1 Introduction

Biology has traditionally been an observational rather than a deductive science. Although recent developments have not altered this basic orientation, the nature of the data has radically changed. It is arguable that until recently all biological observations were fundamentally anecdotal – admittedly with varying degrees of precision, some very high indeed. However, in the last generation the data have become not only much more quantitative and precise, but, in the case of nucleotide and amino acid sequences, they have become *discrete*. It is possible to determine the genome sequence of an individual organism or clone not only

completely, but in principle *exactly*. Experimental error can never be avoided entirely, but for modern genomic sequencing it is extremely low.

Not that this has converted biology into a deductive science. Life does obey principles of physics and chemistry, but for now life is too complex, and too dependent on historical contingency, for us to deduce its detailed properties from basic principles.

A second obvious property of the data of bioinformatics is their *very very large amount*. Currently the nucleotide sequence databanks contain 6×10^9 bases (abbreviated 6 Mbp). If we use the approximate size of the human genome – 3×10^9 letters – as a unit, this amounts to two HUman Genome Equivalents (or 2 *huges*, an apt name). For a comprehensible standard of comparison, 1 *huge* is comparable to the number of characters appearing in six complete years of issues of *The New York Times*. The database of macromolecular structures contains 15 000 entries, the full three-dimensional coordinates of proteins, of average length ~400 residues. Not only are the individual databanks large, but their sizes are increasing at a very high rate. Figure 1.1 shows the growth over the past decade of GenBank (archiving nucleic acid sequences) and the Protein Data Bank (PDB) (archiving macromolecular structures). It would be precarious to extrapolate.

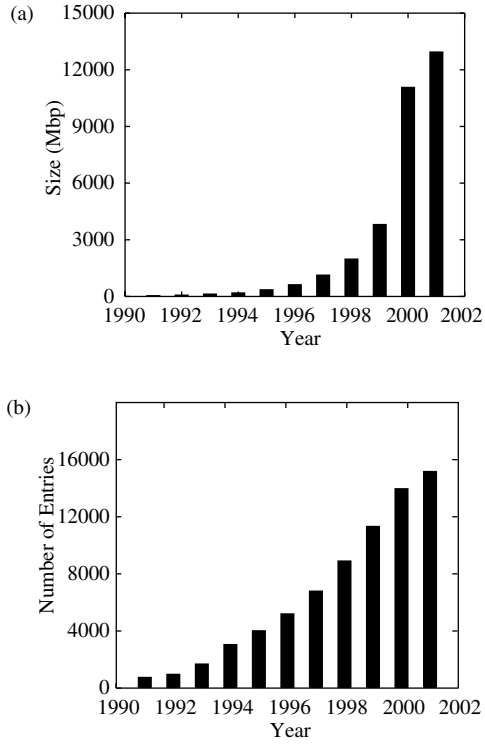


Fig. 1.1 (a) Growth of GenBank, the US National Center for Biotechnology Information genetic sequence archival databank. (b) Growth of Protein Data Bank, archive of three-dimensional biological macromolecular structures.

This quality and quantity of data have encouraged scientists to aim at commensurately ambitious goals:

- To have it said that they 'saw life clearly and saw it whole.' That is, to understand aspects of the biology of organisms, viewed as coherent complex *integrative* systems.
- To interrelate sequence, three-dimensional structure, interactions, and function of individual proteins, nucleic acids and protein–nucleic acid complexes.
- To use data on contemporary organisms as a basis for travel forward and backward in time – back to deduce events in evolutionary history, forward to greater deliberate scientific modification of biological systems.
- To support applications to medicine, agriculture and other scientific fields.

A scenario

For a fast introduction to the role of computing in molecular biology, imagine a crisis – sometime in the future – in which a new biological virus creates an epidemic of fatal disease in humans or animals. Laboratory scientists will isolate its genetic material – a molecule of nucleic acid consisting of a long polymer of four different types of residues – and determine the sequence. Computer programs will then take over.

Screening this new genome against a data bank of all known genetic messages will characterize the virus and reveal its relationship with viruses previously studied [10]. The analysis will continue, with the goal of developing antiviral therapies. Viruses contain protein molecules which are suitable targets, for drugs that will interfere with viral structure or function. Like the nucleic acids, the proteins are also linear polymers; the sequences of their residues, amino acids, are messages in a twenty-letter alphabet. From the viral DNA sequences, computer programs will derive the amino acid sequences of one or more viral proteins crucial for replication or assembly [01].

From the amino acid sequences, other programs will compute the structures of these proteins according to the basic principle that the amino acid sequences of proteins determine their three-dimensional structures, and thereby their functional properties. First, data banks will be screened for related proteins of known structure [15]; if any are found, the problem of structure prediction will be reduced to its 'differential form' – the prediction of the effects on a structure of *changes* in sequence – and the structures of the targets will be predicted by methods known as *homology modelling* [25]. If no related protein of known structure is found, and a viral protein appears genuinely new, the structure prediction must be done entirely *ab initio* [55]. This situation will arise ever more

infrequently as our databank of known structures grows more nearly complete, and our ability to detect distant relatives reliably grows more powerful.

Knowing the viral protein structure will make it possible to design therapeutic agents. Proteins have sites on their surfaces crucial for function, that are vulnerable to blocking. A small molecule, complementary in shape and charge distribution to such a site, will be identified or designed by a program to serve as an antiviral drug [50]; alternatively, one or more antibodies may be designed and synthesized to neutralize the virus [50].

This scenario is based on well-established principles, and I have no doubt that someday it will be implemented as described. One reason why we cannot apply it today against AIDS is that many of the problems are as yet unsolved. (Another is that viruses know how to defend themselves.) Computer scientists reading this book may have recognized that the numbers in square brackets are not literature citations, but follow the convention of D.E. Knuth in his classic books, *The Art of Computer Programming,* in indexing the difficulty of the problem! Numbers below 30 correspond to problems for which solutions already exist; higher numbers signal themes of current research.

Finally, it should be recognized that purely experimental approaches to the problem of developing antiviral agents may well continue to be more successful than theoretical ones for many years.

Life in space and time

It is difficult to define life, and it may be necessary to modify its definition – or to live, uncomfortably, with the old one – as computers grow in power and the silicon–life interface gains in intimacy. For now, try this: A biological organism is a naturally-occurring, self-reproducing device that effects controlled manipulations of matter, energy and information.

From the most distant perspective, life on Earth is a complex self-perpetuating system distributed in space and time. It is of the greatest significance that in many cases it is composed of *discrete* individual organisms, each with a finite lifetime and in most cases with unique features.

