Eduardo Campos dos Santos

Uma introdução à Bioinformática através da análise de algumas ferramentas de *software* livre ou de código aberto utilizadas para o estudo de alinhamento de seqüências

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras e à FAEPE como requisito para obtenção do título de Especialista em Administração em Redes Linux

Orientador
Prof. MSc. Joaquim Quinteiro Uchôa

Lavras
Minas Gerais - Brasil
2004

Eduardo Campos dos Santos

Uma introdução à Bioinformática através da análise de algumas ferramentas de *software* livre ou de código aberto utilizadas para o estudo de alinhamento de seqüências

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras e à FAEPE como requisito para obtenção do título de Especialista em Administração em Redes Linux

Aprovada em *18 de setembro de 2004*

Prof. MSc Fernando Cortez Sica

Prof. DSc José Monserrat Neto

Prof. MSc. Joaquim Quinteiro Uchôa
(Orientador)

Lavras
Minas Gerais - Brasil

Sumário

Lista de Figuras

Lista de Tabelas

A todos aqueles que, de alguma forma, trabalham pelo bem social e pela liberdade na transmissão do conhecimento.

Agradecimentos

Aos meus pais, Marcos e Nely, pela formação e educação que me propiciaram.

À minha amada esposa, Rejanni. Que suportou meus momentos de nervosismo e ambivalência e apoiou-me nos momentos de angústia na minha árdua jornada até chegar à conclusão deste trabalho.

Ao Sr. Hugo Camargo Pádua: patrão, amigo e patrocinador. A conclusão deste curso não teria sido possível sem seu apoio e compreenção.

Ao meu orientador, por toda sua atenção e paciência. E também por suas valiosas dicas. Muito obrigado Joaquim.

Resumo

Neste trabalho, uma introdução à Bioinformática é desenvolvida através da análise de algumas das ferramentas de *software* mais usadas no estudo de alinhamento de seqüências. Os conceitos biológicos fundamentais são introduzidos, formando a base necessária para se compreender como agem alguns algoritmos e como se pode desenvolver outros que atendam mais diretamente às necessidades do pesquisador. A licença de algumas ferramentas são analisadas ilustram a diferença entre os conceitos (e suas implicações) de *software* livre e de código-aberto.

Capítulo 1

Introdução

Neste trabalho, apresenta-se o uso de ferramentas livres ou de código aberto em plataforma GNU/Linux no estudo do mais fundamental problema de Bioinformática: o estudo de alinhamento de seqüências. Neste capítulo, apresenta-se dados justificativos para uma abordagem sobre *software* livre e GNU/Linux em um estudo relacionado com Bioinformática.

A preferência por plataformas Unix e compatíveis e o uso das ferramentas de *software* livre na Bioinformática já são bastante consagrados entre os pesquisadores da área. Como aspectos motivadores pela escolha de uma plataforma Unix-compatível no estudo de tópicos em Bioinformática, [Gibas & Jambeck (2001)] apontam não apenas o alto grau de confiabilidade e desempenho dessa plataforma. Eles lembram também que é possível encontrar uma grande quantidade de ferramentas de software de excelente qualidade e popularidade no meio científico, compatíveis com o UNIX. Isso se deve ao fato do Unix ser usado extensivamente em universidades, onde é comum o desenvolvimento de software para análise de dados científicos. Nas palavras desses autores:

> *Os pesquisadores de Biologia Computacional e de Bioinformática têm ainda maior probabilidade de ter desenvolvido software para Unix, porque até meados da década de 90 as únicas estações de trabalho capazes de visualizar os dados de estruturas de proteínas em tempo real eram máquinas Silicon Graphics e Sun Unix.*

A opção por uma plataforma GNU/Linux torna-se então uma escolha imediata dado o tempo de amadurecimento e de conquistas dessa plataforma. Afinal, são

mais de dez anos de grande desenvolvimento e excelentes resultados. Tem-se uma confiabilidade comparável com a do Unix, mas disponível para máquinas mais modestas e a um custo imbatível, devido à gratuidade do GNU/Linux.

[Gibas & Jambeck (2001)] apresentam três motivos para utilizar e defender a plataforma GNU/Linux: custo-benefício - *com a disponibilidade do Linux, o Unix passa a ser barato*; melhor utilização dos recursos computacionais - *PCs antigos e considerados "obsoletos" por usuários do Windows tornam-se estações de trabalho surpreendentemente flexíveis e úteis*; grande número de ferramentas - *há uma rica biblioteca de ferramentas disponíveis para Biologia Computacional e para a pesquisa em geral*.

O conteúdo do texto foi distribuído da seguinte forma:

Capítulo 1: Introdução, em que se apresenta a justificativa para o desenvolvimento deste trabalho.

Capítulo 2: São introduzidos, metodicamente, os fundamentos sobre Biologia Celular e Biologia Molecular necessários para a discussão sobre as ferramentas analisadas nos capítulos seguintes.

Capítulo 3: Apresenta-se alguns bancos de dados públicos que contêm dados e artigos biológicos. Informações sobre formatos de arquivos aceitos por esses repositórios são citadas em caracter introdutório sobre o processo de disponibilização colaborativa de dados de pesquisas.

Capítulo 4: Analisa-se o BLAST (*Basic Local Alignment Tool*), um conjunto de ferramentas para alinhamento de seqüências. Quanto à forma de utilização analisou-se especialmente a ferramenta *blast2 sequence* usada na comparação de duas seqüências fornecidas pelo próprio usuário. Além disso, a estrutura de diretórios do arquivo compactado que contém os fontes do pacote foi analisada brevemente.

Capítulo 5: O uso do ClustalW é descrito detalhadamente através da execução de um alinhamento múltiplo de cinco seqüências. A ferramenta ClustalX que provê uma interface gráfica para a utilização do ClustalW é citada brevemente e utilizada para o alinhamento das mesmas cinco seqüências.

Capítulo 6: O TEXshade é apresentado como um poderoso recurso para a formatação final dos resultados obtidos com as ferramentas de alinhamento. É evidenciado sua capacidade em fornecer uma excelente qualidade gráfica além de uma grande flexibilidade ao pesquisador.

Conclusão: Os tópicos desenvolvidos são sintetizados e a mensagem final da pretensão deste trabalho é enfatizada.

Capítulo 2

Fundamentos de Biologia Celular e Biologia Molecular

2.1 DNA e RNA

As informações genéticas são armazenadas nos ácidos nucléicos - o ácido desoxirribonucléico (DNA) e o ácido ribonucléico (RNA). O DNA é encontrado principalmente no núcleo da célula. Mais especificamente nos cromossomos. O RNA, por sua vez, é encontrado principalmente no citoplasma, e em pouca escala também nos cromossomos.

A descoberta de que é na molécula de DNA onde se encontram as informações genéticas já serve como incentivo ao estudo do código genético. É importante ainda destacar que as informações contidas no DNA podem ser representadas em uma estrutura relativamente simples.

Os ácidos nucléicos são formados por uma ou duas cadeias (ou fitas) de elementos estruturais denominados nucleotídeos. Dessa forma, moléculas de DNA e RNA são classificadas como polímeros. Um polímero é uma molécula composta de pequenos elementos (os monômeros) que se repetem em sua estrutura. No caso de moléculas de DNA e RNA, os monômeros são os nucleotídeos.

A simplicidade da estrutura de moléculas de DNA e RNA se constitui pelo pequeno número de nucleotídeos distintos - são apenas quatro, seja para DNA ou para RNA.

> *Cada nucleotídeo é constituído por uma base nitrogenada, uma molécula de açúcar e um grupamento de fosfato. Há dois tipos de açúcar*

nos ácidos nucléicos: desoxiribose no DNA e ribose no RNA. As bases nitrogenadas são as primidinas: citosina (c), timina (t) e uracila (u) e as purinas: adenina (a) e guanina (g). O DNA contém a, c, g e t, enquanto o RNA contém u em vez de t. Em ambos DNA e RNA, os nucleotídeos estão ligados formando uma longa cadeia polinucleotídica. Essa cadeia é formada por ligações entre o grupo fosfato de carbono 5 de um nucleotídeo e o carbono 3 do açúcar do nucleotídeo adjacente [Oliveira].

As seqüências de nucleotídeos de moléculas de DNA e RNA podem ser representadas através de longas cadeias de letras. Essas letras estão contidas em um conjunto de quatro letras: a, c, g, t para moléculas de DNA e a, c, g, u para moléculas de RNA. Apesar da simplicidade no que se refere ao número de letras possíveis, as cadeias tendem a ser bastante complexas por serem extremamente longas. Mesmo para microorganismos a menssagem é longa, tipicamente 10^6 caracteres. [Lesk (2002)]

Na realidade, o DNA é composto por duas seqüências de aminoácidos entrelaçadas. Mas isso não representa um fator complicador à sua estrutura, uma vez que os nucleotídeos se ligam de maneiras específicas: a só pode fazer par com t, e g só pode fazer par com c. É exatamente essa característica que garante o sucesso da replicação.

Quando uma célula se divide para formar duas novas células-filhas, o DNA é replicado desenrolando as duas fitas e usando cada fita como um modelo para criar a sua imagem química espelhada, ou fita complementar [Gibas & Jambeck (2001)].

Moléculas de RNA, em geral, apresentam uma única fita de nucleotídeos que pode assumir uma grande variedade de conformações espaciais.

2.2 Genes, DNA genômico, cDNA, cromossomos e genoma

Genes são trechos de uma molécula de DNA que contêm as informações que determinam as características de uma espécie como um todo e de cada indivíduo em si. [Alberts et al. (1999)] *caput* [Oliveira e Inoue (2002)].

Existem três tipos de genes: os *genes codificadores de proteína*, que constituem-se em modelos para gerar moléculas de proteína; os genes especificadores de RNA e; os genes não transcritos, que são regiões do DNA genômico que possuem algum propósito funcional, mas não alcançam esse propósito, sendo transcritos ou convertidos para criar outra molécula. [Gibas & Jambeck (2001)]

O termo **DNA genômico** refere-se ao gene completo. Isso serve para diferenciar do chamado **DNA complementar** que refere-se ao gene sem as partes que não são codificantes - os **íntrons**. As partes codificantes são denominadas **éxons**. Essa divisão aplica-se somente aos organismos *eucariontes* (organismos cujas células possuem núcleos). Nos organismos *procariontes* (organismos cujas células não possuem núcleos), a região codificante se estende de forma ininterrupta.

Um **cromossomo** é uma molécula muito longa de DNA que contém muitos genes. E o conjunto completo dos cromossomos de uma célula é denominado **genoma**.

2.3 Proteínas

As proteínas são as moléculas responsáveis pela maior parte das estruturas e das atividades dos organismos. Outros elementos importantes nos organismos, que não são proteínas, são tratados por intermédio de enzimas, que, por sua vez, são proteínas. A importância das proteínas para os organismos é evidenciada pela própria origem etmológica da palavra: *o sueco Berzelius (1779-1848) criou o conceito proteína baseado na palavra de origem grega* ***proteios****, que significa primeiro, ou de principal importância* [Anônimo]. A importância do estudo de alinhamentos de seqüências protéicas é ressaltada por [Altschul et al. (1990)]:

> *Observa-se que genes ou proteínas com seqüências similares têm grande chance de possuírem funções similares. As primeiras informações para determinação da função de um gene, cuja seqüência foi recentemente obtida, quase sempre são obtidas pela busca de similaridades entre a nova seqüência e seqüências de proteínas ou famílias de proteínas conhecidas.*

Moléculas de DNA são, em primeira aproximação, uniformes. Proteínas, no entanto, mostram uma grande variedade de conformações tridimensionais. Isto é necessário para garantir a grande diversidade de suas características funcionais e

estruturais [Lesk (2002)]. É a estrutura tridimensional de uma proteína que define suas funções.

A seqüência dos aminoácidos de uma proteína dita sua estrutura tridimensional. O paradigma que se estabelece, portanto é:

- A seqüência do DNA determina a seqüência da proteína;
- A seqüência da proteína determina sua estrutura;
- A estrutura da proteína determina sua função.

Assim como o DNA e o RNA, as moléculas de proteínas também são polímeros. Mas no caso das proteínas, os elementos fundamentais - os aminoácidos, são mais diversificados em relação aos nucleotídeos. A Tabela 2.1 apresenta os nucleotídeos (que constiuem o DNA e o RNA) e os aminoácidos (que constituem as proteínas). Na tabela, os aminoácidos aparecem classificados como polares, apolares e eletricamente carregados. Outras classificações dos aminoácidos podem ser úteis. Por exemplo, pode-se classificar os aminoácidos conforme suas funções nos seres humanos, denotando quais são essenciais e quais são não-essenciais.

Tabela 2.1: Nucleotídeos e aminoácidos naturais

Os quatro nucleotídeos presentes em moléculas de DNA			
a adenina	c citosina	g guanina	t timina
Os quatro nucleotídeos presentes em moléculas de RNA			
a adenina	c citosina	g guanina	u uracila
Os vinte aminoácidos naturais presentes em moléculas de Proteínas			
Aminoácidos apolares			
G glicina	A alanina	P prolina	V valina
I isoleucina	L leucina	F fenilalanina	M metionina
Aminoácidos polares			
S serina	C cisteína	T treonina	N asparagina
Q glutamina	H histidina	Y tirosina	W triptofano
Aminoácidos elétricamente carregados			
D ácido aspártico	E ácido glutâmico	K lisina	R arginina

A seqüência dos aminoácidos em uma molécula de proteína constituem a chamada *estrutura primária* da proteína. É essa estrutura que define a forma e a função da proteína. As interações moleculares entre aminoácidos geram uma cadeia protéica denominada *estrutura secundária* e algumas vezes, uma *estrutura terciária* [Oliveira e Inoue (2002)]

A determinação das estruturas tridimensionais das proteínas permite "*realizar pesquisas mais direcionadas no sentido de encontrar inibidores, ativadores enzimáticos e outros ligantes que permitam a produção de fármacos mais eficientes e específicos: o almejado Desenvolvimento Racional de Fármocos (Rational Drug Design)*" [Prosdocimi et tal (2003)]. Uma infeliz realidade relacionada a isso é o caso do HIV. Como os vírus são organismos mais simples, é mais fácil encontrarmos mutações relevantes nesses organismos do que em outros organismos mais complexos, sobretudo em vírus que se reproduzem muito rapidamente.

Sobre isso, [Leme (2002)] afirma que *a rápida taxa de reprodução do HIV e sua inerente variabilidade genética conduziram à identificação de muitas variantes do vírus, que apresentam suscetibilidades diversas às drogas ARVs.* O HIV apresenta uma grande quantidade de variações e mesmo simples alterações produzem sensibilidades diferentes às drogas. [Leme (2002)] cita um exemplo: *na transcriptase reversa, uma mudança na posição 65, de AAA para AGA, provoca*

Tabela 2.2: Abreviatura dos aminoácidos naturais

G glicina - *glycine* (Gly)	A alanina - *alanine* (Ala)
P prolina - *proline* (Pro)	V valina - *valine* (Val)
I isoleucina - *isoleucine* (Iso)	L leucina - *leucine* - (Leu)
F fenilalanina - *phenylalanine* (Phe)	M metionina - *methionine* (Met)
S serina - *serine* (Ser)	C cisteína - *cysteine* (Cys)
T treonina - *threonine* (Thr)	N asparagina - *asparagine* (Asn)
Q glutamina - *glutamine* (Gln)	H histidina - *histidine* (His)
Y tirosina - *tyrosine* (Tyr)	W triptofano - *tryptophan* (Try)
D ácido aspártico - *aspartic acid* (Asp)	E ácido glutâmico - *glutamic acid* (Glu)
K lisina - *lysine* (Lys)	R arginina - *arginine* (Arg)

uma alteração na proteína produzida - lisina para arginina - ocasionando resistência à droga DDI.

É comum adotar a convenção de escrever nucleotídeos em letras minúsculas e aminoácidos em letras maiúsculas. Isso é bom para evitar confusões: por exemplo, nessa convenção, *atg* representaria a seqüência de nucleotídeos adenina-timina-guanina, enquanto que *ATG* representaria a seqüência de aminoácidos alanina-treonina-glicina. Entretanto, nem todos os autores seguem essa convenção, como no caso do exemplo anterior de [Leme (2002)].

Uma outra convenção que também é comumente utilizada: os nomes dos aminoácidos são freqüentemente abreviados usando as primeiras três letras do nome do aminoácido no idioma inglês com apenas a primeira letra maiúscula. Por exemplo, Gly para *glycine*. As exceções ocorrem para os aminoácidos: asparagina, glutamina e triptofano que são representados por Asn, Gln e Trp, respectivamente. O raro aminoácido selenocisteína é representado por Sec, na representação que usa três letras, e por U, na representação usa uma única letra. A Tabela 2.2 relaciona cada aminoácido com seus respectivos nome e abreviatura em inglês.

Portanto, uma seqüência de proteínas também pode ser representada por uma cadeia de caracteres. Nessa representação, cada aminoácido da seqüência é representado por uma letra ou por um conjunto formado por três letras, sendo apenas a primeira maiúscula. A representação dos aminoácidos por letras únicas é geralmente preferida por fornecer uma vizualização mais simples e por requerer menor dispêndio computacional.

Mesmo quando os nucleotídeos de um DNA são representados em letras maiúsculas, é facil perceber que uma dada seqüência refere-se a um DNA. Isso se deve pela simplicidade de sua estrutura no que se refere aos diferentes caracteres que

aparecem na representação da seqüência - cadeias que representam moléculas de DNA contêm apenas as letras A, C, G e T; cadeias que representam moléculas de RNA são constituídas usando-se apenas as letras A, C, G e U; por fim, representações de proteínas contêm uma maior variedade de letras. A Figura 2.1 ilustra exemplos reais de regiões de seqüências de uma molécula DNA e de uma molécula de proteína.

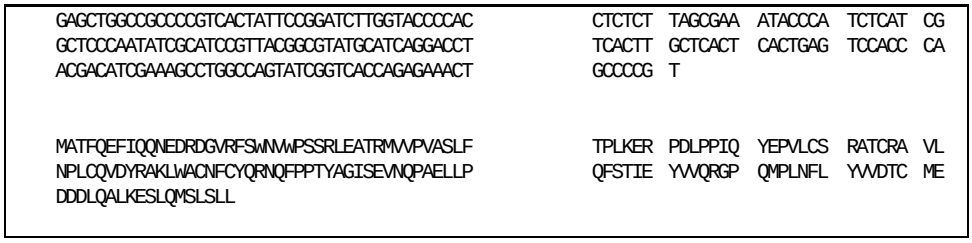

Figura 2.1: Exemplos de trechos de seqüências de DNA e proteína

2.4 O código genético

O assunto tratado nesta seção, disperta uma discussão sobre opniões divergentes. Alguns autores ainda escrevem e se fundamentam no chamado *código genético universal*. O caracter fundamental do conceito é tão marcante que torna-se difícil para alguns pesquisadores descartarem sua validade.

