;

gene\_found = true;

else:

輸出 ’no gene is found.’;

if gene\_found為false:

輸出 ’no gene is found.’;

if genes不為空:

輸出genes;

else:

輸出 ’no gene is found.’;

* 設計流程:

Input:一染色體序列

Output:基因序列

Step1:在輸入的染色體序列中先找出有無起始字串ATG

Step2:檢查是否有找到ATG，並定義gene\_found標記為false以及genes用來存放基因序列。有找到ATG的話就繼續往下查找有無終止字串，且須確定終止字串必須在起始字串之後；沒有則輸出no gene is found。

Step3:檢查是否有找到起始字串與終止字串的染色體序列，有的話提取基因序列，沒有則輸出no gene is found。

Step4:檢查提取的基因序列數量是否是三的倍數，是三的倍數就把基因序列加到genes裡，並將gene\_found設為true；不是三的倍數則輸出no gene is found。

Step5:如果genes不為空就輸出基因序列，否則輸出no gene is found。

* 程式碼(註釋):

chromosome = input('輸入染色體序列:','s');

% 提示使用者輸入染色體序列，並存在變數chromosome中

head = strfind(chromosome, 'ATG');

% 查染色體序列中的起始字串'ATG'，並把結果存在head變數中

if ~isempty(head)

% 如果head變數不為空

tail = {'TAA', 'TAG', 'TGA'};

% 定義三個終止字串，存在tail這個cell array中

genes = {};

% 用來存找到的基因序列

for j = 1:length(head)

% 遍歷所有的起始字串位置

gene\_found = false;

% 用來標記是否找到了基因，預設值為false

for i = 1:length(tail)

% 遍歷所有的終止字串

stop = strfind(chromosome, tail{i});

% 找到染色體序列有終止字串的位置，並存在stop變數中

stop = stop(stop > head(j));

% 確保終止字串在起始字串之後

if ~isempty(stop)

% 如果stop不為空，表示有起始字串和終止字串組成的序列

gene = chromosome(head(j) + 3:stop(1) - 1);

% 提取起始字串的下一個位置與終止字串的前一個位置所包含的基因序列

if mod(length(gene), 3) == 0

% 檢查gene的長度是否為三的倍數

gene\_three = regexp(gene, '.{3}', 'match');

% 將gene以三個一組分組

if numel(gene\_three) ~= 0 && mod(length(gene\_three{1}), 3) == 0

% 如果基因序列不為空且是三的倍數

genes{end+1} = strjoin(gene\_three, '');

gene\_found = true;

break;

else

disp('no gene is found.');

break;

% 不符合條件，顯示沒找到基因

end

else

disp('no gene is found.');

break;

% 不符合條件，顯示沒找到基因

end

end

end

end

if ~gene\_found

% 如果沒有找到基因序列

disp('no gene is found.');

% 顯示沒找到基因

end

if ~isempty(genes)

% 如果找到了基因

disp('Output:');

disp(genes);

% 顯示基因序列

else

disp('no gene is found.');

end

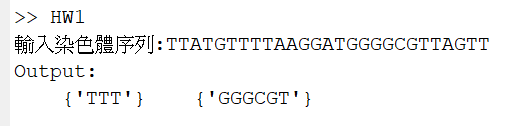
else

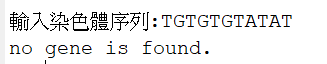
disp('no gene is found.');

% 沒有找到起始字串

end

* 結果:
* 輸出畫面





* 結果說明:

在輸入一段染色體序列後可以輸出正確的基因序列，並且長度都是三的倍數，如果沒有找到基因序列，則會輸出沒找到基因。

* 心得與感想

看到基因序列的題目讓我想起了高中讀三類組學生物的時光，由於大學後就沒接觸了所以格外陌生，matlab也算是我第一次使用，所以也比較不熟悉，很多東西都是上網查查找找來的，我體驗到程式語言其實也都大同小異，即使以前沒學過matlab，但也只是語法的差異而已，邏輯還是相通的。我覺得matlab有個好處是有蠻多函數都可以直接拿來使用，不需要另外寫，還蠻方便的。