Spatially, starting far away and zooming in progressively, one can distinguish, within the biosphere, local *ecosystems,* which are stable until their environmental conditions change or they are invaded. Occupying each ecosystem are sets of *species,* which evolve by Darwinian selection or genetic drift. The generation of variants may arise from natural mutation, or the recombination of genes in sexual reproduction, or direct gene transfer. Each species is composed of *organisms* carrying out individual if not independent activities. Organisms are composed of *cells*. Every cell is an intimate localized ecosystem, not isolated from its environment but interacting with it in specific and controlled ways. Eukaryotic cells contain a complex internal structure of their own, including

nuclei and other subcellular organelles, and a cytoskeleton. And finally we come down to the level of molecules.

Life is extended not only in space but in time. We see today a snapshot of one stage in a history of life that extends back in time for at least 3.5 billion years. The theory of natural selection has been extremely successful in rationalizing the process of life's development. However, historical accident plays too dominant a role in determining the course of events to allow much detailed prediction. Nor does fossil DNA afford substantial access to any historical record at the molecular level. Instead, we must try to read the past in contemporary genomes. US Supreme Court Justice Felix Frankfurter once wrote that '... the American constitution is not just a document, it is a historical stream.' This is also true of genomes, which contain records of their own development.

Dogmas: central and peripheral

The information archive in each organism – the blueprint for potential development and activity of any individual – is the genetic material, DNA, or, in some viruses, RNA. DNA molecules are long, *linear,* chain molecules containing a message in a four-letter alphabet (see Box). Even for microorganisms the message is long, typically 10^6 characters. Implicit in the structure of the DNA are mechanisms for self-replication and for translation of genes into proteins. The double-helix, and its internal self-complementarity providing for accurate replication, are well known (see Plate I). Near-perfect replication is essential for stability of inheritance; but some imperfect replication, or mechanism for import of foreign genetic material, is also essential, else evolution could not take place in asexual organisms.

The strands in the double-helix are antiparallel; directions along each strand are named 3′ and 5′ (for positions in the deoxyribose ring). In translation to protein, the DNA sequence is always read in the $5' \rightarrow 3'$ direction.

The implementation of genetic information occurs, initially, through the synthesis of RNA and proteins. Proteins are the molecules responsible for much of the structure and activities of organisms. Our hair, muscle, digestive enzymes, receptors and antibodies are all proteins. Both nucleic acids and proteins are long, linear chain molecules. The genetic 'code' is in fact a cipher: Successive triplets of letters from the DNA sequence specify successive amino acids; stretches of DNA sequences encipher amino acid sequences of proteins. Typically, proteins are 200–400 amino acids long, requiring 600–1200 letters of the DNA message to specify them. Synthesis of RNA molecules, for instance the RNA components of the ribosome, are also directed by DNA sequences. However, in most organisms not all of the DNA is expressed as proteins or RNAs. Some regions of the DNA sequence are devoted to control mechanisms, and a substantial amount of

the genomes of higher organisms appears to be 'junk'. (Which in part may mean merely that we do not yet understand its function.)

The four naturally-occurring nucleotides in DNA (RNA)

a adenine g guanine c cytosine t thymine (u uracil)

The twenty naturally-occurring amino acids in proteins

Non-polar amino acids

G	glycine	A	alanine	P	proline	V	valine
I	isoleucine	L	leucine	F	phenylalanine	M	methionine

Polar amino acids

S	serine	C	cysteine	T	threonine	N	asparagine
Q	glutamine	H	histidine	Y	tyrosine	W	tryptophan

Charged amino acids

D aspartic acid E glutamic acid K lysine R arginine

Other classifications of amino acids can also be useful. For instance, histidine, phenylalanine, tyrosine, and tryptophan are aromatic, and are observed to play special structural roles in membrane proteins.

Amino acid names are frequently abbreviated to their first three letters, for instance Gly for glycine; except for asparagine, glutamine and tryptophan, which are abbreviated to Asn, Gln and Trp, respectively. The rare amino acid selenocysteine has the three-letter abbreviation Sec and the one-letter code U.

It is conventional to write nucleotides in lower case and amino acids in upper case. Thus atg = adenine-thymine-guanine and ATG = alanine-threonine-glycine.

In DNA the molecules comprising the alphabet are chemically similar, and the structure of DNA is, to a first approximation, uniform. Proteins, in contrast, show a great variety of three-dimensional conformations. These are necessary to support their very diverse structural and functional roles.

The amino acid sequence of a protein dictates its three-dimensional structure. For each natural amino acid sequence, there is a unique stable *native state* that under proper conditions is adopted spontaneously. If a purified protein is heated, or otherwise brought to conditions far from the normal physiological environment, it will 'unfold' to a disordered and biologically-inactive structure. (This is why our bodies contain mechanisms to maintain nearly-constant internal conditions.) When normal conditions are restored, protein molecules will generally readopt the native structure, indistinguishable from the original state.

The spontaneous folding of proteins to form their native states is the point at which Nature makes the giant leap from the one-dimensional world of genetic

The standard genetic code

ttt	Phe	tct	Ser	tat	Tyr	tgt	Cys
ttc	Phe	tcc	Ser	tac	Tyr	tgc	Cys
tta	Leu	tca	Ser	taa	STOP	tga	STOP
ttg	Leu	tcg	Ser	tag	STOP	tgg	Trp
ctt	Leu	cct	Pro	cat	His	cgt	Arg
ctc	Leu	ccc	Pro	cac	His	cgc	Arg
cta	Leu	cca	Pro	caa	Gln	cga	Arg
ctg	Leu	ccg	Pro	caa	Gln	cga	Arg
att	Ile	act	Thr	aat	Asn	agt	Ser
atc	Ile	acc	Thr	aac	Asn	agc	Ser
ata	Ile	aca	Thr	aaa	Lys	aga	Arg
atg	Met	acg	Thr	aag	Lys	agg	Arg
gtt	Val	gct	Ala	gat	Asp	ggt	Gly
gtc	Val	gcc	Ala	gac	Asp	ggc	Gly
gta	Val	gca	Ala	gaa	Glu	gga	Gly
gtg	Val	gcg	Ala	gag	Glu	ggg	Gly

Alternative genetic codes appear, for example, in organelles – chloroplasts and mitochondria.

and protein sequences to the three-dimensional world we inhabit. There is a paradox: The translation of DNA sequences to amino acid sequences is very simple to describe logically; it is specified by the genetic code. The folding of the polypeptide chain into a precise three-dimensional structure is very difficult to describe logically. However, translation requires the immensely complicated machinery of the ribosome, tRNAs and associated molecules; but protein folding occurs spontaneously.

The functions of proteins depend on their adopting the native three-dimensional structure. For example, the native structure of an enzyme may have a cavity on its surface that binds a small molecule and juxtaposes it to catalytic residues. We thus have the paradigm:

- DNA sequence determines protein sequence
- Protein sequence determines protein structure
- Protein structure determines protein function

Most of the organized activity of bioinformatics has been focused on the analysis of the data related to these processes.