O código genético pode ser representado por uma tabela que permite rotular todas as possíveis tríades formadas com os quatro nucleotídeos presentes em moléculas de RNA.

O princípio do código genético afirma que, na síntese das proteínas, a seqüência de três nucleotídeos do RNA formam um determinado aminoácido. A cada tríade de nucleotídeos corresponde um dado aminoácido. Alguns aminoácidos podem ser constituídos pela combinação de diferentes tríades, mas cada tríade especifica um único aminoácido.

Opiniões de especialistas divergem quanto à universalidade dessa correspondência. [Oliveira] afirma que *essa correspondência é universal para todos os organismos vivos*. [Gibas & Jambeck (2001)] também defendem essa idéia e apresentam uma tabela da "correspondência universal".

Por sua vez, [Brown (2002)], apresenta uma seção que define exatamente o oposto, como se mostra evidente pelo próprio título da seção - *The genetic code is not universal*. O autor encerra a discussão sobre a não-universalidade do có-

digo genético com a afirmação de que o código dito universal, aplica-se sim a uma grande variedade de genes de uma grande variedade de organismos, mas que desvios são possíveis.

Uma base concreta sobre a contestação quanto à não-universalidade do código genético é apresentada no portal *Biologia na Web*[1]:

> *O fato de ser possível traduzir genes de um organismo em outro, p. ex., genes humanos, em E. coli, sugeria que o código padrão (..) era universal. Todavia, o estudo de diferentes seqüências de DNA a partir dos anos 80 revelaram algumas divergências em relação ao padrão.*
>
> *P. ex., em mitocôndrias de mamíferos o códon para a Met iniciadora pode ser AUG ou AUA (Ile no padrão); UGA especifica Trp e não terminação; AGA e AGG especificam terminação e não Arg. Nas mitocôndrias de plantas, fungos, Drosófila e protozoárias, também ocorrem variações em relação ao padrão. Nos protozoários ciliados, os códons UAA e UAG, ao invés de especificarem parada, codificam Gln. Além disto, foi relatado em Candida spp (Santos et al, 1997), eucariotos unicelulares, a existência de códons polissêmicos, isto é, um códon codificando mais de um aminoácido. No caso citado, CUG codifica tanto Leu como Ser, denotando ambigüidade e nos remetendo as seguintes questões: 1) em Candida, as alterções no Código Genético ainda não estariam completamente estabelecidas, ou 2) a ambiguidade CUG seria vantajosa, permitindo rápida adaptação a desafios ambientais, devendo ser mantida como tal.*
>
> *Estas são algumas das evidências de que o código genético padrão, se bem que amplamente utilizado, não é universal.*[2]

Hinegardner e Engelberg[3], desde 1963, também já se mostravam contrários a esse princípio simplista sobre a evolução das espécies.

A correspondência entre as tríades de nucleotídeos e seus respectivos aminoácido, compõe o que é chamado de **código genético**. Esse conceito talvez possa ser aplicado a organismos de uma mesma espécie.

[1] http://www.biologianaweb.com/

[2] http://www.biologianaweb.com/Livro2/C8/universal.html

[3] [hinegardner & Engelberg (1963)] e [hinegardner & Engelberg (1963)]

Tabela 2.3: Código genético - responsável pela síntese das proteínas

Base 1	**Base 2**				**Base 3**
	U	C	A	G	
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	STOP	STOP	A
	Leu	Ser	STOP	Trp	G
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Met[4]	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Glu	Gly	G

Cada tríade é também chamada de *códon* e cada seqüência de *códons* que codifica um polipeptídeo é denominada *cistron* [Oliveira].

O código genético "universal", que é o responsável pela síntese das proteínas, é apresentado na Tabela 2.3.

2.5 O *eyeless* e a aniridia - Um exemplo para justificativa do estudo de comparação entre seqüências

A mosca-das-frutas (*Drosophila melanogaster*) é muito estudada como modelo na pesquisa sobre a evolução de animais. Por isso, seus *genes* são bastante conhecidos. Ela tem um gene denominado *eyeless* que, se for retirado do genoma (por métodos de Biologia molecular), resulta em mocas-das-frutas sem olhos. É evidente, portanto, que o gene *eyeless* tem uma função importante no desenvolvimento do olho.

O ser humano apresenta um gene denominado *aniridia* que também parece

ter papel fundamental no desenvolvimento de olhos. Essa dedução começou a se formar a partir da observação, citada por [Gibas & Jambeck (2001)], de que *os seres humanos que não têm esse (ou em quem esse gene sofreu uma mutação suficiente para que o produto protéico parasse de funcionar corretamente), os olhos se desenvolvem sem íris.*

A mentalidade dedutiva dos cientistas os levaram a fazer o seguinte questionamento: "e se inserirmos o gene *aniridia* em uma *Drosophila* sem olhos (sem o *eyeless*)? Bom... o que acontece é que a *aniridia* promove a produção de olhos normais na *Drosophila*. Nas palavras de [Gibas & Jambeck (2001)]:

> *É uma coincindência interessante. Poderia haver alguma similaridade em como o "eyeless" e a "aniridia" funcionam, apesar de moscas e seres humanos serem organismos extremamente diferentes? Possivelmente. Para saber como o "eyeless" e a "aniridia" funcionam, juntos, é possível comparar suas seqüências. Entretanto, é preciso lembrar sempre que os genes interagem reciprocamente de maneira complexa. É preciso uma experimentação cuidadosa para obter uma resposta mais definitiva.*

2.6 Alinhamento de seqüências, similaridade, identidade e homologia

Uma vez representadas as seqüências de nucleotídeos ou de aminoácidos de duas moléculas, pode-se então compará-las em busca de similaridades em suas estruturas. Essa comparação permite inferir sobre as propriedades de uma determinada molécula baseando-se em propriedades conhecidas da outra. Ao processo de comparação entre seqüências, denomina-se *alinhamento de seqüências*. No alinhamento de duas seqüências, diferentes eventos são realizados sobre os monômeros de uma dada seqüência buscando-se obter uma maior similaridade entre as duas. A Figura 2.2 ilustra um alinhamento entre duas seqüências hipotéticas. Uma rápida inspeção visual, já indica a existência de certa similaridade entre as duas seqüências. Na primeira aproximação, realizou-se apenas algumas translações de regiões da seqüência. Na segunda aproximação, além das translações, efetuou-se também uma inversão das posições de dois caracteres. As translações foram indicadas com um traço "−" e os locais onde os caracteres não coincidiram foram marcados com um "X".

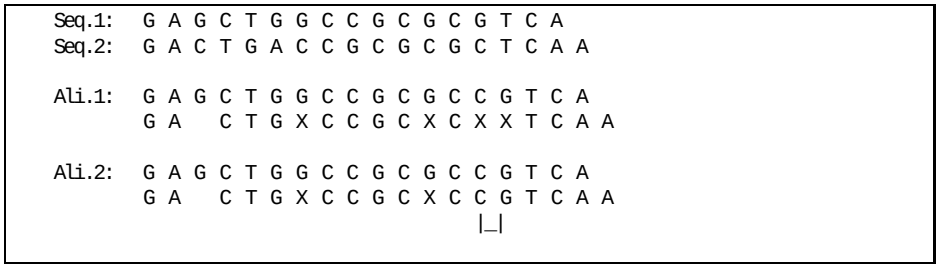

Figura 2.2: Um alinhamento hipotético

É importante destacar que a interpretação dos resultados de um dado alinhamento é fundamental para garantir interpretações coerentes com os fundamentos da Biologia. Boa parte da pesquisa em Bioinformática consiste em procurar obter algoritmos que sejam capazes de tratar as seqüências de caracteres de forma a fornecer resultados cada vez mais precisos biologicamente e reduzir cada vez mais a necessidade de interferência do pesquisador.

A comparação de seqüências permite inferir sobre possíveis mutações. Compara-se genomas de organismos de espécies distintas, supondo prováveis eventos que levaram à mutação de uma espécie para a outra. Dentre os possíveis eventos, pode-se citar a inversão de uma seqüência de genes ou a substituição de alguns genes. A esses eventos deve-se atribuir valores que representem suas probabilidades de ocorrência. Essas probabilidades são traduzidas através do conceito de **distância entre genes**.

Para cada tipo de evento existe a definição de uma "distância" entre os genes [Walter (1999)]. Assim, ao realizar uma inversão na ordem e na orientação dos genes numa determinada porção do genoma, o evento realizado é denominado **reversão** e é computado a **distância de reversão**. O evento denominado **transposição** consiste em mover uma porção de uma região para outra dentro do genoma e a distância relacionada chama-se **distância de transposição**. Quando se move os blocos de genes de um local para outro dentro do genoma, e se inverte a ordem e a orientação dos genes, diz-se que se realizou uma **transversão** e a distância é a chamada **distância de transversão**. Por fim, existe também a **translocação**, e a respectiva **distância de translocação**, que se referem à troca de porções entre dois cromossomos diferentes dentro do genoma.

De forma genérica, alinhar duas seqüências é encontrar uma corres-

pondência entre bases similares. Para o alinhamento são utilizadas mutações puntuais nos genes tais como substituições, remoções e inserções de bases. A distância é computada associando custos a estas operações, e procurando pela composição menos cara dentre as que transformam uma seqüência na outra [Walter (1999)].

A premissa de se buscar o menor custo possível no rearranjo é justificado pela **hipótese da parsimônia**. Neste princípio, assume-se que a Natureza, no processo evolutivo, sempre segue o caminho que exige o menor número possível de transformações. Assim, ao se tentar estudar as possíveis mutações, deve-se optar por uma série de eventos mínimos.

A **hipótese da parsimômia** pode ser contestada e tal contestação pode ser submetida à experimentação. Ainda que não concretize-se como uma Lei, sua suposição permite estabelecer uma linha para pesquisas filogenéticas.

A estrutura do DNA determina os mecanismos para a auto-replicação e para a translação dos genes em proteínas. Portanto, o estudo de alinhamento de seqüências, permite pesquisas variadas no campo da Biologia: pesquisas sobre a evolução de organismos; pesquisas voltadas para o combate de novos vírus a partir de outros já conhecidos; pesquisas voltadas para a obtenção de novos fármacos a partir de similaridades entre seqüências de diferentes proteínas; dentre outras.

Quanto à sua amplitude de aplicação sobre a seqüência, um alinhamento pode ser classificado como **alinhamento global** ou **alinhamento local**. Quando o alinhamento é realizado tomando-se toda a seqüência, ele é chamado de alinhamento global. Quando o alinhamento é realizado em fragmentos de uma seqüência, ele é chamado de alinhamento local. A escolha pelo tipo de alinhamento mais apropriado depende da finalidade desejada.

O alinhamento global é útil para comparar duas seqüências homólogas. Mas quando as duas seqüências apenas possuem certos domínios em comum, ou quando é necessário comparar uma seqüência com todas as entradas de uma base de dados, está-se mais interessado nos melhores alinhamentos locais entre duas subseqüências [Rocha].

Duas seqüências são homólogas, quando elas derivam de um mesmo ancestral [Prosdocimi et tal (2003)]. É importante destacar que homologia e similaridade

são dois conceitos distintos. Segundo [Prosdocimi et tal (2003)], *o alinhamento indica o grau de similaridade entre seqüências, já a homologia é uma hipótese de cunho evolutivo*.

Importante também é destacar que o alinhamento indica apenas o grau de similaridade entre as seqüências pesquisadas e que um mal alinhamento não implica em seqüências não-homólogas. Com efeito, [Pearson (2001)] compara a seqüência e a estrutura de três proteínas: *bovine chymotrypsin, S. griseus trypsin e S. griseus protease A*. As três proteínas apresentam uma estrutura tridimensional bastante similar. As duas primeiras apresentam grande similaridade em suas seqüências, enquanto que a terceira seqüência não apresenta uma similaridade significante. Assim, conclui [Pearson (2001)], proteínas homólogas não apresentam necessariamente seqüências com uma similaridade estatisticamente significante, ou mesmo detectável.

Quando mais que duas seqüências são alinhadas, o processo é chamado **alinhamento múltiplo**. Técnicas de alinhamento múltiplo são aplicadas principalmente a seqüências protéicas [Gibas & Jambeck (2001)].

Exemplo de programas que utilizam o alinhamento global são o ClustalW e o Multialin. O algoritmo do BLAST realiza o alinhamento local. O alinhamento global é usado geralmente para determinar regiões mais conservadas de seqüências homólogas. Já o alinhamento local é, geralmente utilizado na procura por seqüências homólogas ou análogas [Prosdocimi et tal (2003)].

Capítulo 3

Bancos de dados biológicos públicos

O armazenamento de informações e dados científicos, sobretudo quando em escala mundial, precisa ser cuidadosamente organizado de forma a evitar duplicidades e elevadas redundâncias. Deve-se ainda buscar padrões que possibilitem a concentração de informações sem que os próprios padrões tornem-se limitadores no processo do desenvolvimento científico. Assim, é importante que os padrões sejam cuidadosamente projetados de forma a permitir uma maior flexibilidade para se ajustar ao desenvolvimento futuro e é também importante, por vezes, abandonar um padrão substituindo-o por outro mais flexível e melhor dotado de recursos.

Em Bioinformática, existem diversos bancos de dados públicos de periódicos científicos e de resultados de pesquisas. Uma vez que a principal ferramenta de pesquisa em Biologia computacional é o próprio computador, a disponibilização de dados de pesquisas através do próprio meio computacional faz com que a utilização destes dados possa ser aproveitada com extrema facilidade.

Em geral, os dados disponíveis em repositórios públicos podem ser usados livremente para fins não-comerciais, como explicitado, por exemplo, pelo *Copyright* do *Swiss-Prot*[1].

> *This Swiss-Prot entry is copyright. It is produced through a collaboration between the Swiss Institute of Bioinformatics and the EMBL outstation - the European Bioinformatics Institute. There are no restrictions on its use by non-profit institutions as long as its content is in no*

[1] http://www.expasy.ch/sprot/

A principal instituição de compartilhamento de informações e dados biológicos é, sem sombra de dúvidas, o NCBI (*National Center for Biotechnology Information*), sendo citado em praticamente todos (senão em todos) os cursos e livros de Bioinformática. O NCBI está estabelecido nos Estados Unidos, existe desde 1988, criando bancos de dados, conduzindo pesquisas em Biologia Computacional, desenvolvendo ferramentas de software para análise de dados genômicos, e disseminando informações biomédicas [NCBI]. O NCBI[2] é uma divisão da Biblioteca Nacional de Medicina dos Estados Unidos (NLM - *National Library of Medicine*) no Instituto Nacional da Saúde (NIH - *National Institutes of Health*).

O *Entrez*[3] é um recurso do NCBI que procura centralizar consultas nos diversos repositórios do Centro de Informações. Ao submeter uma consulta, essa é realizada tanto nos bancos de dados de artigos científicos e livros *online* como também nos bancos de dados biológicos, como o banco de dados de nucleotídeos (GenBank) e o banco de dados de seqüências protéicas.

Dentre as publicações disponíveis, existe uma separação entre arquivos disponibilizados integral e gratuitamente e, outros, com apenas o *abstract* disponível livremente. Nesse último caso, pode-se obter o restante do conteúdo por *e-mail* mediante o pagamento de alguma taxa.

A Figura 3.1 mostra o resultado da consulta por "Drosophila eyeless aniridia". A quantidade de registros para cada tipo de dado é informada ao lado do respectivo item. Se mais de uma palavra é passada, ocorre uma busca pela ocorrência de todos os termos.

O item *PubMed* do *Entrez* traz apenas citações e resumos (*abstracts*). Mas o item *PubMed Central* traz artigos completos disponíveis gratuitamente. Os artigos são apresentados em uma formatação padrão com ilustrações de excelente resolução. As ilustrações podem ser visualizadas em versões maiores. O leitor pode ainda optar pela visualização da imagem na mesma janela ou em outra janela. Além disso, cada artigo contém *links* para outros artigos citados, bem como para outros artigos que o citaram, facilitando bastante o processo de pesquisa bibliográfica, que geralmente antecede as pesquisas em um novo projeto científico.

O item *Genome* contém as seqüências de genomas completos relativos à pesquisa efetuada. Este item leva ao *NCBI MapViewer*, que mostra, em forma pictó-

[2] http://www.ncbi.nlm.nih.gov/)

[3] http://www.ncbi.nlm.nih.gov/Entrez/index.html

Figura 3.1: Consulta por "Drosophila eyeless aniridia" no *Entrez*

rica, um mapa genético completo, destacando regiões relativa aos seus genes. A partir desse mapa, é possível ampliar determinada região do mapa, ou selecionar um gene específico.

A Figura 3.2 mostra o mapa genômico da Drosófila. Ao clicar no *link* referente ao gene *ey* obteve-se as informações específicas sobre esse gene conforme ilustrado na Figura 3.3. O pesquisador pode ainda visualizar ou mesmo efetuar um *download* da seqüência que desejar, clicando em *Download View Sequence Evidence*. No *download* ou na visualização, pode-se optar pelo formato FASTA ou pelo formato GenBank.

