DCC–IME–USP

CARLOS EDUARDO FERREIRA
SALA 297A TEL.: 818 6140
E-MAIL cef@ime.usp.br

JOSÉ COELHO DE PINA
SALA 164B TEL.: 818 6295
E-MAIL coelho@ime.usp.br

MAC-115 – Introdução à Computação

INSTITUTO DE QUÍMICA - PRIMEIRO SEMESTRE DE 2000

Exercício-Programa 3 Entrega: **15 de junho de 2000**

Decifrando o código genético

A mesma química professora do IQ-USP (na realidade, ela é biologista molecular e participou do seqüenciamento do genoma da bactéria *Xylella fastidiosa*) do Exercício-Programa 1 e 2, deseja um programa que determina se uma seqüência de genes possui uma proteína cuja ausência pode resultar em uma doença gravíssima.

Preocupada em ter certeza de obter resultados confiáveis, a professora resolveu ordenar que os alunos de MAC-115 façam um programa que recebe:

- a estrutura primária de uma proteína;
- um trecho de uma seqüência de genes (DNA);

e determina, caso exista,

- um trecho da seqüência de genes que contém uma ocorrência da tal proteína.

Proteínas, aminoácidos, nucleotídeos e outras quimeras

Considere duas classes de macromoléculas: proteínas e ácidos nucléicos.

Proteínas são polímeros responsáveis, entre outras, pela catálise das transformações químicas e pela realização de movimento. São formadas por subunidades chamadas de aminoácidos. Existem 20 aminoácidos naturais.

Ácidos nucléicos, por sua vez, são responsáveis pela hereditariedade. O DNA (ácido desoxirribonucléico) é constituído por duas cadeias em dupla hélice unidas através de pontes de hidrogênio. Estas cadeias são polímeros formados pela combinação de 4 subunidades de desoxirribonucleotídeos contendo as bases adenina (representada pela letra `'a'`), citosina (representada por `'c'`), guanina (representada por `'g'`) e timina (representada por `'t'`).

Na transcrição do DNA apenas um dos seus filamentos é usado para a síntese do RNA-mensageiro (RNA = ácido ribonucléioco). Durante esta síntese um setor de um filamento de DNA (gene) se afasta do seu complemento, expondo suas bases, onde se encaixam os nucleotídeos de RNA. Esse encaixe obedece à obrigatoriedade de ligação entre as bases. Porém, onde houver uma adenina no DNA, vai se encaixar a base uracil (representada pela letra `'u'`). Por exemplo, para uma seqüência `'tacggacta'` do DNA, haverá a seqüência `'augccugau'` no RNA. Assim o código genético que estava no DNA foi transcrito para o RNA.

Traduzir o código genético significa passar o código que está na forma de uma seqüência de bases no RNA, para uma seqüência de aminoácidos. Desse modo, o código genético contido no DNA é materializado na forma de um proteína estrutural ou enzimática, que por sua vez é responsável por uma característica do organismo.

O mecanismo da tradução é o seguinte. Cada grupo de três bases consecutivas do RNA-mensageiro corresponde a um **aminoácido**. Esses trios de bases, chamados de **códons**, já são conhecidos e podem ser consultados na tabela abaixo. Eles são os mesmos para todos os seres vivos e, por isso, dizemos que o código genético é universal. Observemos que há códons (`'tag', 'taa'` e `'tga'`) que não especificam nenhum aminoácido, mas indicam o fim de uma cadeia de aminoácidos. O códon `'atg'` serve tanto para codificar o aminoácido metionina como para determinar o início de uma nova cadeia.

Outro fato que chama a atenção é que o código genético é degenerado, ou, em outras palavras, há mais de um códon com o mesmo "significado". A alanina, por exemplo, pode ser escolhida por qualquer um dos seguintes códons: 'gct', 'gcc', 'gca' e 'gcg'.

Código	aminoácido	trios de bases (códons) no DNA
A	fenilanina	ttt, ttc
B	leucina	tta, ttg, ctt, ctc, cta, ctg
C	isoleucina	att, atc, ata
D	metionina	atg
E	valina	gtt, gtc, gta, gtg
F	serina	tct, tcc, tca, tcg, agt, agc
G	prolina	cct, ccc, cca, ccg
H	treonina	act, acc, aca, acg
I	alanina	gct, gcc, gca, gcg
J	tirosina	tat, tac
K	fim	taa, tag, tga
L	histidina	cat, cac
M	glutamina	caa, cag
N	asparagina	aat, aac
O	lisina	aaa, aag
P	ácido aspártico	gat, gac
Q	cisteína	tgt, tgc
R	triptófano	tgg
S	arginina	cgt, cgc, cga, cgg, aga, agg
T	glicina	ggt, ggc, gga, ggg
U	ácido glutámico	gaa, gag

O quê o seu programa deve fazer

Escreva um programa na linguagem C que, dadas:

- a estrutura primária de uma proteína; e
- uma seqüência genética;

e determina, caso exista,

- um trecho da seqüência genética que contém uma ocorrência da tal proteína.