So far, this paradigm does not include levels higher than the molecular level of structure and organization, including, for example, such questions as how tissues become specialized during development or, more generally, how environmental

effects exert control over genetic events. In some cases of simple feedback loops, it is understood at the molecular level how increasing the amount of a reactant causes an increase in the production of an enzyme that catalyses its transformation. More complex are the programs of development during the lifetime of an organism. These fascinating problems about the information flow and control within an organism have now come within the scope of mainstream bioinformatics.

Observables and data archives

A databank includes an archive of information, a logical organization or 'structure' of that information, and tools to gain access to it. Databanks in molecular biology cover nucleic acid and protein sequences, macromolecular structures, and function. They include:

- Archival databanks of biological information
  - DNA and protein sequences, including annotation
  - Nucleic acid and protein structures, including annotation
  - Databanks of protein expression patterns
- Derived databanks: These contain information collected from the archival databanks, and from the analysis of their contents. For instance:
  - sequence motifs (characteristic 'signature patterns' of families of proteins)
  - mutations and variants in DNA and protein sequences
  - classifications or relationships (connections and common features of entries in archives; for instance a databank of sets of protein sequence families, or a hierarchical classification of protein folding patterns)
- Bibliographic databanks
- Databanks of web sites
  - databanks of databanks containing biological information
  - links between databanks

Database queries seek to identify a set of entries (e.g. sequences or structures) on the basis of specified features or characteristics, or on the basis of similarity to a probe sequence or structure. The most common query is: 'I have determined a new sequence, or structure – what do the databanks contain that is like it?' Once a set of sequences or structures similar to the probe object is fished out of the appropriate database, the researcher is in a position to identify and investigate their common features.

The mechanism of access to a databank is the set of tools for answering questions such as:

- 'Does the databank contain the information I require?' (Example: In which databanks can I find amino acid sequences for alcohol dehydrogenases?)

- 'How can I assemble selected information from the databank in a useful form?' (Example: How can I compile a list of globin sequences, or even better, a table of aligned globin sequences?)
- Indices of databanks are useful in asking 'Where can I find some specific piece of information?' (Example: What databanks contain the amino acid sequence of porcupine trypsin?) Of course if I know and can specify exactly what I want the problem is relatively straightforward.

A databank without effective modes of access is merely a data graveyard. How to achieve effective access is an issue of database design that ideally should remain hidden from users. It has become clear that effective access cannot be provided by bolting a query system onto an unstructured archive. Instead, the logical organization of the storage of the information must be designed with the access in mind – what kinds of questions users will want to ask – and the structure of the archive must mesh smoothly with the information-retrieval software.

A variety of possible kinds of database queries can arise in bioinformatics. These include:

(1) Given a sequence, or fragment of a sequence, find sequences in the database that are similar to it. This is a central problem in bioinformatics. We share such string-matching problems with many fields of computer science. For instance, word processing and editing programs support string-search functions.

(2) Given a protein structure, or fragment, find protein structures in the database that are similar to it. This is the generalization of the string matching problem to three dimensions.

(3) Given a sequence of a protein of unknown structure, find *structures* in the database that adopt similar three-dimensional structures. One is tempted to cheat – to look in the sequence data banks for proteins with sequences similar to the probe sequence: For if two proteins have sufficiently similar sequences, they will have similar structures. However, the converse is not true, and one can hope to create more powerful search techniques that will find proteins of similar structure even though their sequences have diverged beyond the point where they can be recognized as similar by sequence comparison.

(4) Given a protein structure, find *sequences* in the data bank that correspond to similar structures. Again, one can cheat by using the structure to probe a structure data bank, but this can give only limited success because there are so many more sequences known than structures. It is, therefore, desirable to have a method that can pick out the structure from the sequence.

(1) and (2) are solved problems; such searches are carried out thousands of times a day. (3) and (4) are active fields of research.

Tasks of even greater subtlety arise when one wishes to study relationships between information contained in separate databanks. This requires links that facilitate simultaneous access to several databanks. Here is an example: 'For which proteins of known structure involved in diseases of purine biosynthesis in humans, are there related proteins in yeast?' We are setting conditions on: known structure, specified function, detection of relatedness, correlation with disease, specified species. The growing importance of simultaneous access to databanks has led to research in databank interactivity – how can databanks 'talk to one another', without too great a sacrifice of the freedom of each one to structure its own data in ways appropriate to the individual features of the material it contains.

A problem that has not yet arisen in molecular biology is control of updates to the archives. A database of airline reservations must prevent many agents from selling the same seat to different travellers. In bioinformatics, users can read or extract information from archival databanks, or submit material for processing by the staff of an archive, but not add or alter entries directly. This situation may change. From a practical point of view, the amount of data being generated is increasing so rapidly as to swamp the ability of archive projects to assimilate it. There is already movement towards greater involvement of scientists at the bench in preparing data for the archive.

Although there are good arguments for unique control over the archives, there is no need to limit the ways to access them – colloquially, the design of 'front ends'. Specialized user communities may extract subsets of the data, or recombine data from different sources, and provide specialized avenues of access. Such 'Boutique databases' depend on the primary archives as the source of the information they contain, but redesign the organization and presentation as they see fit. Indeed, different derived databases can slice and dice the same information in different ways. A reasonable extrapolation suggests the concept of specialized 'virtual databases', grounded in the archives but providing individual scope and function, tailored to the needs of individual research groups or even individual scientists.

Curation, annotation, and quality control

The scientific and medical communities are dependent on the quality of databanks. Indices of quality, even if they do not permit correction of mistakes, may help us avoid arriving at wrong conclusions.

Databank entries comprise raw experimental results, and supplementary information, or annotations. Each of these has its own sources of error.

The most important determinant of the quality of the data themselves is the state of the art of the experiments. Older data were limited by older techniques; for instance, amino acid sequences of proteins used to be determined by peptide sequencing, but almost all are now translated from DNA sequences. One consequence of the data explosion is that most data are new data, governed by current technology, which in most cases does quite a good job.

Annotations include information about the source of the data and the methods used to determine them. They identify the investigators responsible, and cite relevant publications. They provide links to related information in other databanks. In sequence databanks, annotations include *feature tables:* lists of segments of the sequences that have biological significance – for instance, regions of a DNA sequence that code for proteins. These appear in computer-parsable formats, and their contents may be restricted to a controlled vocabulary.

Until recently, a typical DNA sequence entry was produced by a single research group, investigating a gene and its products in a coherent way. Annotations were grounded in experimental data and written by specialists. In contrast, full-genome sequencing projects offer no experimental confirmation of the expression of most putative genes, nor characterization of their products. Curators at databanks base their annotations on the analysis of the sequences by computer programs.