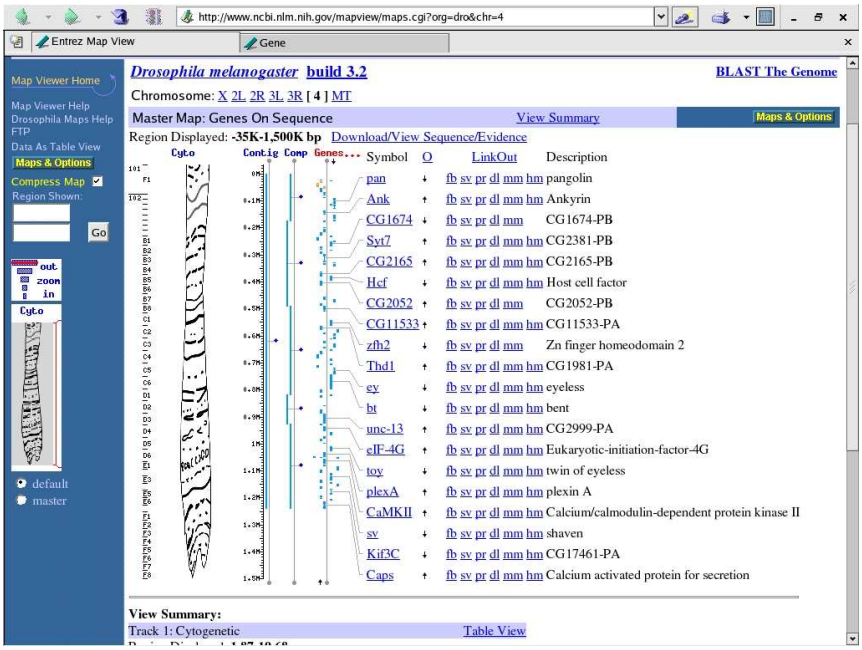

Figura 3.2: Mapa genômico da Drosófila no *MapViewer* do NCBI

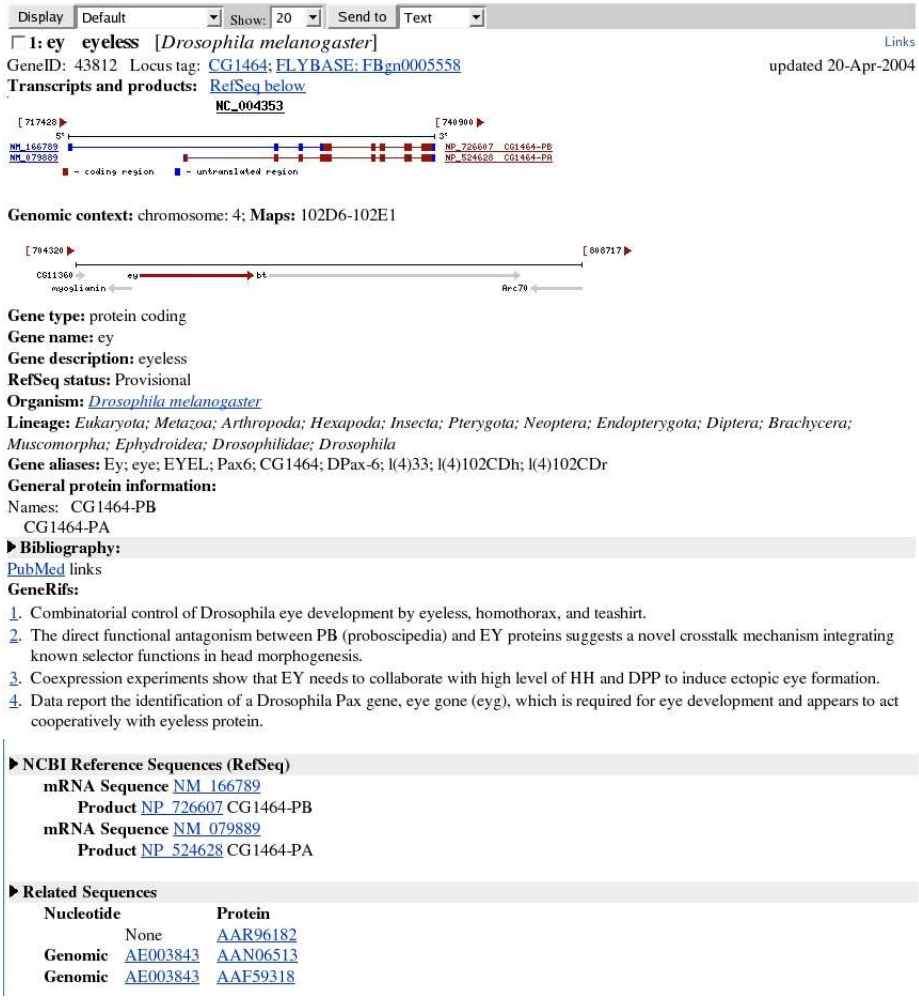

Figura 3.3: Informações sobre o gene *ey*

Uma seqüência no formato FASTA inicia com uma linha de comentário seguida da seqüência em si nas linhas subsequentes. A linha de comentário é iniciada com o caracter “>”. Logo após o caracter marcador de comentário “>”, é comum encontrar-se “gi” referente a *GenBank Identifier*. O NCBI recomenda que as linhas da seqüência tenham no máximo 80 caracteres[4]. O conhecimento dessas recomendações é importante ao submeter uma nova seqüência ao NCBI ou a outro repositório público que aceite seqüências no formato FASTA.

As seqüências submetidas ao NCBI devem estar representadas no padrão IUB/IUPAC para aminoácidos e nucleotídeos, com as seguintes exceções: letras minúsculas são aceitas e são convertidas para maiúsculas; um hífen ou travessão pode ser usado para representar uma lacuna (*gap*) de comprimento indeterminado.

Pode-se ainda usar a letra N para representar um nucleotídeo residual desconhecido.

A Figura 3.4 apresenta todos os resíduos permitidos em seqüências de nucleotídeos.

```
A --> adenosine              M --> A C (amino)
C --> cytidine               S --> G C (strong)
G --> guanine                W --> A T (weak)
T --> thymidine              B --> G T C
U --> uridine                D --> G A T
R --> G A (purine)           H --> A C T
Y --> T C (pyrimidine)       V --> G C A
K --> G T (keto)             N --> A G C T (any)
                             -  gap of indeterminate     length
```

Figura 3.4: Resíduos aceitos pelo NCBI em seqüências de nucleotídeos no formato FASTA.

Em seqüências de aminoácidos, U e * são aceitáveis e a letra X pode ser usada para representar resíduos de aminoácidos desconhecidos. Os resíduos aceitos em arquivos no formato FASTA no NCBI pelos programas que tratam seqüências de aminoácidos (BLASTP, BLASTX e TBLASTN)[5] estão apresentados na Figura 3.5.

[4] http://www.ncbi.nlm.nih.gov/BLAST/fasta.shtml

[5] Os programas que compõem o pacote BLAST estão relacionados em um capítulo próprio.

```
A  alanine                          P  proline
B  aspartate   or  asparagine       Q  glutamine
C  cystine                         R  arginine
D  aspartate                       S  serine
E  glutamate                       T  threonine
F  phenylalanine                   U  selenocysteine
G  glycine                         V  valine
H  histidine                       W  tryptophan
I  isoleucine                      Y  tyrosine
K  lysine                          Z  glutamate   or  glutamine
L  leucine                         X  any
M  methionine                      *  translation   stop
N  asparagine                      -  gap  of  indeterminate    length
```

Figura 3.5: Resíduos aceitos pelo NCBI em seqüências de aminoácidos no formato FASTA.

A Figura 3.6 mostra a seqüência do gene *ey* da drosófila no formato FASTA. As Figuras 3.7 e 3.8 mostram a anotação da mesma seqüência no formato GenBank. O formato GenBank traz mais informações além da seqüência em si, que aparece no final. Dentre as variadas informações, tem-se, por exemplo, diversos artigos relacionados, indicando-se autores, local de publicação (ou *Unpublished* quando ainda não publicado), comentários sobre o artigo, quando houver etc. O conhecimento do formato GenBank é importante não apenas para a análise de um arquivo nesse formato. É importante também para se construir algoritmos que extraiam e comparem determinadas informações em vários arquivos, automatizando e agilizando uma tarefa rotineira que seria muito desgastante, caso executada manualmente.

```
>ref|NC_004353.1|:734034-734222          Drosophila    melanogaster    chromosome    4,
 complete    sequence
GAGCTGGCCGCCCCGTCACTATTCCGGATCTTGGTACCCCAC          CTCTCT  TAGCGAA  ATACCCA  TCTCAT  CG
GCTCCCAATATCGCATCCGTTACGGCGTATGCATCAGGACCT          TCACTT  GCTCACT  CACTGAG  TCCACC  CA
ACGACATCGAAAGCCTGGCCAGTATCGGTCACCAGAGAAACT          GCCCCG  T
```

Figura 3.6: Seqüência do gene *ey* da drosófila no formato FASTA. A primeira linha foi truncada na adaptação para a impressão.

```
LOCUS        NC_004353          189 bp     DNA       linear     INV  19-APR-2004
DEFINITION   Drosophila   melanogaster   chromosome   4,  complete   sequence.
ACCESSION    NC_004353   REGION:   734034..734222
VERSION      NC_004353.1     GI:24638835
KEYWORDS     .
SOURCE       Drosophila   melanogaster   (fruit   fly)
  ORGANISM   Drosophila   melanogaster
             Eukaryota;   Metazoa;   Arthropoda;   Hexapoda;   Insecta;
             Pterygota;   Neoptera;   Endopterygota;    Diptera;   Brachycera;
             Muscomorpha;
             Ephydroidea;    Drosophilidae;    Drosophila.
REFERENCE    1  (bases  1 to  189)
  AUTHORS    Kaminker,J.S.,    Bergman,C.M.,    Kronmiller,B.,    Carlson,J.,
             Svirskas,R.,    Patel,S.,   Frise,E.,   Wheeler,D.A.,    Lewis,S.E.,
             Rubin,G.M.,   Ashburner,M.   and  Celniker,S.E.
  TITLE      The transposable   elements   of the  Drosophila   melanogaster
             euchromatin:   a genomics   perspective
  JOURNAL    Genome  Biol.  3 (12),  RESEARCH0084   (2002)
  MEDLINE    22426070
   PUBMED    12537573
    .....................................          .......  .......  ....
    .....................................          .......  .......  ....
REFERENCE    6  (bases  1 to  189)
  AUTHORS    .
  CONSRTM    FlyBase
  TITLE      Direct  Submission
  JOURNAL    Submitted   (06-SEP-2002)   University   of California   Berkeley,
             539  Life
             Sciences   Addition,   Berkeley,   CA  94720,   USA
REFERENCE    7  (bases  1 to  189)
  AUTHORS    Adams,M.D.,   Celniker,S.E.,   Gibbs,R.A.,   Rubin,G.M.   and
             Venter,C.J.
  TITLE      Direct  Submission
  JOURNAL    Submitted   (21-MAR-2000)   Celera  Genomics,   45  West  Gude
             Drive,   Rockville,   MD  20850,   USA
COMMENT      PROVISIONAL   REFSEQ:   This  record  has  not  yet  been  subject
             to  final
             NCBI  review.   The  reference   sequence   was  derived   from
             AE014135.
             COMPLETENESS:    full  length.
FEATURES             Location/Qualifiers
     source          1..189
                     /organism="Drosophila      melanogaster"
                     /mol_type="genomic      DNA"
                     /db_xref="taxon:7227"
                     /chromosome="4"
```

Figura 3.7: Parte da seqüência do gene *ey* da drosófila no formato GenBank. Modificada na adaptação para a impressão.

```
     gene            <1..>189
                     /gene="ey"
                     /locus_tag="CG1464"
                     /note="eyeless;      synonyms:    Ey,  eye,  EYEL,  Pax6,  CG1464,
                     DPax-6,   l(4)33,   l(4)102CDh,    l(4)102CDr"
                     /map="102D6-102E1"
                     /db_xref="FLYBASE:FBgn0005558"
                     /db_xref="GeneID:43812"
     mRNA            1..>189
                     /gene="ey"
                     /locus_tag="CG1464"
                     /product="CG1464-RB"
                     /transcript_id="NM_166789.1"
                     /db_xref="GI:24638703"
                     /db_xref="FLYBASE:FBgn0005558"
                     /db_xref="GeneID:43812"
     CDS             1..>189
                     /gene="ey"
                     /locus_tag="CG1464"
                     /codon_start=1
                     /protein_id="NP_524628.2"
                     /db_xref="GI:24638702"
                     /db_xref="FLYBASE:FBgn0005558"
                     /db_xref="GeneID:43812"
     CDS             1..>189
                     /gene="ey"
                     /locus_tag="CG1464"
                     /codon_start=1
                     /protein_id="NP_726607.1"
                     /db_xref="GI:24638704"
                     /db_xref="FLYBASE:FBgn0005558"
                     /db_xref="GeneID:43812"
ORIGIN
     1 gagctggccg    ccccgtcact    attccggatc    ttggtacccc    acctctctta    gcgaaatacc
    61 catctcatcg    gctcccaata    tcgcatccgt    tacggcgtat    gcatcaggac    cttcacttgc
   121 tcactcactg    agtccaccca    acgacatcga    aagcctggcc    agtatcggtc    accagagaaa
   181 ctgccccgt
//
```

Figura 3.8: Continuação da seqüência do gene *ey* da drosófila no formato GenBank. Modificada na adaptação para a impressão.

Capítulo 4

BLAST - *Basic Local Alignment Tool*

O volume de dados contidos nos repositórios públicos é enorme e continua crescendo. É impressindível, portanto, que haja alguma ferramenta que facilite o processo de comparação de uma nova seqüência com as seqüências já conhecidas.

Dentre as ferramentas existentes destaca-se o BLAST (*Basic Local Alignment Tool*), que *é a ferramenta mais popular de comparação de seqüências de DNA com os bancos de dados genômicos* [Santos & Queiroga (2003)].

Por ser uma ferramenta livre para uso não-comercial, pode-se encontrar diferentes implementações do BLAST. A mais conhecida é a NCBI-BLAST do *National Center for Biotechnology Information*. Outra muito conhecida é a WU-BLAST[1] da Universidade de Washington [Higa (2001)]. Uma comparação entre os parâmetros das versões WU e NCBI do BLAST pode ser vista no *site* do WU-BLAST[2].

Um centro de pesquisa, ou mesmo algum pesquisador, pode optar por implementar localmente o BLAST. Mas isso não é uma prática comum. Neste capítulo, a implementação o NCBI-BLAST foi escolhida. A utilização do BLAST é ilustrada em um alinhamento dos genes *eyeless* e *aniridia*. Além disso, parte da estrutura de diretórios e de alguns códigos-fontes é analisada.

O BLAST é constituído na verdade de uma série de programas. Segundo [Higa (2001)], são eles:

- **blastp**, para comparação de seqüências de aminoácidos em bancos de dados de proteínas;

[1] http://blast.wustl.edu/blast/

[2] http://blast.wustl.edu/blast/cparms.html

- **blastn**, para comparação de seqüências de nucleotídeos em bancos de dados de DNA;
- **blastx**, para comparação de uma seqüência de nucletotídeo transladada em todos os ORFs (*Open Reading Frames*) com bancos de dados de proteínas;
- **tblastn**, para comparação de seqüência de proteína com um banco de dados de seqüências de nucleotídeos dinamicamente transladados em todos os seus ORFs e;
- **tblastx**, para comparar os ORFs de uma seqüência de nucleotídeos com os ORFs de todos os nucleotídeos em um banco de dados de nucleotídeos.

O pacote dos códigos-fontes contém também alguns arquivos que servem apenas para fornecer uma interface mais amigável ao pesquisador. O sub-diretório `wwwblast` traz, por exemplo, rotinas CGI (*Common Gateway Interface*) e arquivos HTML para prover a o acesso ao BLAST via *browser*.

Como exemplo de utilização do BLAST, simulou-se aqui uma pesquisa de comparação entre os genes *eyeless* e *aniridia* em busca de similaridades significantes.

Para comparar duas seqüências específicas com o BLAST no NCBI, deve-se utilizar a interface própria para comparação de duas seqüências[3]. A Figura 4.1 mostra a página do *Blast 2 sequences* do NCBI. Na figura, já se vê as duas seqüências lançadas pelo pesquisador - foram utilizadas as seqüências dos genes: *eyeless* e *aniridia*.

A Figura 4.2 mostra uma página equivalente disponibilizada no Swiss-Prot[4]. Trata-se de uma implementação escrita em Perl. As seqüências do *eyeless* e do *aniridia* também foram alinhadas através do Swiss-Prot para efeitos de comparação do comportamento das duas ferramentas. Foi necessário retirar as linhas de comentário para inserir as seqüências como seqüências fornecidas pelo usuário.

Foram utilizados os mesmo parâmetros nos dois portais (NCBI e Swiss-Prot). Os resultados foram bastante similares. A Figura 4.3 mostra o resultado obtido no NCBI e a Figura 4.4 exibe o resultado obtido com pelo Swiss-Prot. No NCBI, as similaridades são demarcadas através de uma linha entre as duas linhas referentes às duas seqüências submetidas para o alinhamento. Para cada similaridade encontrada ocorre a impressão da letra referente ao monômero. Nos resultados fornecidos através do Swiss-Prot, as similaridades são apontadas através de uma terceira linha, abaixo das duas seqüências alinhadas, onde aparece um * (asterisco) para cada similaridade encontrada.

[3] http://www.ncbi.nlm.nih.gov/blast/bl2seq/bl2.html

[4] http://us.expasy.org/tools/sim-prot.html

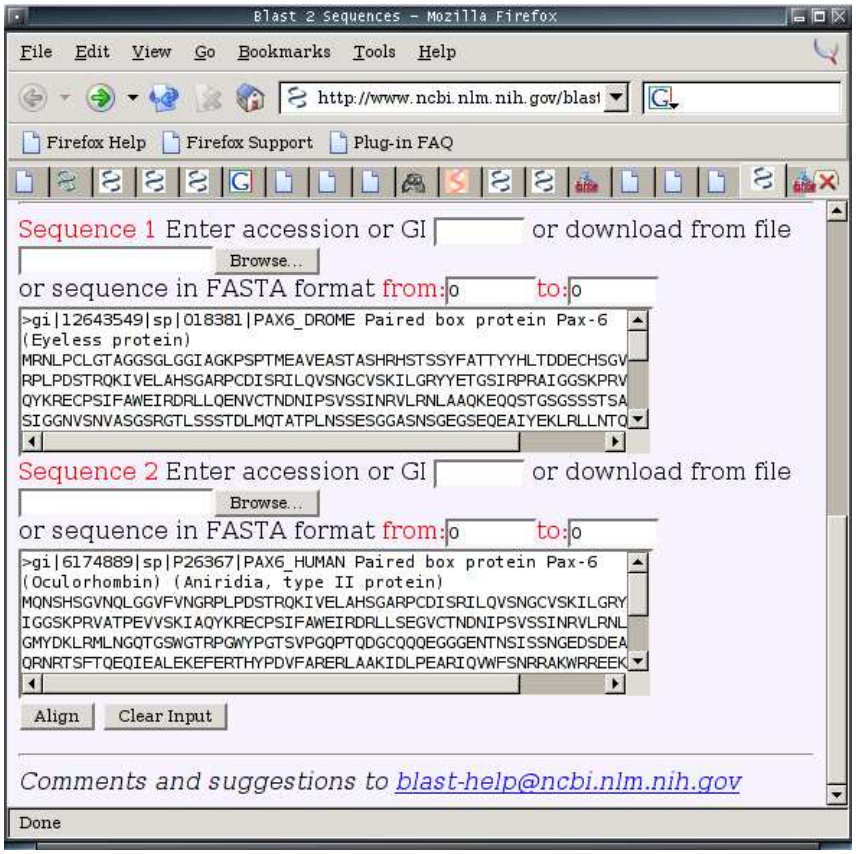

Figura 4.1: Comparando as seqüências do gene *eyeless* com o gene *aniridia* no BLAST através do portal do NCBI.