Para este programa considere uma **seqüência genética** como sendo uma seqüência de caracteres em `'a'`, `'c'`, `'g'`, `'t'`. Existem 20 aminoácidos que serão referenciados com as letras maiúsculas, `'A'` a `'U'` (como mostrado na tabela acima). A estrutura primária de uma **proteína** será representada por uma seqüência de aminoácidos, ou seja, seqüência de letras maiúsculas entre 'A' e 'U'. Existem $4 \times 4 \times 4 = 64$ codificações possíveis para um aminoácido, sendo que cada codificação corresponde a no máximo um aminoácido; mas um aminoácido pode ter mais do que uma codificação (como vimos na tabela acima).

Através do exemplo a seguir, vamos ilustrar o método que deverá ser utilizado neste programa.

Considere a seguinte estrutura primária de uma proteína

```
M  S  I  M  K
```

Notemos que esta proteína pode corresponder à várias seqüências genéticas como, por exemplo,

```
M  S  I  M  K
caacgcgcccagtaa
caacgtgctcaataa
cagcgagctcaatgt
...
```

Considere agora a seqüência genética

```
attgctagcaatgctagcaattgctagcaattcat
```

Sabemos que a proteína pode ocorrer nessa seqüência, mas não sabemos onde, e nem conhecemos.

Inicialmente, tentaremos alinhar os primeiros caracteres:

```
attcaacgtaatcagcgtgcccaatagcaattcat
M  S  I  M  K
```

Nesta situação temos que, logo no início da seqüência, ao aminoácido `M` (glutamina) não corresponde o código `att` (que é uma codificação da isoleucina, `C`). Assim, as codificações genéticas obtidas por esse alinhamento não são válidas.

Agora, vamos tentar alinhar a proteína com o segundo caractere da seqüência genética:

```
attcaacgtaatcagcgtgcccaatagcaattcat
 M  S  I  M  K
```

Novamente, temos uma mesma `M` não corresponde ao código `ttc`. O mesmo acontece quando tentamos alinhar a proteína com o terceiro caractere da seqüência genética (verifique).

A seguir, vamos tentar alinhar a proteína com o quarto caractere da seqüência genética:

```
attcaacgtaatcagcgtgcccaatagcaattcat
   M  S  I  M  K
```

Nesta situação temos que ambos os `M` da estrutura da proteína estão alinhados com a codificação do aminoácido glutamina e `S` também está alinhado com bases que de fato corresponde a uma codificação genética da arginina (`cgt`). Entretanto, `I` e `K` não estão alinhados a codificação que correspondem a alanina (`aat`) e a fim (`cgt`), repectivamente.

Continuamos tentando os demais alinhamentos (verifique) até finalmente obter alinhamento que corresponde à proteína `M S I M K`.

```
attcaacgtaatcagcgtgcccaatagcaattcat
               M  S  I  M  K
```

Por este alinhamento temos que `cag` e `caa` são codificações genéticas do aminoácido `M` (glutamina), `cgt` é uma codificação para `S` (arginina), `gcc` para `I` (alanina), `tag` para `K` (fim).

Este é um exemplo artificial que utilizamos para descrever o método. Com dados reais, os comprimentos das duas seqüências são muuuuuito maiores.

Funções do seu programa

O seu programa deve conter e utilizar pelo menos as seguintes funções:

1. `int leia_seq_genetica (char dna[MAX_TAMANHO_DNA]);`

   Lê o nome de um arquivo e, caso o arquivo exista, lê os caracteres nele gravados, e armazena no vetor `dna` somente os caracteres que podem ocorrer numa seqüência genética, ou seja, `'a', 'c', 'g'` ou `'t'`. A função devolve o comprimento da seqüência e, caso o arquivo não exista, imprime uma mensagem e devolve 0.

2. `int leia_proteina (char proteina[MAX_TAMANHO_PROTEINA]);`

   Lê o nome de um arquivo e, caso o arquivo exista, lê os caracteres nele gravados, e armazena no vetor `proteina` somente os caracteres que podem ocorrer numa proteina, ou seja, as letras entre 'A' e 'U'. A função devolve o comprimento da seqüência e, caso o arquivo não exista, imprime uma mensagem e devolve 0.

3. `char codigo_aminoacido (char codon[3]);`

Devolve um caractere `ch` associado ao aminoácido que corresponde ao códon contido nas três primeiras posições do vetor `codon` (ou seja `codon[0] , codon[1]` e `codon[2]`), como mostrado na tabela acima. Assim, se, por exemplo, `codon[0] = 't', codon[1] = 'c', codon[2] = 't'`, então o caractere devolvido pela função deve ser `'F'`, correspondente a serina (verifique). Caso o vetor `codon` não contenha um trio de bases de um aminoácido então a função deve retornar o caractere (digamos) `'Z'`.

Observações

1. Para testar inicialmente o seu programa crie dois arquivos de entrada, `gene1.txt` e `prot1.txt`, contendo, respectivamente, a seqüência genética e a proteína, utilizados no exemplo acima.

2. Deixaremos na página da disciplina dois arquivos: `gene2.txt` e `prot2.txt`, que contêm dados genéticos reais. O primeiro contém a seqüência genética para um vírus humano, e o segundo contém a seqüência de aminoácidos para uma proteína "beta-lactamase". Teste o seu programa também para esses dados.

3. Entregue num envelope (preferencialmente de plástico transparente) :
   - Um disquete (que lhe será devolvido) contendo o programa em C. Não se esqueça de colocar uma etiqueta no disquete com o seu nome completo, turma e professor.
   - Uma listagem do programa.
   - As saídas impressas correspondentes às duas execuções do seu programa.