Annotation is the weakest component of the genomics enterprise. Automation of annotation is possible only to a limited extent; getting it right remains labour-intensive, and allocated resources are inadequate. But the importance of proper annotation cannot be underestimated. P. Bork has commented that errors in gene assignments vitiate the high quality of the sequence data themselves.

Growth of genomic data will permit improvement in the quality of annotation, as statistical methods increase in accuracy. This will allow improved *reannotation* of entries. The improvement of annotations will be a good thing. But the inevitable concomitant, that annotation will be in flux, is disturbing. Will completed research projects have to be revisited periodically, and conclusions reconsidered? The problem is aggravated by the proliferation of web sites, with increasingly dense networks of links. They provide useful avenues for applications. But the web is also a vector of contagion, propagating errors in raw data, in immature data subsequently corrected but the corrections not passed on, and variant annotations.

The only possible solution is a *distributed* and *dynamic* error-correction and annotation process. Distributed, in that databank staff will have neither the time nor the expertise for the job; specialists will have to act as curators. Dynamic, in that progress in automation of annotation and error identification/correction will permit reannotation of databanks. We will have to give up the safe idea of a stable databank composed of entries that are correct when first distributed and stay fixed. Databanks will become a seething broth of information growing in size, and maturing – we must hope – in quality.

The World Wide Web

It is likely that all readers will have used the World Wide Web, for reference material, for news, for access to databases in molecular biology, for checking out personal information about individuals – friends or colleagues or celebrities

– or just for browsing. Fundamentally, the Web is a means of interpersonal and intercomputer contact over networks. It provides a complete global village, containing the equivalent of library, post office, shops, and schools.

You, the user, run a browser program on your own computer. Common browsers are Netscape and Internet Explorer. With these browser programs you can read and display material from all over the world. A browser also presents control information, allowing you to follow trails forward and back or to interrupt a side-trip. And it allows you to download information to your local computer.

The material displayed contains embedded links that allow you to jump around to other pages and sites, adding new dimensions to your excursions. The interconnections animate the Web. What makes the human brain so special is not the absolute number of neurons, but the density of interconnections between them. Similarly, it is not only the number of entries that makes the Web so powerful, but their reticulation.

The links are visible in the material you are viewing at any time. Running your browser program, you view a page, or frame. This view will contain active objects: words, or buttons, or pictures. These are usually distinguished by a highlighted colour. Selecting them will effect a transfer to a new page. At the same time, you automatically leave a trail of 'electronic breadcrumbs' so that you can return to the calling link, to take up further perusal of the page you started from.

The Web can be thought of as a giant world wide bulletin board. It contains text, images, movies, and sounds. Virtually anything that can be stored on a computer can be made available and accessed via the Web. An interesting example is a page describing the poetry of William Butler Yeats. The highest level page contains material appropriate for a table of contents. Via links displayed on this top page, you can see printed text of different poems. You can compare different editions. You can access critical analysis of the poems. You can see versions of some poems in Yeats' manuscripts. For some poems, there is even a link to an audio file, from which you can hear Yeats himself reading the poem.

Links can be internal or external. Internal links may take you to other portions of the text of a current document, or to images, movies, or sounds. External links may allow you to move *down* to more specialized documents, *up* to more general ones (perhaps providing background to technical material), *sideways* to parallel documents (other papers on the same subject), or *over*, to directories that show what other relevant material is available.

The main thing to do, to get started using the Web effectively, is to find useful entry points. Once a session is launched, the links will take you where you want to go. Among the most important sites are *search engines* that index the entire Web and permit retrieval by keywords. You can enter one or more terms, such as 'phosphorylase', 'allosteric change', 'crystal structure', and the search program will return a list of links to sites on the Web that contain these terms. You will thereby identify sites relevant to your interest.

Once you have completed a successful session, when you next log in the inter-session memory facilities of the browsers allow you to pick up cleanly where you left off. During any session, when you find yourself viewing a document to which you will want to return, you can save the link in a file of *bookmarks* or *favourites*. In a subsequent session you can return to any site on this list directly, not needing to follow the trail of links that led you there in the first place.

A personal *home page* is a short autobiographical sketch (with links, of course). Your professional colleagues will have their own home pages which typically include name, institutional affiliation, addresses for paper and electronic mail, telephone and fax numbers, a list of publications and current research interests. It is not uncommon for home pages to include personal information, such as hobbies, pictures of the individual with his or her spouse and children, and even the family dog!

Nor is the Web solely a one-way street. Many Web documents include forms in which you can enter information, and launch a program that returns results within your session. Search engines are common examples. Many calculations in bioinformatics are now launched via such web servers. If the calculations are lengthy the results may not be returned within the session, but sent by e-mail.

The hURLy-bURLy

Even brief experience with the Web will bring you into contact with the strange-looking character strings that identify locations. These are URLs – *Uniform Resource Locators*. They specify the format of the material and its location. After all, every document on the Web must be a file on some computer somewhere. An example of a URL is:

```
http://www.lib.berkeley.edu/TeachingLib/Guides/Internet/FindInfo.html
```

This is the URL of a useful tutorial about Finding Information on the Internet. The prefix `http://` stands for hypertext transfer protocol. This tells your browser to expect the document in `http` format, by far the most common one. The next section, `www.lib.berkeley.edu` is the name of a computer, in this case in the central library at the University of California at Berkeley. The rest of the URL specifies the location and name of the file on the computer the contents of which your browser will display.

Electronic publication

More and more publications are appearing on the Web. A scientific journal may post only its table of contents, a table of contents together with abstracts of articles, or even full articles. Many institutional publications – newsletters and technical reports – appear on the Web. Many other magazines and newspapers are showing up as well. You might want to try `http://www.nytimes.com` Many printed publications now contain references to Web links containing supplementary material that never appears on paper.

We are in an era of a transition to paper-free publishing. It is already a good idea to include, in your own printed articles, your e-mail address and the URL of your home page.

Electronic publication raises a number of questions. One concerns peer review. How can we guarantee the same quality for electronic publication that we rely on for printed journals? Are electronic publications 'counted', in the publish-or-perish sense of judging the productivity (if not the quality) of a job candidate? A well-known observer of the field has offered the sobering (if possibly exaggerated) prediction: 'The first time Harvard or Stanford gives tenure to someone for electronic publication, 90% of the scientific journals will disappear overnight'.

Computers and computer science

Bioinformatics would not be possible without advances in computing hardware and software. Fast and high-capacity storage media are essential even to maintain the archives. Information retrieval and analysis require programs; some fairly straightforward and others extremely sophisticated. Distribution of the information requires the facilities of computer networks and the World Wide Web.