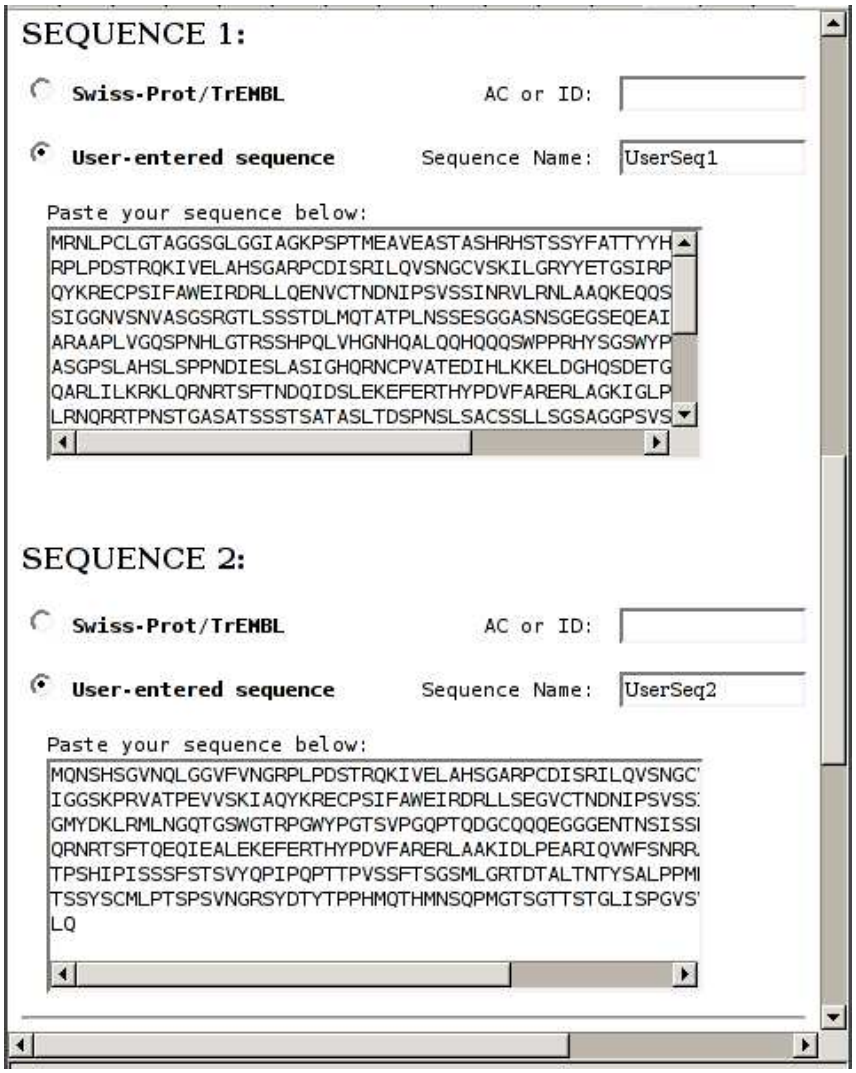

Figura 4.2: Comparando as seqüências do gene *eyeless* com o gene *aniridia* no BLAST através do portal do Swiss-Prot.

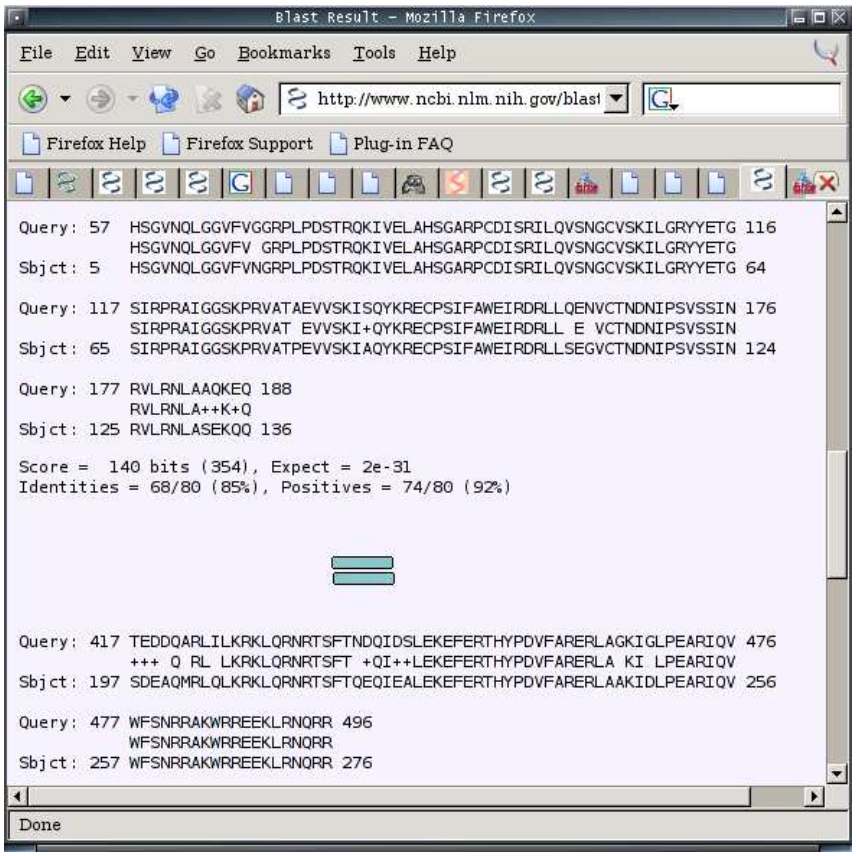

Figura 4.3: Resultado da comparação entre o gene *eyeless* com o gene *aniridia*.

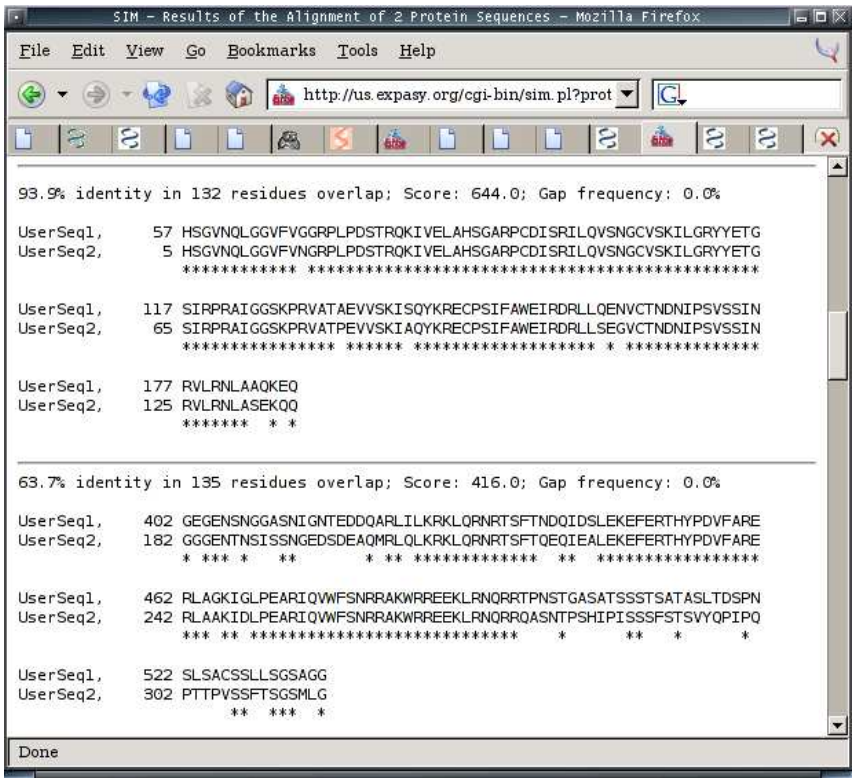

Figura 4.4: Resultado da comparação entre o gene *eyeless* com o gene *aniridia*.

Para efetuar uma breve análise da estrutura do BLAST, obteve-se o pacote com os códigos-fontes das ferramentas do NCBI - ncbi-tools. Esse pacote está disponível no servidor FTP[5] do NCBI via FTP anônimo. O pacote está disponível no arquivo `ncbi.tar.gz` .

As ferramentas estão escritas em linguagem C. E cada arquivo .c ou .h apresenta uma descrição de sua função, bem como um relatório completo de suas revisões.

A maior parte do código-fonte das ferramentas está localizada no subdiretório `tools` - tanto os *headers* como os arquivos principais. A Figura 4.5 ilustra a estrutura de diretórios do pacote descompactado. A Figura 4.6 exibe o conteúdo do sub-diretório `network` .

```
$ls -F
access/       build/          corelib/    gif/        README
algo/         build.me*       ctools/     include/    README.htm
api/          build.me64*     data/       lib/        regexp/
asn/          cdromlib/       ddv/        link/       sequin/
asnlib/       checkout.date   demo/       make/       tools/
asnstat/      cn3d/           desktop/    network/    util/
bin/          config/         doc/        object/     VERSION
biostruc/     connect/        errmsg/     platform/   vibrant/
```

Figura 4.5: Conteúdo do arquivo ncbi.tar.gz descompactado.

```
apple/      entrez/     medarch/    nsdemocl/   spell/      vibnet/
blast3/     id1arch/    netmanag/   pcnfs/      suggest/    wwwblast/
encrypt/    id2arch/    nsclilib/   socks/      taxon1/
```

Figura 4.6: Conteúdo do sub-diretório network.

Os arquivos que disponibilizam a interface Web para o acesso à ferramenta estão localizados no sub-diretório `network/wwwblast`. A Figura 4.7 exibe o conteúdo deste sub-diretório.

[5] ftp://ftp.ncbi.nih.gov/

```
blast.cgi*              megablast_cs.html       rpsblast_cs.html
blast_cs.cgi*           megablast.html          rpsblast.html
blast_cs.html           ncbi_blast.rc           rpsblast.log
blast.html              psiblast.cgi*           rpsblast.rc
blast.rc                psiblast_cs.cgi*        Src/
config_setup.pl         psiblast_cs.html        wblast2.cgi*
data/                   psiblast.html           wblast2_cs.cgi*
db/                     psiblast.log            wblast2_cs.html
discontiguous.html      psiblast.rc             wblast2.html
docs/                   readme.html             wwwblast.log
images/                 README.rps
index.html              readme.txt
```

Figura 4.7: Conteúdo do sub-diretório network/wwwblast.

No sub-diretório `network/wwwblast` encontra-se tanto os arquivos HTML como os arquivos CGI (Common Gateway Interface). A página padrão, definida pelo arquivo `index.html`, exibe apenas os *links* para os diversos programas:

```
* Regular BLAST without client-server support
* Regular BLAST with client-server support
* PSI/PHI BLAST without client-server support
* PSI/PHI BLAST with client-server support
* Mega BLAST without client-server support
* Mega BLAST with client-server support
* RPS BLAST without client-server support
* RPS BLAST with client-server support
* BLAST 2 sequences without client-server support
* BLAST 2 sequences with client-server support
* Readme file
```

O arquivo `blast.html` exibe a interface Web para acesso ao BLAST contendo um formulário HTML bastante simplificado. O acesso ao BLAST é feito via CGI pelo arquivo `blast.cgi`:

```
<FORM ACTION="blast.cgi" METHOD = POST NAME="MainBlastForm"
                         ENCTYPE= "multipart/form-data">
```

A página inicial para submeter uma seqüência para alinhamento através do BLAST é o arquivo `blast.html`.

A página referente ao *Blast 2 sequences*, utilizada no experimento do alinhamento do *eyeless* e *aniridia*, é definida pelo arquivo `wblast2.html`. Este, por sua vez, utiliza o arquivo de CGI `wblast2.cgi`.

```
<HTML>
<HEAD>
<title>Blast 2 Sequences</title>

<FORM NAME="bl2" method="Post" action="wblast2.cgi?0"
                        enctype="multipart/form-data">
```

Até mesmo o conteúdo do *Entrez* está disponível no pacote. Os respectivos arquivos encontram-se no sub-diretório `network/entrez/client`.

```
$ls network/entrez/client/ -F

netentr.asn     netentr.h     netlib.h     objneten.c
netentr.c       netlib.c      netpriv.h    objneten.h
```

As matrizes BLOSUM e PAM, que são utilizadas no processo de alinhamento, conforme a configuração do usuário, estão localizadas no sub-diretório `data`.

```
$ more data/BLOSUM62
#  Matrix made by matblas from blosum62.iij
#  * column uses minimum score
#  BLOSUM Clustered Scoring Matrix in 1/2 Bit Units
#  Blocks Database = /data/blocks_5.0/blocks.dat
#  Cluster Percentage: >= 62
#  Entropy = 0.6979, Expected = -0.5209
   A  R  N  D  C  Q  E  G  H  I  L  K  M  F  P  S  T  W  Y  V  B  Z  X  *
A  4 -1 -2 -2  0 -1 -1  0 -2 -1 -1 -1 -1 -2 -1  1  0 -3 -2  0 -2 -1  0 -4
R -1  5  0 -2 -3  1  0 -2  0 -3 -2  2 -1 -3 -2 -1 -1 -3 -2 -3 -1  0 -1 -4
N -2  0  6  1 -3  0  0  0  1 -3 -3  0 -2 -3 -2  1  0 -4 -2 -3  3  0 -1 -4
D -2 -2  1  6 -3  0  2 -1 -1 -3 -4 -1 -3 -3 -1  0 -1 -4 -3 -3  4  1 -1 -4
C  0 -3 -3 -3  9 -3 -4 -3 -3 -1 -1 -3 -1 -2 -3 -1 -1 -2 -2 -1 -3 -3 -2 -4
Q -1  1  0  0 -3  5  2 -2  0 -3 -2  1  0 -3 -1  0 -1 -2 -1 -2  0  3 -1 -4
E -1  0  0  2 -4  2  5 -2  0 -3 -3  1 -2 -3 -1  0 -1 -3 -2 -2  1  4 -1 -4
G  0 -2  0 -1 -3 -2 -2  6 -2 -4 -4 -2 -3 -3 -2  0 -2 -2 -3 -3 -1 -2 -1 -4
H -2  0  1 -1 -3  0  0 -2  8 -3 -3 -1 -2 -1 -2 -1 -2 -2  2 -3  0  0 -1 -4
I -1 -3 -3 -3 -1 -3 -3 -4 -3  4  2 -3  1  0 -3 -2 -1 -3 -1  3 -3 -3 -1 -4
L -1 -2 -3 -4 -1 -2 -3 -4 -3  2  4 -2  2  0 -3 -2 -1 -2 -1  1 -4 -3 -1 -4
K -1  2  0 -1 -3  1  1 -2 -1 -3 -2  5 -1 -3 -1  0 -1 -3 -2 -2  0  1 -1 -4
M -1 -1 -2 -3 -1  0 -2 -3 -2  1  2 -1  5  0 -2 -1 -1 -1 -1  1 -3 -1 -1 -4
F -2 -3 -3 -3 -2 -3 -3 -3 -1  0  0 -3  0  6 -4 -2 -2  1  3 -1 -3 -3 -1 -4
P -1 -2 -2 -1 -3 -1 -1 -2 -2 -3 -3 -1 -2 -4  7 -1 -1 -4 -3 -2 -2 -1 -2 -4
S  1 -1  1  0 -1  0  0  0 -1 -2 -2  0 -1 -2 -1  4  1 -3 -2 -2  0  0  0 -4
T  0 -1  0 -1 -1 -1 -1 -2 -2 -1 -1 -1 -1 -2 -1  1  5 -2 -2  0 -1 -1  0 -4
W -3 -3 -4 -4 -2 -2 -3 -2 -2 -3 -2 -3 -1  1 -4 -3 -2 11  2 -3 -4 -3 -2 -4
Y -2 -2 -2 -3 -2 -1 -2 -3  2 -1 -1 -2 -1  3 -3 -2 -2  2  7 -1 -3 -2 -1 -4
```

```
V  0 -3 -3 -3 -1 -2 -2 -3 -3  3  1 -2  1 -1 -2 -2  0 -3 -1  4 -3 -2 -1 -4
B -2 -1  3  4 -3  0  1 -1  0 -3 -4  0 -3 -3 -2  0 -1 -4 -3 -3  4  1 -1 -4
Z -1  0  0  1 -3  3  4 -2  0 -3 -3  1 -1 -3 -1  0 -1 -3 -2 -2  1  4 -1 -4
X  0 -1 -1 -1 -2 -1 -1 -1 -1 -1 -1 -1 -1 -1 -2  0  0 -2 -1 -1 -1 -1 -1 -4
* -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4 -4  1
```

A licença, segundo a qual as ferramentas do NCBI são disponibilizadas, aparece logo no início de todos os arquivos.

```
* ==========================================      ======  ======  ======  ======  ==
*
*                                 PUBLIC   DOMAIN   NOTICE
*                     National   Center   for  Biotechnology     Information
*
*   This  software/database      is  a  "United   States   Government    Work"  under   the
*   terms   of  the  United   States   Copyright    Act.    It  was  written   as  part   of
*   the  author's   offical   duties   as  a  United   States   Government    employee    and
*   thus   cannot   be  copyrighted.      This  software/database      is  freely   available
*   to  the  public   for  use.  The  National   Library   of  Medicine   and  the  U.S.
*   Government    have  not  placed   any  restriction    on  its  use  or  reproduction.
*
*   Although   all  reasonable    efforts   have  been  taken  to  ensure   the  accuracy
*   and  reliability    of  the  software   and  data,  the  NLM  and  the  U.S.
*   Government    do  not  and  cannot   warrant   the  performance    or  results   that
*   may  be  obtained   by  using  this  software   or  data.  The  NLM  and  the  U.S.
*   Government    disclaim   all  warranties,    express   or  implied,   including
*   warranties    of  performance,    merchantability     or  fitness   for  any  particular
*   purpose.
*
*   Please   cite   the  author   in  any  work   or  product   based   on  this   material.
*
* ==========================================      ======  ======  ======  ======  ==*/
```

Dados sobre o código, tais como autor, data de criação, versão da revisão e descrição ou função do programa, também estão sempre presentes.

```
* File  Name:    $RCSfile:    wwwblast.c,v    $
*
* Author:    Sergei   Shavirin
*
* Initial   Creation   Date:   03/15/2000
*
* $Revision:    1.13  $
*
* File  Description:
*         Standalone    WWW  Blast   CGI  program.
```

Capítulo 5

ClustalW e ClustalX

O ClustalW é a versão Web de um dos programas de alinhamento múltiplo mais utilizados (Clustal) [Prosdocimi et tal (2003)]. O ClustalX nada mais é do que uma interface gráfica (X Window) para o ClustalW.

A Figura 5.1 mostra o alinhamento de cinco seqüências protéicas obtido com o ClustalW, executado localmente. As seqüências utilizadas representam um caso real de estudo de seqüências de proteínas repressoras de imunidade. Todos os procedimentos empregados estão apresentados ao longo da presente seção desde a obtenção das seqüências em um banco de dados públicos até a obtenção do alinhamento. A Figura 5.2 mostra o mesmo alinhamento obtido com o ClustalX executado localmente. Foi utilizado o mesmo arquivo de entrada usado para o alinhamento com o ClustalW.

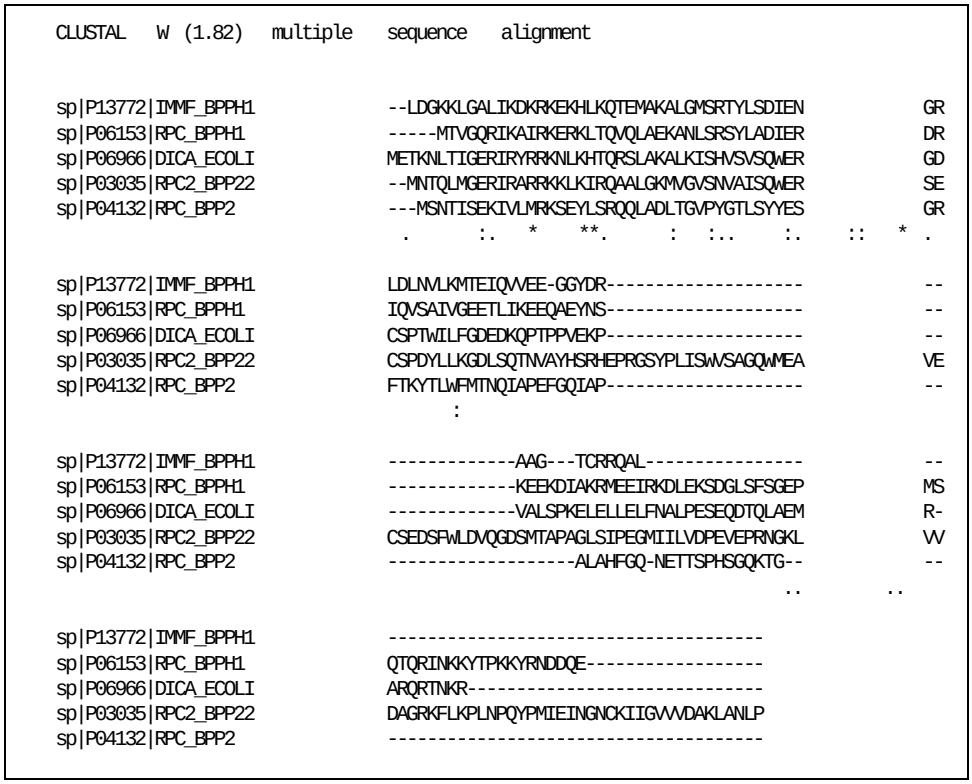

Figura 5.1: O alinhamento de cinco proteínas no ClustalW.

Alternativamente, o ClustalW pode ser utilizado via internet através de alguns *sites* que disponibilizam a ferramenta mediante uma interface Web. Mas o processo de obtenção, instalação e utilização local das ferramentas é bastante simples. E é isso que passa a ser descrito neste capítulo.

5.1 Obtendo e instalando o ClustalW e o ClustalX

O ClustalW e o ClustalX podem ser obtidos facilmente na internet em diferentes formatos binários (.deb, .rpm, tar.gz etc.)[1]. Pode-se optar também por obter o código fonte e compilar o pacote.

[1]Pacotes Debian, por exemplo, podem ser obtidos em:
http://packages.qa.debian.org/c/clustalw-mpi.html

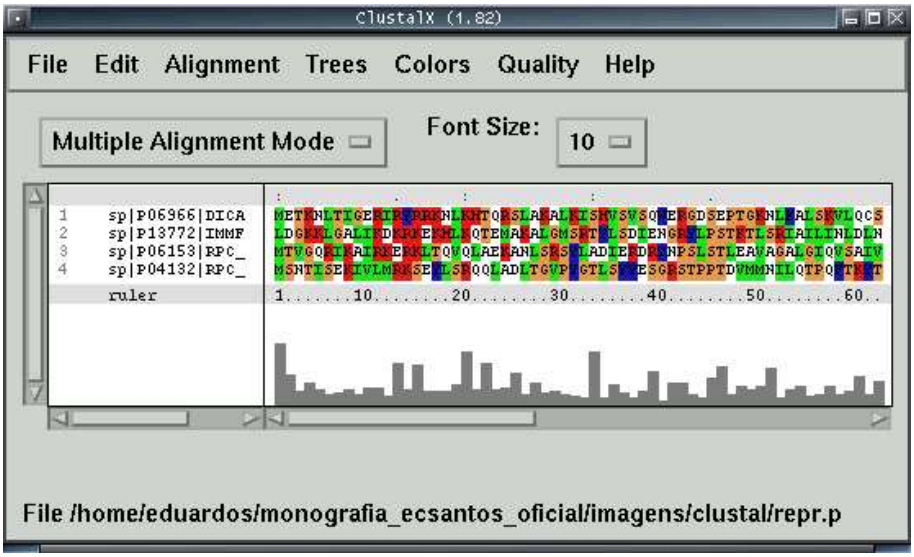

Figura 5.2: O alinhamento de proteínas no ClustalX

A instalação do ClustalX requer o ClustalW além de outros pacotes NCBI. Por isso, a instalação do ClustalX normalmente já traz junto o ClustalW. Com efeito, isso se mostrou verdadeiro tanto na versão oficial em formato tar.gz como também na versão do projeto Debian. A Figura 5.3 ilustra a instalação do pacote `clustalx` via `apt-get` .

```
# apt-get install clustalx
Lendo Lista de Pacotes... Pronto
Construindo Árvore de Dependências... Pronto
Os pacotes extra a seguir serão instalados:
  clustalw libncbi6 libvibrant6 ncbi-data ncbi-tools6 vibrant6
Suggested packages:
  seaview
Os NOVOS pacotes a seguir serão instalados:
  clustalw clustalx libncbi6 libvibrant6 ncbi-data ncbi-tools6
  vibrant6
0 pacotes atualizados, 7 novos instalados, 0 a serem removidos e
  422 não atualizados.
É precis fazer o download de 5752kB de arquivos.
Depois de desempacotar, 16,9MB adicionais de espaço em disco serão
usados.
Quer continuar? [S/n]
```

Figura 5.3: Instalação dos programas ClustalX/ClustalW e dependências.

5.2 Sobre a licença do ClustalW/ClustalX

Apesar de serem programas de código aberto, o ClustalW e o ClustalX não se enquadram como software livre. A licença segundo a qual são disponibilizados apresenta uma restrição: para poder distribuir uma versão alterada do programa é preciso requerer autorização dos autores. É importante salientar que isso vale também para os pacotes derivados do ClustalW e ClustalX distribuídos pela *Debian*[2]. As Figuras 5.4, 5.5 e 5.6 ilustram a licença contida no pacote distribuído pela Debian.

```
This package was debianized by Andreas Tille <tille@debian.org> on

Sat, 27 Oct 2001 22:16:53 +0200

It was downloaded from:

        ftp://ftp-igbmc.u-strasbg.fr/pub/ClustalX          /  and
        ftp://ftp-igbmc.u-strasbg.fr/pub/ClustalW          /

while the source was merged to one common upstream source
(see README.Debian)

Authors:

Toby Gibson <Toby.Gibson@EMBL-Heidelberg.de>
Julie Thompson <julie@titus.u-strasbg.fr>
Des Higgins <d.higgins@ucc.ie>

Copyright:

Non-free. You cannot distribute it at will.
Licence included here:
```

Figura 5.4: Licença do ClustalW no pacote distribuído pela Debian - parte 1.

[2]http://packages.debian.org/unstable/science/

```
**********************
LICENCE   FOR  CLUSTAL   W
**********************

Clustal  W (hereafter   "the  program")   is copyright   (c)  1994-1998
by Julie  D.  Thompson,   Desmond   G.  Higgins   and  Toby  J.  Gibson.

Permission   is  granted   to  copy,  distribute   and  use  the  program
provided   no  fee  is  charged   for  it  and  provided   that  this  copyright
and  licence   notice   is  not  removed   or  altered.

The  full  source   code  of  the  program   is  provided   free.  You  should  not
distribute   a  modified   version   of  the  program   without   obtaining   the
permission   of  the  authors.   You  must  keep  the  original   copyright   and
licence   notice.   You  must  also  document   clearly   the  modifications    you
have  made.  You  must  make  clear  that  this  is  not  the  original   version.

Commercial   distributors    of  Clustal   W  are  requested   to  contact   the
Clustal  W  authors   in  order   to  take  out  a  non-exclusive    licence.   See
the  README   file  included   with  Clustal   W  for  a  rationale.

You  should  understand   that  this  software   is  provided   as-is.   The  authors
make  no  claims   towards   its  suitability    for  any  purpose   and  accept
absolutely   no  liability   for  any  damages   the  program   may  cause.   Use  at
your  own  risk.

*  End  of  licence
```

Figura 5.5: Licença do ClustalW no pacote distribuído pela Debian - parte 2.

```
Special authorization for Debian:

From: "Toby Gibson" <Toby.Gibson@EMBL-Heidelberg.de>
Date: Thu, 17 Dec 1998 14:37:02 +0100
To: Stephane Bortzmeyer <bortzmeyer@debian.org>
Subject: Re: Fwd: clustalw_1.7-4_i386.changes REJECTED

Hi Stephane,

Now that we have thought about it, I don't think we can meet your
stricter free criterion. There are already several companies who
bundle Clustal W in sequence analysis packages and so are effectively
selling it. They have paid for non-exclusive licences even though
anyone can get the program for free: but they must have a multiple
alignment engine, so we might as well earn some money which we can
put toward further development.

I think the main thing is to allow the distribution at all by Debian.
We seem to have reached this point.

Please do include this licence in the Debian package and I hope the
release can go smoothly from now on.
```

Figura 5.6: Licença do ClustalW no pacote distribuído pela Debian - parte 3.

5.3 Alinhamento Múltiplo no ClustalW

O uso do ClustalW em um problema de alinhamento múltiplo é apresentado nessa seção, seguindo um exercício proposto por [Tekaia (1996)]. Neste exercício, diferentes proteínas repressoras são comparadas pela técnica de alinhamento múltiplo. As seqüências alinhadas foram:

- *dica_ecoli*
- *immf_bpph1*
- *rpc_bpph1*
- *rpc_bpp2*
- *rpc2_bpp22*

As seqüências foram obtidas no SwissProt[3]. Passos para obtenção das seqüências no formato FASTA através do SwissProt:

1. Utilizar a ferramenta de busca do próprio SwissProt. Procurar por *dica_ecoli*.

2. No final da página sobre a proteína pesquisada, o pesquisador encontra um *link* entitulado *# in FASTA format*, onde # é o *Primary accession number*.

3. Surgirá uma janela contendo apenas a seqüência requisitada em formato FASTA. Salvar a seqüência em um arquivo e repetir o processo para as demais seqüências.

A metodologia empregada aqui na obtenção das seqüências propostas difere daquela descrita por [Tekaia (1996)], mas os dados obtidos são exatamente os mesmos, fato constatado com a utilização do comando `diff` .

Para submeter seqüências ao ClustalW, deve-se preparar um arquivo, texto contendo as seqüências em um dos formatos válidos [Tekaia (1996)]. Todas as seqüências devem estar contidas no mesmo arquivo uma após a outra. A Figura 5.7 ilustra um exemplo de arquivo de entrada com seqüências no formato FASTA. O arquivo foi gerado a partir das seqüências obtidas na consulta ao SwissProt.

Nas versões anteriores à versão 1.7 do ClustalW, já eram aceitos seis formatos de arquivos de entrada: FASTA (Pearson), NBRF/PIR, EMBL/Swiss Prot, GDE, CLUSTAL, GCG/ MSF. Na versão 1.7, foi acrescentado o suporte ao formato RSF, usado pela versão 9 do GCG.

A descrição de cada um dos formatos válidos para as seqüências de entrada foge ao escopo deste trabalho. Sua citação é feita aqui para destacar a importância do conhecimento desses formatos por parte do pesquisador.

Para cada um dos formatos, existe uma determinação quanto ao primeiro caracter ou palavra que deve aparecer no arquivo. A Tabela 5.1 apresenta as definições para cada tipo de formato. Na versão original do ClustalW, arquivos no formato GCG/MSF tinham que ser iniciados com a palavra PILEUP. A partir da versão 1.7, o arquivo pode ser iniciado por: PILEUP, !!AA_MULTIPLE_ALIGNMENT, !!NA_MULTIPLE_ALIGNMENT ou ainda pelos caracteres MSF. Neste último caso, deve-se ter os caracteres .. (dois pontos seguidos) no final da linha.

Não é necessário explicitar ao ClustalW (ou ao ClustalX) qual é o formato do arquivo de entrada. O próprio programa identifica isso de acordo com os caracteres iniciais do arquivo. Todas as seqüências no arquivo devem estar no mesmo formato.

Também não é necessário explicitar o tipo de seqüências: ácidos nucléicos (DNA/RNA) ou aminoácidos (proteínas). O próprio programa identifica isso.

[3]http://us.expasy.org/

Tabela 5.1: Formatos de entrada possíveis para o ClustalW e os repectivos caracteres iniciais.

Formato de arquivo	**Caracter ou *string* inicial**
FASTA	>
NBRF	>P1; ou >D1;
EMBL/SWISS	ID
GDE protein	%
GDE nucleotide	#
CLUSTAL	CLUSTAL
GCG/MSF	PILEUP ou !!AA_MULTIPLE_ALIGNMENT ou !!NA_MULTIPLE_ALIGNMENT ou MSF (finalizando a primeira linha com ..)

```
>sp|P06966|DICA_ECOLI       HTH-type   transcriptional     regulator    dicA  (Repre
METKNLTIGERIRYRRKNLKHTQRSLAKALKISHVSVSQWERGDSE            PTGKNLF  ALSKVL  Q
CSPTWILFGDEDKQPTPPVEKPVALSPKELELLELFNALPESEQDT            QLAEMRA  RVKNFN  K
LFEELLKARQRTNKR
>sp|P13772|IMMF_BPPH1       ImmF  control   region   10  kDa  protein   -  Bacteriop
LDGKKLGALIKDKRKEKHLKQTEMAKALGMSRTYLSDIENGRYLPS            TKTLSRI  AILINL  D
LNVLKMTEIQVVEEGGYDRAAGTCRRQAL
>sp|P06153|RPC_BPPH1       Immunity   repressor    protein   -  Bacteriophage    phi-
MTVGQRIKAIRKERKLTQVQLAEKANLSRSYLADIERDRYNPSLST            LEAVAGA  LGIQVS  A
IVGEETLIKEEQAEYNSKEEKDIAKRMEEIRKDLEKSDGLSFSGEP            MSQEAVE  SLMEAM  E
HIVRQTQRINKKYTPKKYRNDDQE
>sp|P04132|RPC_BPP2       Repressor    protein    C  -  Bacteriophage     P2.
MSNTISEKIVLMRKSEYLSRQQLADLTGVPYGTLSYYESGRSTPPT            DVMMNIL  QTPQFT  K
YTLWFMTNQIAPEFGQIAPALAHFGQNETTSPHSGQKTG
>sp|P03035|RPC2_BPP22       Repressor    protein    C2  -  Bacteriophage     P22,  *
MNTQLMGERIRARRKKLKIRQAALGKMVGVSNVAISQWERSETEPN            GENLLAL  SKALQC  S
PDYLLKGDLSQTNVAYHSRHEPRGSYPLISWVSAGQWMEAVEPYHK            RAIENWH  DTTVDC  S
EDSFWLDVQGDSMTAPAGLSIPEGMIILVDPEVEPRNGKLVVAKLE            GENEATF  KKLVMD  A
GRKFLKPLNPQYPMIEINGNCKIIGVVVDAKLANLP
```

Figura 5.7: Exemplo de arquivo de entrada para o ClustalW. Algumas linhas foram truncadas para fins de impressão.

Seguindo com a execução do exercício proposto por [Tekaia (1996)], o programa ClustalW foi iniciado.

```
$ clustalw

 ****************************************           *******  ******  ******
 ********   CLUSTAL   W (1.82)   Multiple   Sequence   Alignments     ********
 ****************************************           *******  ******  ******


      1. Sequence   Input   From  Disc
      2. Multiple   Alignments
      3. Profile   / Structure   Alignments
      4. Phylogenetic    trees

      S. Execute   a system   command
      H. HELP
      X. EXIT  (leave   program)


Your  choice:   1
```

Inicialmente, escolhe-se a opção 1 para explicitar o arquivo de entrada, previamente preparado.

```
Sequences   should  all  be  in  1 file.

7 formats   accepted:
NBRF/PIR,   EMBL/SwissProt,     Pearson   (Fasta),   GDE,  Clustal,   GCG/MSF,   RSF.


Enter  the  name  of  the  sequence   file:    repr.pep
```

O programa pede o nome do arquivo de entrada com as seqüências, `repr.pep` , no exemplo.

```
Sequence   format   is  Pearson
Sequences    assumed   to  be  PROTEIN


Sequence   1:  dica_ecoli          135  aa
Sequence   2:  immf_bpph1           89  aa
Sequence   3:  rpc_bpph1           144  aa
Sequence   4:  rpc_bpp2             99  aa
Sequence   5:  rpc2_bpp22          216  aa

 ****************************************           *******  ******  ******
```

```
********   CLUSTAL   W (1.82)   Multiple   Sequence   Alignments     ********
*****************************************          *******  ******  ******


     1. Sequence   Input  From  Disc
     2. Multiple   Alignments
     3. Profile   / Structure   Alignments
     4. Phylogenetic    trees

     S. Execute   a system   command
     H. HELP
     X. EXIT  (leave   program)


Your  choice:   2
```

O formato do arquivo de entrada é identificado pelo próprio programa e o tamanho de cada seqüência é calculado. O menu principal volta a aparecer. Agora que as seqüências já foram lidas, pode-se optar pelo alinhamento múltiplo das mesmas (opção 2 do menu principal). Surge o submenu relacionado com alinhamento múltiplo.

```
******   MULTIPLE   ALIGNMENT   MENU  ******


    1.   Do complete   multiple   alignment   now  (Slow/Accurate)
    2.   Produce   guide  tree  file  only
    3.   Do alignment   using  old  guide  tree  file

    4.   Toggle   Slow/Fast   pairwise   alignments    = SLOW

    5.   Pairwise   alignment   parameters
    6.   Multiple   alignment   parameters

    7.   Reset  gaps  before   alignment?    = OFF
    8.   Toggle   screen   display             = ON
    9.   Output   format   options

    S.   Execute   a system   command
    H.   HELP
    or  press   [RETURN]   to  go  back  to  main  menu


Your  choice:
```

Antes de proceder com o alinhamento, é interessante verificar os parâmetros que serão usados. Para tratar disso, é necessário antes um breve comentário sobre como o alinhamento múltiplo é realizado pelo programa. O algoritmo do

ClustalW produz inicialmente um alinhamento par-a-par entre as seqüências. A partir daí, o programa gera um arquivo com dados sobre a árvore filogenética com as seqüências envolvidas. Analisando a filogenética do conjunto de seqüências, o algoritmo realiza automaticamente o alinhamento múltiplo. Os parâmetros usados no alinhamento par-a-par podem ser configurados através da opção 5 do menu *MULTIPLE ALIGNMENT*. E os parâmetros usados no alinhamento múltiplo propriamente dito, podem ser visualizados e alterados através da opção 6 do menu *MULTIPLE ALIGNMENT*. O sub-menu *PAIRWISE ALIGNMENT PARAMETERS* que apresenta as configurações para os alinhamentos par-a-par é exibido da seguinte forma:

```
*********    PAIRWISE    ALIGNMENT    PARAMETERS    *********


      Slow/Accurate     alignments:

      1.  Gap  Open  Penalty          :10.00
      2.  Gap  Extension    Penalty    :0.10
      3.  Protein   weight   matrix    :Gonnet   series
      4.  DNA  weight   matrix         :IUB

      Fast/Approximate      alignments:

      5.  Gap  penalty                 :3
      6.  K-tuple   (word)   size      :1
      7.  No.  of  top  diagonals      :5
      8.  Window   size                :5

      9.  Toggle   Slow/Fast    pairwise    alignments    = SLOW

      H.  HELP


Enter   number   (or  [RETURN]    to  exit):
```

O sub-menu *MULTIPLE ALIGNMENT PARAMETERS* que apresenta as configurações para o alinhamento múltiplo é exibido da seguinte forma:

```
 *********    MULTIPLE    ALIGNMENT    PARAMETERS    *********


      1.  Gap  Opening   Penalty                  :10.00
      2.  Gap  Extension    Penalty               :0.20
      3.  Delay   divergent    sequences          :30  %

      4.  DNA  Transitions    Weight              :0.50

      5.  Protein   weight   matrix               :Gonnet   series
```

```
      6. DNA weight matrix                    :IUB
      7. Use negative matrix                  :OFF

      8. Protein Gap Parameters

      H. HELP

Enter number (or [RETURN] to exit):
```

Neste momento, é interessante observar que a versão do ClustalW utilizada aqui, versão 1.8, apresenta algumas diferenças em relação à versão 1.4 do programa, utilizada por [Tekaia (1996)]. Além de apresentar alguns parâmetros e opções a mais, a versão 1.8 traz alguns valores padrões diferentes da versão 1.4.