Computer science is a young and flourishing field with the goal of making most effective use of information technology hardware. Certain areas of theoretical computer science impinge most directly on bioinformatics. Let us consider them with reference to a specific biological problem: 'Retrieve from a database all sequences similar to a probe sequence'. A good solution of this problem would appeal to computer science for:

- *Analysis of algorithms. An algorithm is a complete and precise specification of a method for solving a problem*. For the retrieval of similar sequences, we need to measure the similarity of the probe sequence to every sequence in the database. It is possible to do much better than the naive approach of checking every pair of positions in every possible juxtaposition, a method that even without allowing gaps would require a time proportional to the product of the number of characters in the probe sequence times the number of characters in the database. A speciality in computer science known colloquially as 'stringology' focuses on developing efficient methods for this type of problem, and analysing their effective performance.

- *Data structures, and information retrieval.* How can we organize our data for efficient response to queries? For instance, are there ways to index or otherwise 'preprocess' the data to make our sequence-similarity searches more efficient? How can we provide interfaces that will assist the user in framing and executing queries?

- *Software engineering*. Hardly ever anymore does anyone write programs in the native language of computers. Programmers work in higher-level

languages, such as C, C++, PERL ('Practical Extraction and Report Language') or even FORTRAN. The choice of programming language depends on the nature of the algorithm and associated data structure, and the expected use of the program. Of course most complicated software used in bioinformatics is now written by specialists. Which brings up the question of how much programming expertise a bioinformatician needs.

Programming

Programming is to computer science what bricklaying is to architecture. Both are creative; one is an art and the other a craft.

Many students of bioinformatics ask whether it is essential to learn to write complicated computer programmes. My advice (not agreed upon by everyone in the field) is: 'Don't. Unless you want to specialize in it'. For working in bioinformatics, you will need to develop expertise in using tools available on the Web. Learning how to create and maintain a web site is essential. And of course you will need facility in the use of the operating system of your computer. Some skill in writing simple scripts in a language like PERL provides an essential extension to the basic facilities of the operating system.

On the other hand, the size of the data archives, and the growing sophistication of the questions we wish to address, demand a healthy respect. Truly creative programming in the field is best left to specialists, well-trained in computer science. Nor does *using* programs, via highly polished (not to say flashy) Web interfaces, provide any indication of the nature of the activity involved in writing and debugging programs. Bismarck once said: 'Those who love sausages or the law should not watch either being made'. Perhaps computer programs should be added to his list.

I recommend learning some basic skills with PERL. PERL is a very powerful tool. It makes it very easy to carry out many very useful simple tasks. PERL also has the advantage of being available on most computer systems.

How should you learn enough PERL to be useful in bioinformatics? Many institutions run courses. Learning from colleagues is fine, depending on the ratio of your adeptness to their patience. Books are available. A very useful approach is to find lessons on the Web – ask a search engine for 'PERL tutorial' and you will turn up many useful sites that will lead you by the hand through the basics. And of course use it in your work as much as you can. This book will not teach you PERL, but it will provide opportunities to practice what you learn elsewhere.

Examples of *simple* PERL programs appear in this book. The strength of PERL at character-string handling make it suitable for sequence analysis tasks in biology. Here is a very simple PERL program to translate a nucleotide sequence into an amino acid sequence according to the standard genetic code. The first line, `#!/usr/bin/perl,` is a signal to the UNIX (or LINUX) operating system that what follows is a PERL program. Within the program, all text commencing with a `#`, through to the end of the line on which it appears, is merely comment. The line `__END__` signals that the program is finished and what follows is the

```
#!/usr/bin/perl
#translate.pl -- translate nucleic acid sequence to protein sequence
#               according to standard genetic code

#   set up table of standard genetic code

%standardgeneticcode = (
 "ttt"=> "Phe",   "tct"=> "Ser", "tat"=> "Tyr",   "tgt"=> "Cys",
 "ttc"=> "Phe",   "tcc"=> "Ser", "tac"=> "Tyr",   "tgc"=> "Cys",
 "tta"=> "Leu",   "tca"=> "Ser", "taa"=> "TER",   "tga"=> "TER",
 "ttg"=> "Leu",   "tcg"=> "Ser", "tag"=> "TER",   "tgg"=> "Trp",
 "ctt"=> "Leu",   "cct"=> "Pro", "cat"=> "His",   "cgt"=> "Arg",
 "ctc"=> "Leu",   "ccc"=> "Pro", "cac"=> "His",   "cgc"=> "Arg",
 "cta"=> "Leu",   "cca"=> "Pro", "caa"=> "Gln",   "cga"=> "Arg",
 "ctg"=> "Leu",   "ccg"=> "Pro", "cag"=> "Gln",   "cgg"=> "Arg",
 "att"=> "Ile",   "act"=> "Thr", "aat"=> "Asn",   "agt"=> "Ser",
 "atc"=> "Ile",   "acc"=> "Thr", "aac"=> "Asn",   "agc"=> "Ser",
 "ata"=> "Ile",   "aca"=> "Thr", "aaa"=> "Lys",   "aga"=> "Arg",
 "atg"=> "Met",   "acg"=> "Thr", "aag"=> "Lys",   "agg"=> "Arg",
 "gtt"=> "Val",   "gct"=> "Ala", "gat"=> "Asp",   "ggt"=> "Gly",
 "gtc"=> "Val",   "gcc"=> "Ala", "gac"=> "Asp",   "ggc"=> "Gly",
 "gta"=> "Val",   "gca"=> "Ala", "gaa"=> "Glu",   "gga"=> "Gly",
 "gtg"=> "Val",   "gcg"=> "Ala", "gag"=> "Glu",   "ggg"=> "Gly"
);


#   process input data

while ($line = <DATA>) {                                # read in line of input
    print "$line";                                      # transcribe to output
    chop();                                             # remove end-of-line character
    @triplets = unpack("a3" x (length($line)/3), $line); # pull out successive triplets
    foreach $codon (@triplets) {                        # loop over triplets
        print "$standardgeneticcode{$codon}";           # print out translation of each
    }                                                   # end loop on triplets
    print "\n\n";                                       # skip line on output
}                                                       # end loop on input lines

#   what follows is input data

__END__
atgcatccctttaat
tctgtctga
```

Running this program on the given input data produces the output:

```
atgcatccctttaat
MetHisProPheAsn

tctgtctga
SerValTER
```

input data. (All material that the reader might find useful to have in computer-readable form, including all programs, appear in the web site associated with this book: `http://www.oup.com/uk/lesk/bioinf`)

Even this simple program displays several features of the PERL language. The file contains background data (the genetic code translation table), statements that tell the computer to do something with the input (i.e. the sequence to be translated), and the input data (appearing after the `__END__` line). Comments summarize sections of the program, and also describe the effect of each statement.