Assim, para que o resultado obtido no exemplo proposto fique mais próximo do resultado apresentado por [Tekaia (1996)], deve-se observar atentamente cada parâmetro que foi utilizado pelo autor.

A exata compreenção do significado de cada parâmetro é fundamental para o sucesso da pesquisa. Quanto à operacionalização, a configuração dos parâmetros se mostra bastante intuitiva.

A escolha do formato de saída desejado é definida através da opção 9 (*Output format options*) do menu *MULTIPLE ALIGNMENT MENU*. É possível optar por mais de um formato de saída.

```
******  MULTIPLE  ALIGNMENT  MENU  ******


    1.  Do complete  multiple  alignment  now (Slow/Accurate)
    2.  Produce  guide  tree  file  only
    3.  Do alignment  using  old  guide  tree  file

    4.  Toggle  Slow/Fast  pairwise  alignments  = SLOW

    5.  Pairwise  alignment  parameters
    6.  Multiple  alignment  parameters

    7.  Reset  gaps  before  alignment?  = OFF
    8.  Toggle  screen  display          = ON
    9.  Output  format  options

    S.  Execute  a system  command
    H.  HELP
    or press  [RETURN]  to go back  to main  menu


Your choice:  9
```

```
*********   Format  of  Alignment   Output  *********

     1. Toggle  CLUSTAL  format  output       =  ON
     2. Toggle  NBRF/PIR  format  output      =  OFF
     3. Toggle  GCG/MSF  format  output       =  ON
     4. Toggle  PHYLIP  format  output        =  OFF
     5. Toggle  NEXUS  format  output         =  OFF
     6. Toggle  GDE  format  output           =  OFF

     7. Toggle  GDE  output  case             =  LOWER
     8. Toggle  CLUSTALW  sequence  numbers   =  OFF
     9. Toggle  output  order                 =  ALIGNED

     0. Create  alignment  output  file(s)  now?

     T. Toggle  parameter  output             = OFF

     H. HELP

Enter  number  (or  [RETURN]  to  exit):
```

Após definir todos os parâmetros desejados, pode-se iniciar o alinhamento através da opção 1 (*Do complete multiple alignment now*) do menu *MULTIPLE ALIGNMENT MENU*.

```
******  MULTIPLE  ALIGNMENT  MENU  ******

    1.  Do  complete  multiple  alignment  now  (Slow/Accurate)
    2.  Produce  guide  tree  file  only
    3.  Do  alignment  using  old  guide  tree  file

    4.  Toggle  Slow/Fast  pairwise  alignments  = SLOW

    5.  Pairwise  alignment  parameters
    6.  Multiple  alignment  parameters

    7.  Reset  gaps  before  alignment?  = OFF
    8.  Toggle  screen  display            = ON
    9.  Output  format  options

    S.  Execute  a  system  command
    H.  HELP
    or  press  [RETURN]  to  go  back  to  main  menu

Your  choice:  1
```

O programa pede então para que o usuário forneça um nome para cada arquivo de saída. Além de um arquivo para cada formato ativado no menu *Format of*

Alignment Output, também é criado um arquivo .dnd com as informações para a construção da árvore genética.

O alinhamento par-a-par é realizado e suas respectivas pontuações calculadas. A partir dessas pontuações, o programa constrói o arquivo .dnd e inicia o alinhamento múltiplo.

```
Enter a name for the CLUSTAL output file [repr.aln]:
Enter a name for the GCG/MSF output file [repr.msf]:

Enter name for new GUIDE TREE file [repr.dnd]:

Start of Pairwise alignments
Aligning...
Sequences (1:2) Aligned. Score: 17
Sequences (1:3) Aligned. Score: 21
Sequences (1:4) Aligned. Score: 16
Sequences (1:5) Aligned. Score: 27
Sequences (2:3) Aligned. Score: 29
Sequences (2:4) Aligned. Score: 19
Sequences (2:5) Aligned. Score: 15
Sequences (3:4) Aligned. Score: 12
Sequences (3:5) Aligned. Score: 11
Sequences (4:5) Aligned. Score: 11
Guide tree file created: [repr.dnd]
Start of Multiple Alignment
There are 4 groups
Aligning...
Group 1: Delayed
Group 2: Delayed
Group 3: Delayed
Group 4: Delayed
Sequence:3 Score:563
Sequence:1 Score:887
Sequence:5 Score:1063
Sequence:4 Score:360
Alignment Score 449
```

A ordem em que as seqüências são exibidas nos arquivos de alinhamento, é definida pelo parâmetro *Toggle output order* do menu *Format of Alignment Output*. Aqui, optou-se por imprimir as seqüências na ordem em que as seqüências foram alinhadas segundo o guia para a árvore no arquivo .dnd. Essa é a opção padrão da versão 1.8 do ClustalW. Ela faz com que as seqüências sejam agrupadas conforme seu grau de relacionamento. No tutorial de [Tekaia (1996)], as seqüências foram impressas no arquivo de saída na mesma ordem que aparecem no arquivo de entrada. Essa alternativa é selecionada escolhendo-se o valor *INPUT FILE* para o parâmetro *Toggle output order* do menu *Format of Alignment Output*.

```
Consensus    length   = 218
CLUSTAL-Alignment      file  created    [repr.aln]
GCG/MSF-Alignment      file  created    [repr.msf]

CLUSTAL   W (1.82)   multiple   sequence   alignment


immf_bpph1          --LDGKKLGALIKDKRKEKHLKQTEMAKALGMSRTYLSDIE          NGRYLPS  TKTLSRI  AILIN
rpc_bpph1           -----MTVGQRIKAIRKERKLTQVQLAEKANLSRSYLADIE          RDRYNPS  LSTLEAV  AGALG
dica_ecoli          METKNLTIGERIRYRRKNLKHTQRSLAKALKISHVSVSQWE          RGDSEPT  GKNLFAL  SKVLQ
rpc2_bpp22          --MNTQLMGERIRARRKKLKIRQAALGKMVGVSNVAISQWE          RSETEPN  GENLLAL  SKALQ
rpc_bpp2            ---MSNTISEKIVLMRKSEYLSRQQLADLTGVPYGTLSYYE          SGRSTPP  TDVMMNI  LQTPQ
                     .      :.   *    **.     :   :..    :.    ::   * .    *   . :   :


immf_bpph1          LDLNVLKMTEIQVVEE-GGYDR-------------------          -------  -------  -----
rpc_bpph1           IQVSAIVGEETLIKEEQAEYNS-------------------          -------  -------  -----
dica_ecoli          CSPTWILFGDEDKQPTPPVEKP-------------------          -------  -------  -----
rpc2_bpp22          CSPDYLLKGDLSQTNVAYHSRHEPRGSYPLISWVSAGQWME          AVEPYHK  RAIENWH  DTTVD
rpc_bpp2            FTKYTLWFMTNQIAPEFGQIAP-------------------          -------  -------  -----
                          :

immf_bpph1          -------------AAG---TCRRQAL---------------          -------  -------  -----
rpc_bpph1           -------------KEEKDIAKRMEEIRKDLEKSDGLSFSGE          PMSQEAV  ESLMEAM  EHIVR
dica_ecoli          -------------VALSPKELELLELFNALPESEQDTQLAE          MR--ARV  KNFNKLF  EELLK
rpc2_bpp22          CSEDSFWLDVQGDSMTAPAGLSIPEGMIILVDPEVEPRNGK          LVVAKLE  GENEATF  KKLVM
rpc_bpp2            ------------------ALAHFGQ-NETTSPHSGQKTG-          -------  -------  -----

Press   [RETURN]    to  continue    or    X   to  stop:
```

É fácil compreender a estrutura do arquivo .dnd e compreender a ordem seguida no alinhamento. Basta uma rápida inspeção do conteúdo do arquivo. Primeiramente, ocorre o alinhamento dos pares:

dica_ecoli:rpc2_bpp22 e immf_bpph1:rpc_bpph1.

Em seguida ocorre o alinhamento múltiplo entre os dois alinhamentos já obtidos e a seqüência que restou (rpc_bpp2).

A ordem da impressão, para a opção *output order = ALIGNED*, segue em ordem crescente dos fatores referentes aos alinhamentos dos pares (0.05245 e 0.06317) ficando a seqüência sem par por último.

```
# conteúdo   do  arquivo   repr.dnd:

(
  (dica_ecoli:0.33467,rpc2_bpp22:0.39125):0.          06317,
  (immf_bpph1:0.33596,rpc_bpph1:0.37191):0.0          5245,
  rpc_bpp2:0.43751
);
```

Sobre a metodologia empregada pelo ClustalW no alinhamento de seqüências, [Gibas & Jambeck (2001)] explicam: *a heurística usada no ClustalW se baseia na análise filogenética*.

Ainda sobre a estratégia empregada no ClustalW, [Gibas & Jambeck (2001)] afirmam que ela *produz alinhamentos razoáveis em diversas condições*. Entretanto, os autores enfatizam que tal estratégia não é a prova de falhas, podendo apresentar resultados imprecisos no alinhamento e na análise filogenética de seqüências fracamente relacionadas. Apesar disso, *o alinhamento par-a-par de seqüências por meio de programação dinâmica é muito preciso para seqüências fortemente relacionadas, independentemente da matriz de pontuação ou dos valores de penalidades que sejam usados*. Quanto ao caso de seqüências fracamente relacionadas, vale destacar ainda que a precisão do alinhamento par-a-par, empregado no ClustalW, aumenta na medida em que se utiliza um número maior de seqüências.

Como já afirmado, existem diversos parâmetros envolvidos no alinhamento múltiplo de seqüências. É importante que o pesquisador tenha uma boa compreenção sobre matrizes de pontuação, valores de penalidade, perfis etc. No ClustalW, os parâmetros são definidos a partir de dois sub-menus: além do *Multiple Alignment* (Alinhamentos Múltiplos), descrito brevemente nessa seção, existe o *Profile Structure Alignment* (Alinhamentos da estrutura dos perfis). No ClustalX, os parâmetros são definidos a partir do menu suspenso *Alignment* (Alinhamento).

Capítulo 6

O TEXshade

O preparo de artigos apresentando resultados de alinhamento de seqüências constitue, geralmente, uma tarefa que pode ser dividida em duas partes. A primeira é composta das rotinas de estudo das seqüências e obtenção dos possíveis alinhamentos. A segunda refere-se ao tratamento dos dados obtidos, ou seja, dos alinhamentos obtidos, de forma a obter uma boa impressão gráfica.

O ClustalX, analisado no Capítlo anterior, tem uma opção no menu *File* para gerar um arquivo de saída *PostScript*. Mas esse recurso não mostra a mesma flexibilidade encontrada com a utilização do TEXshade. Por exemplo, o resultado é impresso em um arquivo PS à parte e deve ser inserido no documento posteriormente. Já com o TEXshade, a marcação do alinhamento é gerada pelo próprio LATEX e pode, portanto, ser gerada pelo código-fonte do próprio documento que contém o relatório, dissertação etc.

Aliando o enorme poder de construcão de macros, a grande flexibilidade para utilização de cores e o alto grau de qualidade gráfica do LATEX, Eric Beitz construiu uma poderosa ferramenta, baseada unicamente em recursos do LATEX, como macros e arquivos de estilo. Ele denominou esta ferramenta de TEXshade e a disponibilizou segundo a GPL.

O TEXshade é *um programa para criação de imagens de alinhamento de seqüências com qualidade gráfica profissional.*[Gibas & Jambeck (2001)]

A ferramenta deve receber seqüências alinhadas como dados de entrada. Essas seqüências devem estar contidas em um único arquivo texto. Esse arquivo pode apresentar-se em três diferentes formatos: FASTA, MSF e ALN.

O usuário pode construir diferentes saídas utilizando perfis pré-definidos ou criando perfis personalizados. O usuário tem ainda total liberdate na definição das cores de marcação de características do alinhamento.

Para um dado alinhamento, é possível fazer marcações diversas, conforme o

interesse da pesquisa em andamento. É possível marcar apenas os resíduos idênticos, ou ainda destacar especialmente os resíduos idênticos que aparecem em todas as seqüências alinhadas. Se o pesquisador desejar, pode também optar por uma marcação que identifique resíduos protéicos de acordo com suas funcionalidades: ácidos; bases; polares; aromáticos etc. O TEXshade permite ainda marcar regiões do alinhamento, utilizar estruturas secundárias em arquivos nos formatos DSSP, STRIDE ou PHD, e muitos outros recursos.

A ferramenta possui quatro modos de marcação pré-definidos: *identical*; *similar*; *diverse* e; *functional*. As Figuras 6.1, 6.2, 6.3 e 6.4 foram inseridas nessa seção para ilustrar algumas das possibilidades do TEXshade.

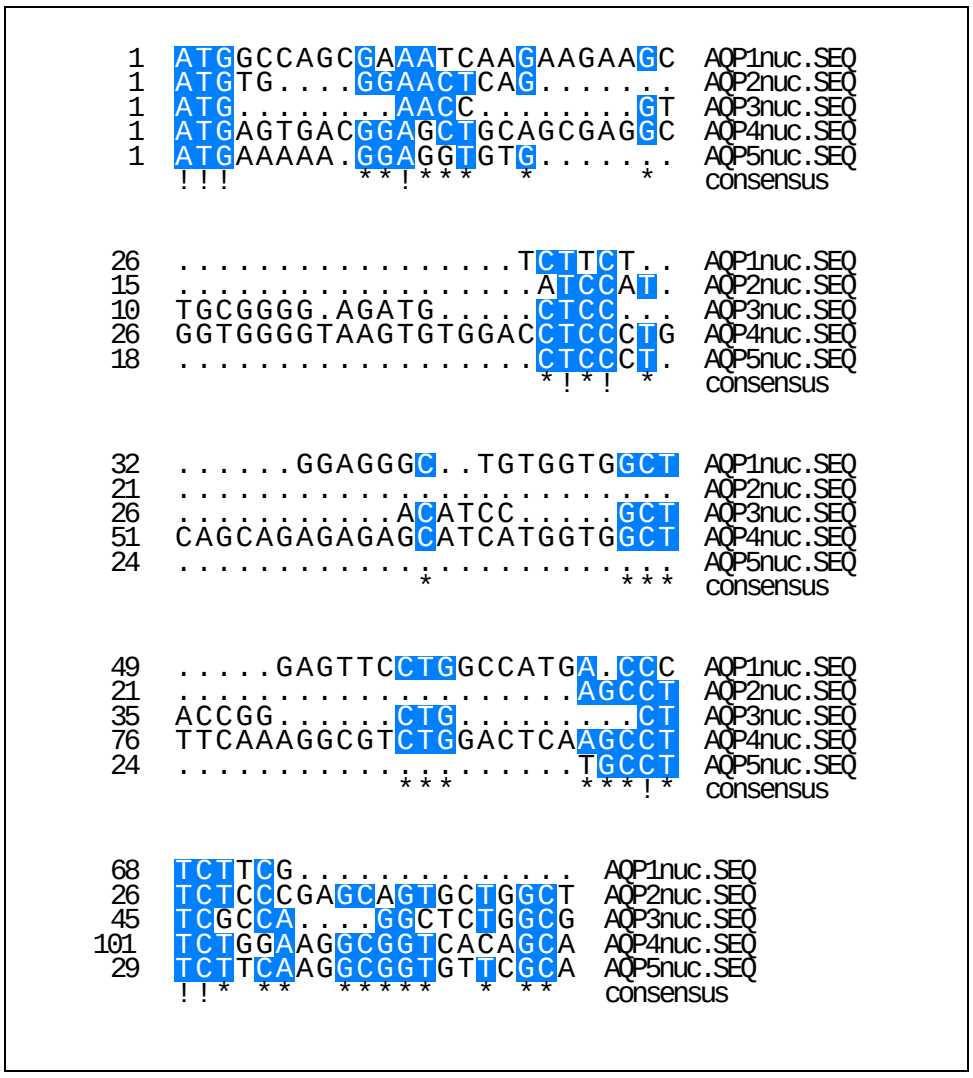

Figura 6.1: Exemplo com modo: *identical*. O arquivo de entrada foi o arquivo de exemplo AQPDNA.MSF que acompanha o pacote.