The program is structured as blocks enclosed in curly brackets: {. . .}, which are useful in controlling the flow of execution. Within blocks, individual statements

(each ending in a ;) are executed in order of appearance. The outer block is a *loop:*

```
while ($line = <DATA>) {
        ...
}
```

<DATA> refers to the lines of input data (appearing after the __END__). The block is executed once for each line of input; that is, `while` there is any line of input remaining.

Three types of data structures appear in the program. The line of input data, referred to as `$line`, is a simple *character string*. It is split into an *array* or vector of triplets. An array stores several items in a linear order, and individual items of data can be retrieved from their positions in the array. For ease of looking up the amino acid coded for by any triplet, the genetic code is stored as an *associative array*. An associative array, or hash table, is a generalization of a simple or sequential array. If the elements of a simple array are indexed by consecutive integers, the elements of an associative array are indexed by *any* character strings, in this case the 64 triplets. We utilize the input triplets *in order of their appearance* in the nucleotide sequence, but we need to access the elements of the genetic code table *in an arbitrary order* as dictated by the succession of triplets. A simple array or vector of character strings is appropriate for processing successive triplets, and the associative array is appropriate for looking up the amino acids that correspond to them.

Here is another PERL program, that illustrates additional aspects of the language.* This program reassembles the sentence:

All the world's a stage,
And all the men and women merely players;
They have their exits and their entrances,
And one man in his time plays many parts.

after it has been chopped into random overlapping fragments (\n in the fragments represents end-of-line in the original):

the men and women merely players;\n
one man in his time
All the world's
their entrances,\nand one man
stage,\nAnd all the men and women
They have their exits and their entrances,\n
world's a stage,\nAnd all
their entrances,\nand one man
in his time plays many parts.
merely players;\nThey have

*This section may be skipped on a first reading.

```
#!/usr/bin/perl
#assemble.pl -- assemble overlapping fragments of strings

#  input of fragments
while ($line = <DATA>) {                #   read in fragments, 1 per line
   chop($line);                         #   remove trailing carriage return
   push(@fragments,$line);              #   copy each fragment into array
}
#  now array  @fragments  contains fragments

#  we need two relationships between fragments:
#  (1) which fragment shares no prefix with suffix of another fragment
#      * This tells us which fragment comes first
#  (2) which fragment shares longest suffix with a prefix of another
#      * This tells us which fragment  follows  any fragment

#  First set array of prefixes to the default value   "noprefixfound".
#      Later, change this default value when a prefix is found.
#      The one fragment that retains the default value must be come first.

#  Then loop over pairs of fragments to determine maximal overlap.
#      This determines successor of each fragment
#      Note in passing that if a fragment has a successor then the
#          successor must have a prefix

foreach $i (@fragments) {               #   initially set  prefix of each fragment
    $prefix{$i} = "noprefixfound";      #       to  "noprefixfound"
    }                                   #   this will be overwritten when a prefix is found

#  for each pair, find longest overlap of suffix of one with prefix of the other
#       This tells us which fragment FOLLOWS any fragment

foreach $i (@fragments) {               #   loop over fragments
    $longestsuffix = "";                #   initialize longest suffix to null

    foreach $j (@fragments) {           #   loop over fragment pairs
        unless ($i eq $j) {             #   don't check fragment against itself

            $combine = $i . "XXX" . $j;  #   concatenate fragments, with fence XXX
            $combine =~ /([\S ]{2,})XXX\1/;   #   check for repeated sequence
            if (length($1) > length($longestsuffix)) {    # keep longest overlap
                $longestsuffix = $1;     #   retain longest suffix
                $successor{$i} = $j;     #   record that $j follows $i
            }

        }
    }
    $prefix{$successor{$i}} = "found";   #  if $j follows $i then $j must have a prefix
}


foreach (@fragments) {                  #  find fragment that has no prefix; that's the start
    if ($prefix{$_} eq "noprefixfound") {$outstring = $_;}
}

$test = $outstring;                     #   start with fragment without prefix
while ($successor{$test}) {             #   append fragments in order
     $test = $successor{$test};         #   choose next fragment
     $outstring = $outstring . "XXX" . $test;  # append to string
     $outstring =~ s/([\S ]+)XXX\1/\1/;  #   remove overlapping segment
}

$outstring =~ s/\\n/\n/g;               #   change signal \n to real carriage return
print "$outstring\n";                   #   print final result

__END__
the men and women merely players;\n
one man in his time
All the world's
their entrances,\nand one man
stage,\nAnd all the men and women
They have their exits and their entrances,\n
world's a stage,\nAnd all
their entrances,\nand one man
in his time plays many parts.
merely players;\nThey have
```

This kind of calculation is important in assembling DNA sequences from overlapping fragments.

Should your programming ambitions go beyond simple tasks, check out the Bioperl Project, a source of freely available PERL programs and components in the field of bioinformatics (See: `http://bio.perl.org/`).

Biological classification and nomenclature

Back to the eighteenth century, when academic life at least was in some respects simpler.

Biological nomenclature is based on the idea that living things are divided into units called species – groups of similar organisms with a common gene pool. (Why living things should be 'quantized' into *discrete* species is a very complicated question.) Linnaeus, a Swedish naturalist, classified living things according to a hierarchy: Kingdom, Phylum, Class, Order, Family, Genus and Species (see Box). Modern taxonomists have added additional levels. For identification it generally suffices to specify the *binomial:* Genus and Species; for instance *Homo sapiens* for human or *Drosophila melanogaster* for fruit fly. Each binomial uniquely specifies a species that may also be known by a common name; for instance, *Bos taurus* = cow. Of course, most species have no common names.

Originally the Linnaean system was only a classification based on observed similarities. With the discovery of evolution it emerged that the system largely reflects biological ancestry. The question of which similarities truly reflect common ancestry must now be faced. Characteristics derived from a common ancestor are called *homologous;* for instance an eagle's wing and a human's arm. Other apparently similar characteristics may have arisen independently by *convergent evolution;* for instance, an eagle's wing and a bee's wing: The most recent common ancestor of eagles and bees did not have wings. Conversely, truly homologous characters may have diverged to become very dissimilar in structure

Classifications of Human and Fruit fly

	Human	Fruit fly
Kingdom	Animalia	Animalia
Phylum	Chordata	Arthropoda
Class	Mammalia	Insecta
Order	Primata	Diptera
Family	Hominidae	Drosophilidae
Genus	*Homo*	*Drosophila*
Species	*sapiens*	*melanogaster*

and function. The bones of the human middle ear are homologous to bones in the jaws of primitive fishes; our eustachian tubes are homologues of gill slits. In most cases experts can distinguish true homologies from similarities resulting from convergent evolution.