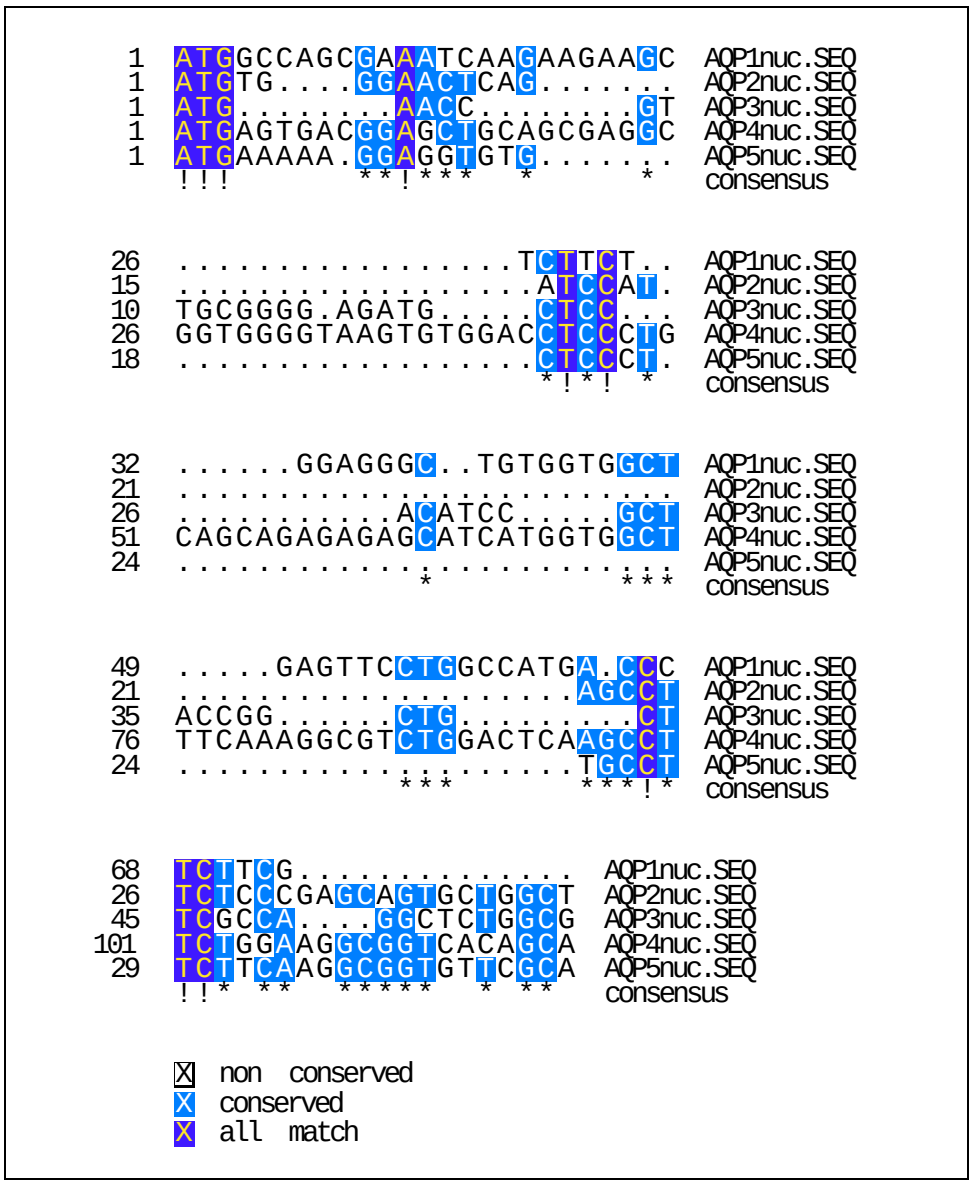

Figura 6.2: Exemplo com modo: *identical* e o parâmetro *allmatchspecial* para marcar com uma cor especial a ocorrência de resíduos idênticos em todas as seqüências utilizadas no alinhamento. O arquivo de entrada foi o arquivo de exemplo AQPDNA.MSF que acompanha o pacote.

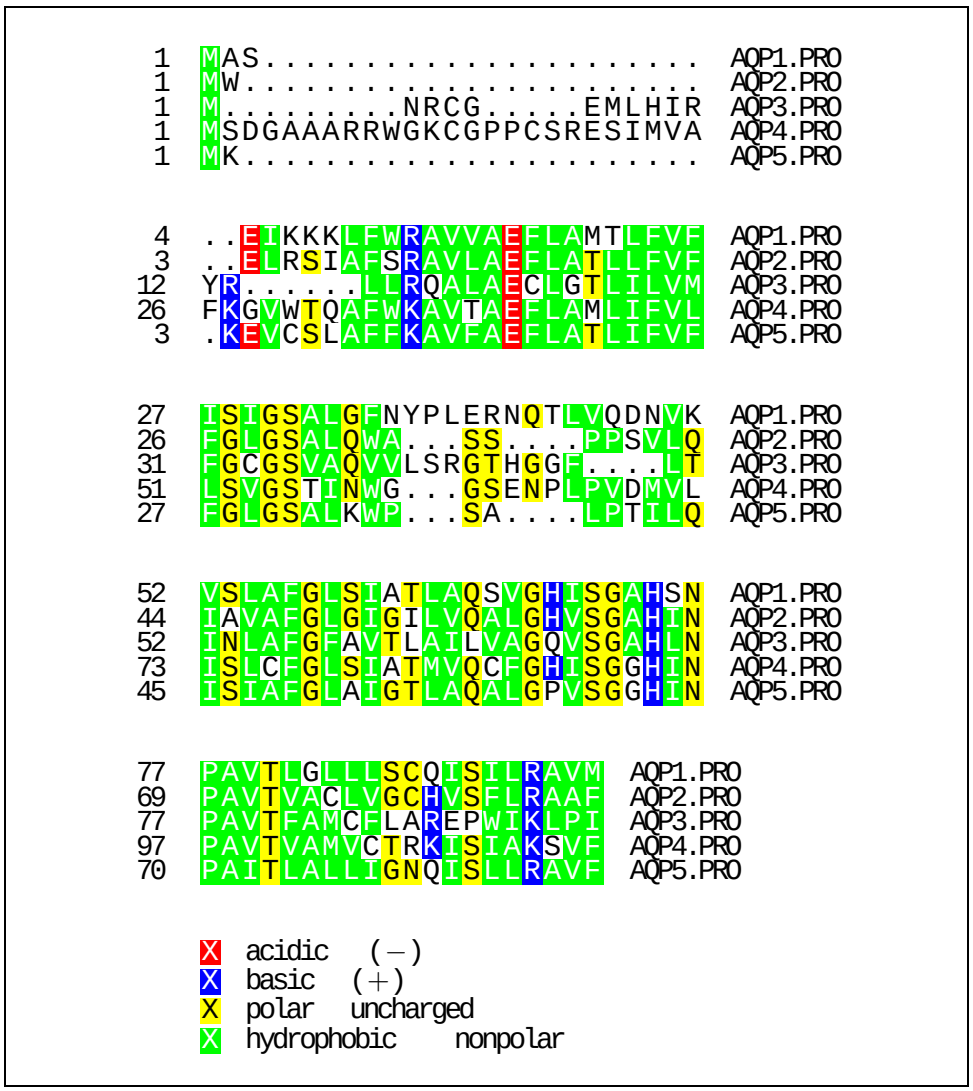

Figura 6.3: Exemplo com modo: *functional* e o tipo *hydropathy*. O arquivo de entrada foi o arquivo de exemplo AQPpro.MSF que acompanha o pacote.

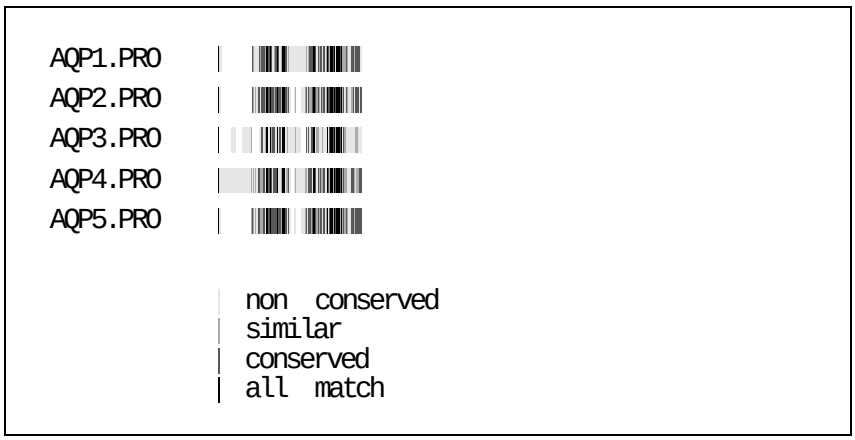

Figura 6.4: Exemplo com modo: *similar*; parâmetro *allmatchspecial*; esquema de cores *grays* e; uso do *fingerprint* com 500 resíduos por linha para imprimir os resultados em formato de linhas sem especificar os resíduos. O arquivo de entrada foi o arquivo de exemplo AQPpro.MSF que acompanha o pacote.

6.1 Requisitos para o sistema

O TEXshade requer o LATEX 2_ε, o arquivo de estilo `color.sty` e o `dvips`. Todos esses recursos são encontrados, em geral, em todas as distribuições de GNU/Linux destinadas a estações de trabalho.

6.2 Obtendo e instalando o TEXshade

O TEXshade pode ser obtido em http://homepages.uni-tuebingen.de/beitz/dltse.html. Atualmente, encontra-se na versão 1.7. Todo o pacote está concentrado em dois arquivos: o arquivo instalador `texshade.ins` e o arquivo contendo arquivos compatactados `texshade.dtx`.

Para executar o arquivo `texshade.ins` com o LATEX, basta utilizar o comando:

```
latex texshade.ins
```

Dos arquivos que são descompactados pela execução do `texshade.ins`, o único arquivo essencial para o funcionamento do TEXshade é o arquivo denominado `texshade.sty`. Basta copiar este arquivo para um diretório consultado pelo LATEX para que o TEXshade esteja pronto para ser utilizado a partir de qualquer diretório. Alternativamente, pode-se manter uma cópia deste arquivo no mesmo diretório do arquivo que será submetido ao TEXshade.

6.3 Analisando os arquivos do pacote

Ao executar o arquivo `texshade.ins` com o LATEX, os arquivos listados na Tabela 6.1 são criados no mesmo diretório.

Como afirmado anteriormente, o único arquivo essencial para a utilização do TEXshade é o arquivo `texshade.sty`. Os demais arquivos, são arquivos suplementares, usados na construção da documentação do TEXshade ou na construção do próprio `texshade.sty`.

No arquivo `standard.cod`, encontra-se as definições padrões do código genético. Tais definições são realizadas usando o comando `condon`. São copiadas para o arquivo `texshade.sty` no momento em que ele é gerado.

```
\codon{A}{GCA,GCG,GCC,GCT,GCU,GCN}
\codon{C}{TGC,TGT,UGC,UGU,TGY}
\codon{D}{GAC,GAT,GAU,GAY}
\codon{E}{GAA,GAG,GAR}
\codon{F}{TTC,TTT,UUC,UUU,TTY}
\codon{G}{GGA,GGG,GGC,GGT,GGU,GGN}
\codon{H}{CAC,CAT,CAY}
\codon{I}{ATA,ATC,ATT,AUA,AUC,AUU,ATH}
\codon{K}{AAA,AAG,AAG,AAR}
\codon{L}{CTA,CTG,CTC,CTT,TTA,TTG,CUG,CUG          ,CUC,CU  U,UUA,  UUG,YTN }
\codon{M}{ATG,AUG,ATG}
\codon{N}{AAC,AAT,AAU,AAY}
\codon{P}{CCA,CCG,CCC,CCT,CCU,CCN}
\codon{Q}{CAA,CAG,CAR}
\codon{R}{AGA,AGG,CGA,CGG,CGC,CGT,CGU,MGN          }
\codon{S}{TCT,TCC,TCG,TCA,AGT,AGC,UCU,UCC          ,UCG,UC  A,AGU,  WSN}
\codon{T}{ACT,ACC,ACG,ACA,ACU,ACN}
\codon{V}{GTA,GTG,GTC,GTT,GUA,GUG,GUC,GUU          ,GTN}
\codon{W}{TGG,UGG,TGG}
\codon{Y}{TAC,TAT,UAC,UAU,TAY}
\codon{.}{TAA,TAG,TGA,UAA,UAG,UGA,TRR}
```

Tabela 6.1: Arquivos gerados ao executar o arquivo texshade.ins

texshade.sty	:	o arquivo de estilo com todos os comandos do TEXshade;
texshade.def	:	um arquivo de parâmetros de exemplo;
AQPDNA.MSF	:	um arquivo de exemplo de um alinhamento de nucleotídes no formato MSF;
AQPpro.MSF	:	um arquivo de exemplo de um alinhamento de proteínas no formato MSF;
AQP2spec.ALN	:	um exemplo suplementar de alinhamento protéico no formato ALN;
AQP1.phd	:	informações sobre a estrutura secundária no formato PHD
AQP1.top	:	dados da topologia extraídos do AQP1.phd;
standard.cod	:	definições padrões do código genético;
ciliate.cod	:	definições do código genético macromolecular (celular);

6.4 Gerando o arquivo de documentação

A documentação sobre o TEXshade (mantida pelo próprio criador da ferramenta, Eric Beitz) pode ser obtida executando o arquivo texshade.dtx pelo LATEX. Mas deve-se deixar claro que isso só pode ser feito após a execução do arquivo texshade.ins . Assim, se for necessário obter informações prévias sobre a instalação, deve-se consultar o arquivo texshade.txt .

[Beitz (2000)] recomenda que o arquivo seja executado via LATEX duas vezes. (Usuários do LATEX já estão acostumados com esse tipo de comportamento.) Isso é uma exigência do LATEX para que seja possível gerar as referências. Nas experimentações realizadas para esta monografia, mesmo na segunda execução, o LATEX emitiu uma recomendação sugerindo uma nova execução:

```
LaTeX Warning: Label(s) may have changed.
Rerun to get cross-reference s right.
```

Deve-se enfatizar também que alguns dos arquivos gerados pela execução de texshade.ins devem estar presentes no mesmo diretório onde se encontra o texshade.dtx . Como, por exemplo, alguns arquivos de exemplos de alinhamento (AQPDNA.MSF , AQPpro.MSF etc). O arquivo texshade.sty , que é o

arquivo que contém os comandos do TEXshade, também é necessário para gerar a documentação porque o próprio processo de geração da documentação executa comandos do TEXshade sobre os arquivos de exemplos.

A execução do `texshade.dtx` com o LATEX gera, entre outros, o arquivo `texshade.dvi` (desde a primeira execução - não se deve esquecer de executar mais uma ou duas vezes). O arquivo `texshade.dvi`, por sua vez permite a geração do arquivo *PostScript* (ps) através do comando:

```
dvips texshade.dvi -o texshade.ps
```

O `dvips` não é um comando propriamente dito, mas um utilitário e precisa, portanto, estar instalado. Para gerar, uma versão em pdf (*Portable Documetn File*), usuários do GNU/Linux (e compatíveis) podem usar o utilitário `ps2pdf` - basta usar o comando:

```
ps2pdf texshade.ps texshade.pdf
```

Alguns usuários do LATEX, preferem gerar a versão pdf diretamente a partir do aplicativo `pdflatex`. Mas esse procedimento costuma gerar resultados indesejáveis. Com efeito, nas experimentações realizadas para este trabalho, esse procedimento não foi bem sucedido para gerar adequadamente os exemplos de resultados obtidos com o TEXshade.

Alguns sistemas disponibilizam também o comando `dvipdf` que permite gerar a versão PDF diretamente a partir do arquivo dvi.

6.5 Tipos de arquivos reconhecidos pelo TEXshade

O TEXshade aceita como entrada, arquivos de alinhamento em três formatos: MSF, ALN ou FASTA. Arquivos no formato MSF apresentam-se como ilustrado na Figura 6.5. Logo na primeira linha, é informado o tipo de alinhamento - se é um alinhamento de nucleotídeos ou se é um alinhamento de peptídeos. O parâmetro que indica isso é o parâmetro *Type*, que pode assumir o valor P para indicar um alinhamento de peptídeos ou N para indicar um alinhamento de nucleotídeos.

Cada seqüência em um arquivo MSF é identificada por um nome localizado no início de cada linha de sequenciamento. Os nomes identificadores das seqüências são definidos no início do arquivo em linhas contendo os parâmetros *Name*; *Len* (*length*); *Check* (*checksum*) e; *Weight*.

Uma linha contendo duas barras (//) é obrigatória e serve para separar a seção inicial, com informações sobre as seqüências contidas no arquivo, das seqüências propriamente ditas.

```
AQPDNA.MSF     MSF:  979    Type:  N  Freitag,   12.  Februar   1999  Check:   2594
Name:  AQP1nuc.SEQ     Len:  807   Check:    8330     Weight:   1.00
Name:  AQP2nuc.SEQ     Len:  813   Check:    7220     Weight:   1.00
Name:  AQP3nuc.SEQ     Len:  855   Check:    7590     Weight:   1.00
Name:  AQP4nuc.SEQ     Len:  960   Check:    8696     Weight:   1.00
Name:  AQP5nuc.SEQ     Len:  795   Check:    758     Weight:   1.00
//
                1                                                            60
AQP1nuc.SEQ     ATGGCCAGCGAAATCAAGAAGAAGC.................TCTT              CT.....  ..
AQP2nuc.SEQ     ATGTG....GGAACTCAG.......................ATC               CAT....  ..
AQP3nuc.SEQ     ATG........AACC........GTTGCGGGG.AGATG.....CTC              C......  ..
AQP4nuc.SEQ     ATGAGTGACGGAGCTGCAGCGAGGCGGTGGGGTAAGTGTGGACCTC              CCTGCAG  CA
AQP5nuc.SEQ     ATGAAAAA.GGAGGTGTG.......................CTC               CCT....  ..

                61                                                           120
AQP1nuc.SEQ     GGC..TGTGGTGGCT.....GAGTTCCTGGCCATGA.CCCTCTTCG              .......  ..
AQP2nuc.SEQ     .................................AGCCTTCTCCC                GAGCAGT  GC
AQP3nuc.SEQ     .ACATCC.....GCTACCGG......CTG.........CTTCGCCA              ....GGC  TC
AQP4nuc.SEQ     AGCATCATGGTGGCTTTCAAAGGCGTCTGGACTCAAGCCTTCTGGA              AGGCGGT  CA
AQP5nuc.SEQ     .................................TGCCTTCTTCA                AGGCGGT  GT
```

Figura 6.5: Início do arquivo de exemplo `AQPDNA.MSF` que acompanha o pacote.

É possível desconsiderar algumas seqüências no processo de marcação do alinhamento, sem muito esforço, em um arquivo MSF. Com efeito, ao invés de ter que apagar cada linha referente às seqüências que devem ser desconsideradas, pode-se simplesmente designar tais seqüências, acrescentando um sinal de exclamação

(!) no início da linha que define seus respectivos nomes. Contudo, veremos que mesmo isso é desnecessário, pois existe um comando do TEXshade que permite definir seqüências a serem desconsideradas. O uso deste comando dispensa qualquer tipo de edição dos arquivos de entrada.

A Figura 6.6 ilustra um exemplo de arquivo MSF que contém o alinhamento de cinco seqüências, mas onde a segunda e a terceira estão marcadas para serem desconsideradas. No jargão técnico da informática, diz-se que essas seqüências encontram-se comentadas.

```
AQPDNA.MSF     MSF:  979    Type:  N  Freitag,   12.  Februar   1999  Check:   2594
Name:  AQP1nuc.SEQ     Len:  807   Check:    8330      Weight:   1.00
!Name:  AQP2nuc.SEQ     Len:  813   Check:    7220      Weight:   1.00
!Name:  AQP3nuc.SEQ     Len:  855   Check:    7590      Weight:   1.00
Name:  AQP4nuc.SEQ     Len:  960   Check:    8696      Weight:   1.00
Name:  AQP5nuc.SEQ     Len:  795   Check:    758      Weight:   1.00
//
               1                                                            60
AQP1nuc.SEQ     ATGGCCAGCGAAATCAAGAAGAAGC.................TCTT            CT.....  ..
AQP2nuc.SEQ     ATGTG....GGAACTCAG.......................ATC             CAT....  ..
AQP3nuc.SEQ     ATG........AACC........GTTGCGGGG.AGATG.....CTC            C......  ..
AQP4nuc.SEQ     ATGAGTGACGGAGCTGCAGCGAGGCGGTGGGGTAAGTGTGGACCTC            CCTGCAG  CA
AQP5nuc.SEQ     ATGAAAAA.GGAGGTGTG.......................CTC             CCT....  ..

               61                                                           120
AQP1nuc.SEQ     GGC..TGTGGTGGCT.....GAGTTCCTGGCCATGA.CCCTCTTCG            .......  ..
AQP2nuc.SEQ     ................................AGCCTTCTCCC               GAGCAGT  GC
AQP3nuc.SEQ     .ACATCC.....GCTACCGG......CTG.........CTTCGCCA             ....GGC  TC
AQP4nuc.SEQ     AGCATCATGGTGGCTTTCAAAGGCGTCTGGACTCAAGCCTTCTGGA            AGGCGGT  CA
AQP5nuc.SEQ     ................................TGCCTTCTTCA               AGGCGGT  GT
```

Figura 6.6: Exemplo de um arquivo MSF com seqüências comentadas. O comentário é definido com o ponto de exclamação (!) no início da linha.

O formato ALN é bastante similar ao formato MSF. Como principal diferença, deve-se destacar que arquivos ALN não explicitam o tipo das seqüências, de modo que o comando \seqtype {<tipo>} é fundamental nesse caso, com P para indicar peptídeos e N para indicar nucleotídeos. A Figura 6.7 ilustra um exemplo de arquivo ALN mínimo, que pode ser obtido com muitas ferramentas de alinhamento.

```
%%%%%%%%%%%%%%%%%%    %%%%%%%%%%%%%%%%%%    %%%%%%%%%%%%   %%%%%%% %%%%%% %%%%%%% %%
%%%%%                                                                          %%%%%
%%%%%  This  is  a  minimal   .ALN  file                                       %%%%%
%%%%%  --many   sequence   aligners   can  produce   them                      %%%%%
%%%%%                                                                          %%%%%
%%%%%%%%%%%%%%%%%%    %%%%%%%%%%%%%%%%%%    %%%%%%%%%%%%   %%%%%%% %%%%%% %%%%%%% %%

AQP2bt     SIAFSRAVLAEFLATLLFVFFGLGSALNWPQALPSVLQIAMAFGL             AIGTLV  QALGHVS  GA
AQP2cf     SVAFSRAVFAEFLATLLFVFFGLGSALNWPQALPSVLQIAMAFGL             GIGTLV  QALGHVS  GA
AQP2dd     SIAFSRAVFSEFLATLLFVFFGLGSALNWPQALPSVLQIAMAFGL             AIGTLV  QALGHIS  GA
AQP2ec     SIAFSRAVLAEFLATLLFVFFGLGSALNWPQAMPSVLQIAMAFGL             AIGTLV  QALGHVS  GA
AQP2em     SIAFSRAVFSEFLATLLFVFFGLGSALNWPQALPSVLQIAMAFGL             AIGTLV  QTLGHIS  GA

AQP2bt     HINPAVTVACLVGCHVSFLRAVFYVAAQLLGAVAGAALLHEITPP             AIRG
AQP2cf     HINPAVTVACLVGCHVSFLRAAFYVAAQLLGAVAGAALLHEITPP             HVRG
AQP2dd     HINPAVTVACLVGCHVSFLRATFYLAAQLLGAVAGAAILHEITPP             DIRG
AQP2ec     HINPAVTVACLVGCHVSFLRAAFYVAAQLLGAVAGAALLHEITPP             DIRR
AQP2em     HINPAVTVACLVGCHVSFLRATFYLAAQLLGAVAGAALLHELTPP             DIRG
```

Figura 6.7: Início do arquivo de exemplo `AQP2spec.ALN` que acompanha o pacote.

6.6 Utilizando o TEXshade

Para utilizar o TEXshade, primeiramente é necessário preparar um arquivo contendo um alinhamento múltiplo de seqüências em um dos formatos reconhecidos pelo TEXshade: MSF, ALN ou FASTA. O pacote traz três arquivos de exemplos de alinhamento que podem ser usados em testes iniciais: dois no formato MSF e um no formato ALN.

O arquivo `AQPDNA.MSF` traz um alinhamento de moléculas de DNA; o arquivo `AQPpro.MSF` fornece um exemplo de arquivo contendo um alinhamento de proteínas; o arquivo `AQP2spec.ALN` constitue-se em outro exemplo de arquivo de alinhamento protéico, porém, no formato ALN.

Para experimentar o TEXshade, basta criar um arquivo .tex contendo, no preâmbulo, a diretiva:

```
\usepackage{texshade}
```

Um arquivo mínimo usando o arquivo `AQPDNA.MSF` pode, então, ser escrito conforme ilustrado na Figura 6.8.

```
\documentclass{article}

\usepackage{texshade}

\begin{document}
   \begin{texshade}{AQPDNA.MSF}

   \end{texshade}
\end{document}
```

Figura 6.8: Exemplo de um arquivo mínimo a ser usado com o TEXshade.

No exemplo da Figura 6.8, todos os parâmetros do TEXshade assumem o valor padrão. O resultado é uma marcação no modo *identical*. Todos os resíduos idênticos em uma dada posição, desde que ocorram em número maior do que um dado percentual, são marcados em uma dada cor (na cor azul por padrão). Esse resultado está ilustrado na Figura 6.1.

A maioria dos comandos do TEXshade devem ser inseridos entre as diretivas `\begin{texshade}` e `\end{texshade}`. Mas algumas opções são configuradas através de parâmetros opcionais e não de comandos específicos. Pode-se, por exemplo, utilizar a opção *allmatchspecial* para requerer a marcação especial de resíduos idênticos que produzam um alinhamento de 100%, i.e., que ocorrem em todas as seqüências no mesmo ponto - Figura 6.2. Também é possível alterar as definições das cores ou fazer com que os resíduos alinhados, que não sejam idênticos, mas apresentem alguma similaridade, sejam marcados com uma outra cor.

No caso do modo *functional*, existem seis tipos pré-definidos (*charge, hydropath, structure, chemical, standart area, accessible area*) e o usuário pode facilmente definir outros tipos. A Figura 6.9 ilustra um exemplo de marcação no modo *functional* do tipo *hydropathy*. O resultado é o que aparece na Figura 6.3. Nesse caso, além da marcação dos resíduos idênticos, que ocorrem acima de um certo percentual, tem-se também a identificação de resíduos ácidos; resíduos básicos; resíduos não-polares e; resíduos polares não carregados. A exibição da legenda das cores é bastante interessante nesse caso e pode ser obtida simplesmente acrescentando-se o comando `\showlegend`.

```
\begin{texshade}{imagens/AQPpro_mini.MSF}
  \shadingmode[hydropathy]{functional}
  \showlegend
\end{texshade}
```

Figura 6.9: Código que gerou o resultado exibido na Figura 6.3.

Para favorecer visualizações rápidas, foi implementado o comando *fingerprint* que elimina os caracteres que representam os monômeros [Beitz (2000)]. Esse comando permite definir o número de resíduos por linha. Além disso, é possível modificar o esquema de cores para que sejam utilizados somente tons cinza ou somente preto e branco - útil para impressões de baixo custo. A Figura 6.10 mostra um exemplo de código que usa o esquema de cores *grays* e com o uso do *fingerprint* definindo que a impressão deve colocar 500 resíduos por linha. Com esse número elevado de resíduos por linha, as marcações assumem uma espessura bem fina, lembrando um padrão de código de barras. O resultado deste código está ilustrado na Figura 6.4.

```
\begin{texshade}{imagens/AQPpro_mini.MSF}
\shadingcolors{grays}
\fingerprint{500}
  \shadingmode[allmatchspecial]{similar}
  \showlegend
\end{texshade}
```

Figura 6.10: Código que gerou o resultado exibido na Figura 6.4.

O comando \killseq {<número-da-seqüência>} pode ser usado para determinar que a seqüência designada pelo número indicado deve ser desconsiderada na marcação. A numeração das seqüências inicia-se do numeral 1. O uso desse comando elimina a necessidade da edição do arquivo original, quando se deseja desconsiderar algumas das seqüências contidas no arquivo.

O comando \seqtype {<tipo>}, onde <tipo> pode assumir o valor P ou N, serve para indicar o tipo da seqüência: de Peptídeos ou de Nucleotídeos. Esse comando é fundamental quando se trata arquivos ALN de seqüências protéicas, uma vez que arquivos ALN não explicitam o tipo de arquivo e sendo que o padrão assumido pelo TEXshade é o de que as seqüências sejam de nucleotídeos.

Existem muitas outras possibilidades com o TEXshade. Para maiores informações, sugere-se a consulta da documentação oficial ([Beitz (2000)]).

Capítulo 7

Conclusão

Neste trabalho, foram apresentadas diferentes ferramentas para uso no estudo de alinhamento de seqüências de nucleotídeos ou de aminoácidos.

Dentre as ferramentas de *software*, analisou-se:

- Bancos de Dados públicos (disponíveis no NCBI ou no Swiss-Prot);
- O BLAST - *Basic Local Alignment Search Tool*;
- O ClustalW e o ClustalX e;
- O TEXshade.

O que essas ferramentas têm em comum, é o fato de serem distribuídas sob alguma forma de licença que garante a possibilidade de análise do código-fonte. Em alguns casos, a licença garante total liberdade na distribuição, mesmo para fins comerciais, desde que se mantenha a mesma liberdade nos novos produtos derivados. Em outros casos, a licença garante a liberdade para alteração e redistribuição da ferramenta, desde que sem fins lucrativos. Uma outra modalidade de licença permite a análise e mesmo a alteração do código, mas restringe que, em caso de alteração do mesmo, a ferramenta só pode ser redistribuída com a prévia autorização dos autores, mesmo sem fins lucrativos.

Outra característica comum das ferramentas analisadas é o fato de serem versões desenvolvidas para ambiente UNIX e similares. Todas foram instaladas e experimentadas sobre alguma distribuição GNU/Linux.

Evidenciou-se a grande facilidade na obtenção, instalação e configuração das ferramentas para ambientes GNU/Linux. Seja através de pacotes "*tarball*" ou pelo comando `apt-get` .

Conceitos de Biologia Celular e Biologia Molecular foram introduzidos e a aplicação destes conceitos foi ilustrada seguindo uma ordem didática.

Ao analisar os Bancos de Dados públicos, foi possível fixar diversos conceitos tais como: *gene*; *genoma*; *seqüências*; dentre outros. Além disso, evidenciou-se o grande valor e beleza inerentes no processo de trabalho colaborativo. Destacou-se também a grande importância das Novas Tecnologias de Informação e Comunicação como elementos facilitadores de todo o processo.

Na seqüência, o BLAST foi analisado aplicando-o no estudo de um alinhamento de um problema específico: a pesquisa sobre a similaridade entre genes de espécies completamente distintas. Este estudo, como demonstrado, abre portas para possíveis descobertas sobre a evolução dos seres vivos, bem como para a produção de fármacos mais eficientes e com menos danos aos pacientes ou à Natureza.

Enquanto o BLAST foi utilizado para o alinhamento de duas seqüências, o CLUSTAL foi apresentado como uma das principais ferramentas utilizadas no estudo de alinhamentos múltiplos de seqüências.

Concluindo a metodologia na pesquisa de similaridades entre seqüências, analisou-se uma ferramenta que aplica-se somente na marcação dos resultados já alinhados. O grande potencial da TEXshade na marcação dos dados foi mostrado, ainda que, não totalmente. Destacou-se o fato de que a TEXshade é uma ferramenta distribuída sob a GPL e totalmente baseada na LATEX.

Em momento algum, teve-se a pretensão de encerrar os tópicos aqui tratados. Espera-se sim que este trabalho possa ser útil em estudos introdutórios em Bioinformática ao mesmo tempo que estimule os novos pesquisadores dessa área do conhecimento científico, a desenvolverem cada vez códigos de *software* disponíveis para livre e total análise e reutilização. Esses desejos se inspiram na crença deste autor de que o total compartilhamento do conhecimento científico é a verdadeira chave para um desenvolvimento mais rápido, eficaz e coerente com as necessidades humanas.

Muitas outras ferramentas livres aplicadas à Bioinformática encontram-se facilmente disponíveis. [Prosdocimi et tal (2003)] e [Gibas & Jambeck (2001)] apresentam vastas listas de forma bastante didática. Como dica para trabalhos complementares, sugere-se uma abordagem comparativa entre diferentes ferramentas e implantações avaliando, performance e confiabilidade em diferentes tipos de estudos biológicos.

Referências Bibliográficas

[Anônimo] *Proteínas*
Disponível em http://agata.ucg.br/formularios/sites_docentes/zoo/joao/pdf/

[Alberts et al. (1999)] Alberts, B. et al. *Fundamentos da Biologia Celular - Uma Introdução à Biologia Molecular da Célula*. Artmed. Porto Alegre, RS. Brasil, 2001.

[Altschul et al. (1990)] Altschul, S.F., Gish, W., Miller, W., Myers, E.W., & Lipman, D.J. *Basic local alignment search tool*. *Journal of Molecular Biology*, 215:403-410, 1990. Disponível em: http://www.sciencedirect.com/ Encontrado via Entrez (http://www.ncbi.nlm.nih.gov/entrez/). Acessado em: 10 de setembro de 2004.

[Beitz (2000)] Beitz, Erick. *TEXshade: shading and labeling multiple sequence alignments using LATEX 2ε*. Bioinformatics: 16, 135-139.

[Brown (2002)] Brown, T. A. *Genomes*. 2nd ed. Oxford, UK: BIOS Scientific Publishers Ltd, 2002.

[Gibas & Jambeck (2001)] Gibas, Cynthia. & Jambeck, Per. *Desenvolvendo Bioinformática: Ferramentas de software para aplicações em Biologia*. Rio de Janeiro, Ed. Campus, 2001.

[hinegardner & Engelberg (1963)] Hinegardner, T.T. and J. Engelberg, *Rationale for a Universal Genetic Code*, Science, V. 142, 1963, 1083-1085.

[hinegardner & Engelberg (1964)] *Ibid Comment on a criticism by Woese*, V. 144, 1964, p. 1031.

[Higa (2001)] Higa, Roberto Hiroshi. *Entendendo e Interpretando os Parâmetros Utilizados por BLAST*. Instruções Técnicas do Ministério da Agricultura, Pecuária e Abastecimento / Embrapa, 2001.

[Lesk (2002)] Lesk, Arthur M. *Bioinformatics*. University of Cambridge. Oxford University Press, 2002.

[Leme (2002)] Leme, Rodrigo Mendes. *Sistema para Cruzamento de Dados Clínico-Genéticos da Viral Genetic Diversity Network (VGDN)* Trabalho de Formatura Supervisionado. Bacharelado em Ciência da Computação. USP, 2002. Disponível em: http://www.linux.ime.usp.br/ cef/mac499-02/monografias/ Acessado em: 10 de setembro de 2004.

[NCBI] *National Center for Biotechnology Information* - http://www.ncbi.nlm.nih.gov/ Acessado em: setembro de 2004.

[Oliveira] Oliveira, Carlos Jorge Rocha. *Aplicações Teóricas da Biologia Molecular/Engenharia Genética em Análises Clínicas* Apostila da disciplina Engenharia Genética do curso de Biomedicina. Faculdade de Ciências Biológicas e da Saúde. Universidade Metodista.

[Oliveira e Inoue (2002)] Oliveira, Guedes de Oliveira. & Inoue, Marcus Kiyoshi. *Laboratório de Bioinformática do Projeto Genoma Funcional e Diferencial do Fungo Paracoccidioides brasiliensis - Projeto Genoma Pb*. Trabalho de conclusão do curso de graduação em Ciência da Computação. Universidade de Brasília. Brasília - DF, 2002.

[Pearson (2001)] Pearson, William R. *Protein sequence comparison and Protein evolution*. Department of Biochemistry and Molecular Genetics. University of Virginia, USA. 2001.

[Prosdocimi et tal (2003)] Prosdocimi, Francisco et tal. *Bioinformática: Manual do Usuário*. Revista Biotecnologia Ciência & Desenvolvimento - n. 29 - janeiro, 2003.

[Rocha] Rocha, Eduardo. *Apostila do Módulo de Bioinformática - Análise de sequências*. Cadeira de Algorítmica e Programação. Universidade Estadual de Santa Cruz. Atelier de BioInformatique, Universidade de Paris & Institut Pasteur, Paris. Disponível em: http://labbi.uesc.br/apostilas/ Acessado em: 10 de setembro de 2004.

[Schneider (2002)] Schneider, Bruno de Oliveira. *Linguagens de Programação II*. Lavras: UFLA/FAEPE, 2002. (Curso de Pós Graduação "Lato Sensu" (Especialização) a Distância em Administração em Redes Linux).

[sciencedirect] *ScienceDirect* - http://www.sciencedirect.com/

[Santos & Queiroga (2003)] Santos, Fabrício R. & Queiroga, José Miguel. *Bioinformática aplicada à Genômica*. Manuscritos para capítulo do *Biowork IV*. Departamento de Biologia Geral, ICB, UFMG. Belo Horizonte, MG, 2003.

[Stothard (2000)] Stothard, Paul. *The sequence manipulation suite: JavaScript programs for analyzing and formatting protein and DNA sequences.* BioTechniques 28: 1102-1104. University of Alberta, Canada, 2000. Disponível em: http://www.biotechniques.com

[Stothard (2004)] Stothard, Paul. *The sequence manipulation suite 2: JavaScript programs for analyzing and formatting protein and DNA sequences.* University of Alberta, Canada, 2004. Disponível em: http://bioinformatics.org/sms2/ Acesso em: 10 mai. 2004.

[Swiss-Prot] *Swiss-Prot* http://www.expasy.ch/sprot/ Acessado em: setembro de 2004.

[Tekaia (1996)] Tekaia, Fredj. *Multiple Sequence Alignments - CLUSTALW tutorial* Disponível em: http://www.infobiogen.fr/docs/MAcours/clustalw.html Acessado em: 12 de setembro de 2004.

[Tisdall (2001)] Tisdall, James. *Beginning Perl for Bioinformatics*. O'Reilly & Associates Inc. 384 p. 2001.

[Walter (1999)] Walter, Maria Emília Machado Telles. *Algoritmos para Problemas em Rearranjo de Genomas.* Tese (Doutorado em Ciência da Computação) IC-UNICAMP, 1999.