Sequence analysis gives the most unambiguous evidence for the relationships among species. The system works well for higher organisms, for which sequence analysis and the classical tools of comparative anatomy, palaeontology and embryology usually give a consistent picture. Classification of microorganisms is more difficult, partly because it is less obvious how to select the features on which to classify them and partly because a large amount of lateral gene transfer threatens to overturn the picture entirely.

Ribosomal RNAs turned out to have the essential feature of being present in all organisms, with the right degree of divergence. (Too much or too little divergence and relationships become invisible.)

On the basis of 15S ribosomal RNAs, C. Woese divided living things most fundamentally into three Domains (a level *above* Kingdom in the hierarchy): Bacteria, Archaea and Eukarya (see Fig. 1.2). Bacteria and archaea are prokaryotes; their cells do not contain nuclei. Bacteria include the typical microorganisms responsible for many infectious diseases, and, of course, *Escherichia coli*, the mainstay of molecular biology. Archaea comprise extreme thermophiles and halophiles, sulphate reducers and methanogens. We ourselves are Eukarya – organisms containing cells with nuclei, including yeast and all multicellular organisms.

A census of the species with sequenced genomes reveals emphasis on bacteria, because of their clinical importance, and for the relative ease of sequencing genomes of prokaryotes. However, fundamentally we may have more to learn about ourselves from archaea than from bacteria. For despite the obvious differences in lifestyle, and the absence of a nucleus, archaea are in some ways more closely related on a molecular level to eukarya than to bacteria. It is also likely that the archaea are the closest living organisms to the root of the tree of life.

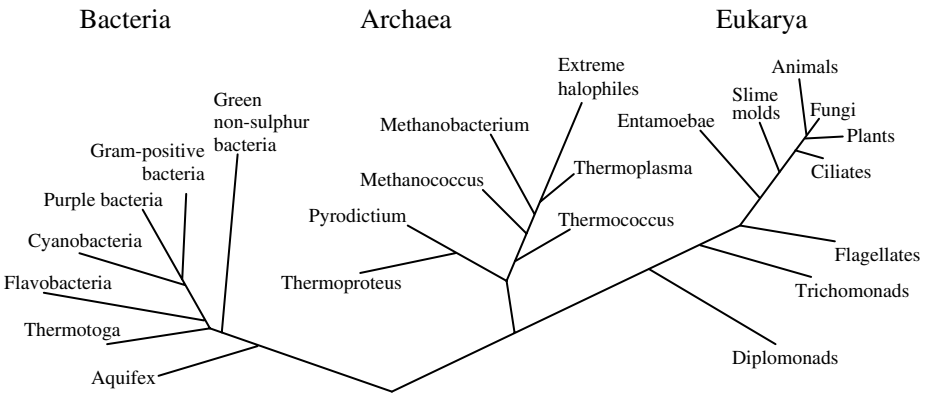


Fig. 1.2 Major divisions of living things, derived by C. Woese on the basis of 15S RNA sequences.

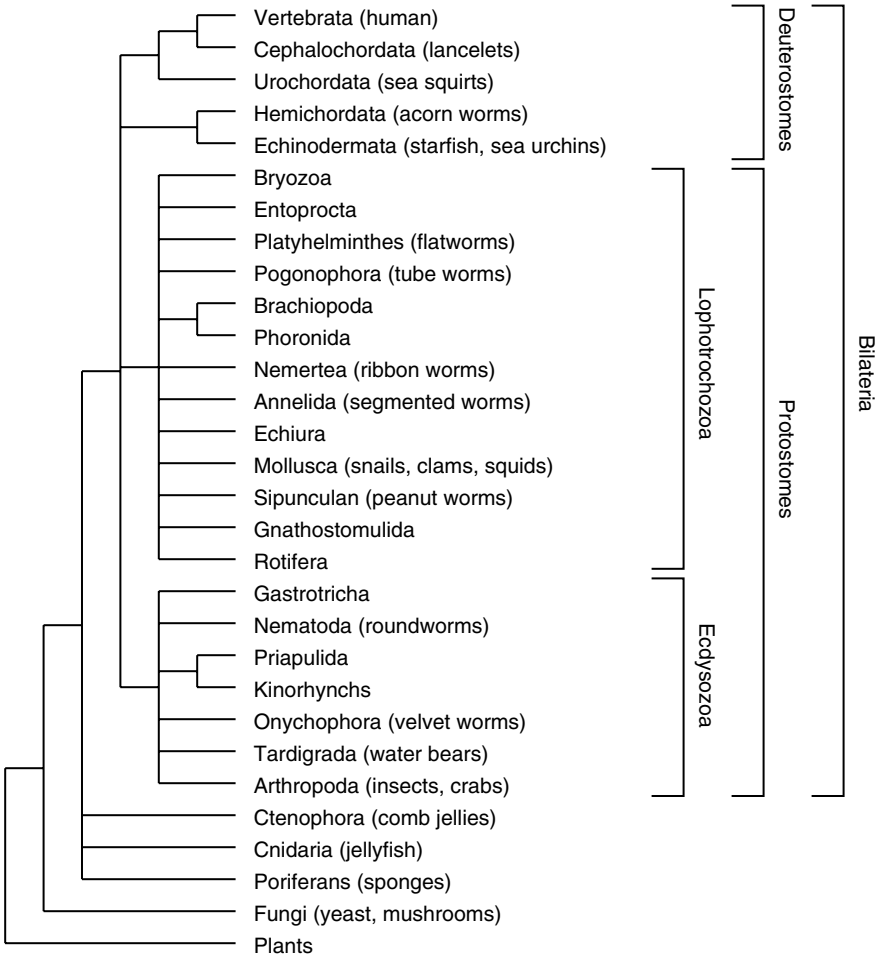


Fig. 1.3 Phylogenetic tree of metazoa (multicellular animals). Bilaterians include all animals that share a left–right symmetry of body plan. Protostomes and deuterostomes are two major lineages that separated at an early stage of evolution, estimated at 670 million years ago. They show very different patterns of embryological development, including different early cleavage patterns, opposite orientations of the mature gut with respect to the earliest invagination of the blastula, and the origin of the skeleton from mesoderm (deuterostomes) or ectoderm (protostomes). Protostomes comprise two subgroups distinguished on the basis of 18S RNA (from the small ribosomal subunit) and HOX gene sequences. Morphologically, Ecdysozoa have a molting cuticle – a hard outer layer of organic material. Lophotrochozoa have soft bodies. (Based on Adouette, A., Balavoine, G., Lartillot, N., Lespinet, O., Prud'homme, B., and de Rosa, R. (2000) 'The new animal phylogeny: reliability and implications', *Proceedings of the National Academy of Sciences USA* 97, 4453–6.)

Figure 1.2 shows the deepest level of the tree of life. The Eukarya branch includes animals, plants, fungi and single-celled organisms. At the ends of the eukarya branch are the metazoa (multicellular organisms) (Fig. 1.3). We and our closest relatives are deuterostomes (Fig. 1.4).

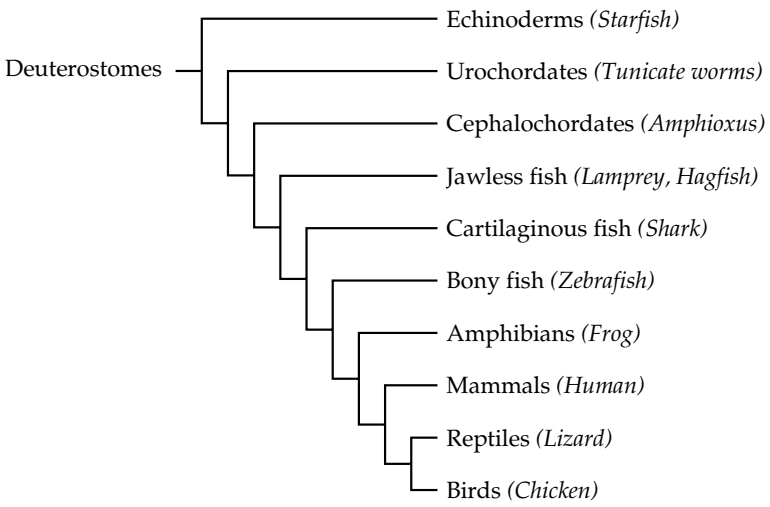


Fig. 1.4 Phylogenetic tree of vertebrates and our closest relatives. Chordates, including vertebrates, and echinoderms are all deuterostomes

Use of sequences to determine phylogenetic relationships

Previous sections have treated sequence databanks and biological relationships. Here are examples of the application of retrieval of sequences from databanks and sequence comparisons to analysis of biological relationships.

EXAMPLE 1.1

Retrieve the amino acid sequence of horse pancreatic ribonuclease.

Use the ExPASy server at the Swiss Institute for Bioinformatics: The URL is: `http://www.expasy.ch/cgi-bin/sprot-search-ful`. Type in the keywords `horse pancreatic ribonuclease` followed by the `ENTER` key. Select `RNP_HORSE` and then `FASTA format` (see Box: FASTA format). This will produce the following (the first line has been truncated):

```
>sp|P00674|RNP_HORSE RIBONUCLEASE PANCREATIC (EC 3.1.27.5) (RNASE 1) ...
KESPAMKFERQHMDSGSTSSSNPTYCNQMMKRRNMTQGWCKPVNTFVHEP
LADVQAICLQKNITCKNGQSNCYQSSSSMHITDCRLTSGSKYPNCAYQTS
QKERHIIVACEGNPYVPVHFDASVEVST
```

which can be cut and pasted into other programs.

For example, we could retrieve several sequences and align them (see Box: Sequence Alignment). Analysis of patterns of similarity among aligned sequences are useful properties in assessing closeness of relationships.

FASTA format

A very common format for sequence data is derived from conventions of FASTA, a program for **FAST A**lignment by W.R. Pearson. Many programs use FASTA format for reading sequences, or for reporting results.

A sequence in FASTA format:

- Begins with a single-line description. A > must appear in the first column. The rest of the title line is arbitrary but should be informative.
- Subsequent lines contain the sequence, one character per residue.
- Use one-letter codes for nucleotides or amino acids specified by the International Union of Biochemistry and International Union of Pure and Applied Chemistry (IUB/IUPAC).
  See `http://www.chem.qmw.ac.uk/iupac/misc/naabb.html`
  and `http://www.chem.qmw.ac.uk/iupac/AminoAcid/`
  Use Sec and U as the three-letter and one-letter codes for selenocysteine:
  `http://www.chem.qmw.ac.uk/iubmb/newsletter/1999/item3.html`
- Lines can have different lengths; that is, 'ragged right' margins.
- Most programs will accept lower case letters as amino acid codes.

An example of FASTA format: Bovine glutathione peroxidase

```
$>$gi$|$121664$|$sp$|$P00435$|$GSHC\_BOVIN GLUTATHIONE PEROXIDASE
MCAAQRSAAALAAAAPRTVYAFSARPLAGGEPFNLSSLRGKVLLIENVASLUGTTVRDYTQMNDLQRRLG
PRGLVVLGFPCNQFGHQENAKNEEILNCLKYVRPGGGFEPNFMLFEKCEVNGEKAHPLFAFLREVLPTPS
DDATALMTDPKFITWSPVCRNDVSWNFEKFLVGPDGVPVRRYSRRFLTIDIEPDIETLLSQGASA
```

The title line contains the following fields:

> is obligatory in column 1

gi|**121664** is the *geninfo number*, an identifier assigned by the US National Center for Biotechnology Information (NCBI) to every sequence in its ENTREZ databank. The NCBI collects sequences from a variety of sources, including primary archival data collections and patent applications. Its gi numbers provide a common and consistent 'umbrella' identifier, superimposed on different conventions of source databases. When a source database updates an entry, the NCBI creates a new entry with a new gi number if the changes affect the sequence, but updates and retains its entry if the changes affect only non-sequence information, such as a literature citation.

sp|**P00435** indicates that the source database was SWISS-PROT, and that the accession number of the entry in SWISS-PROT was P00435.

GSHC_BOVIN GLUTATHIONE PEROXIDASE is the SWISS-PROT identifier of sequence and species, (GSHC_BOVIN), followed by the name of the molecule.

Sequence alignment

Sequence alignment is the assignment of residue–residue correspondences. We may wish to find:

- a *Global match:* align all of one sequence with all of the other.

```
And.--so,.from.hour.to.hour,.we.ripe.and.ripe
||||    ||||||||||||||||||||||||   ||||||
And.then,.from.hour.to.hour,.we.rot-.and.rot-
```

  This illustrates mismatches, insertions and deletions.

- a *Local match:* find a region in one sequence that matches a region of the other.

```
 My.care.is.loss.of.care,.by.old.care.done,
   |||||||||    ||||||||||||   |||||| ||
Your.care.is.gain.of.care,.by.new.care.won
```

  For local matching, overhangs at the ends are not treated as gaps. In addition to mismatches, seen in this example, insertions and deletions within the matched region are also possible.

- a *Motif match:* find matches of a short sequence in one or more regions internal to a long one. In this case one mismatching character is allowed. Alternatively one could demand perfect matches, or allow more mismatches or even gaps.

```
        match
         ||||
for the watch to babble and to talk is most tolerable
```

  or:

```
                                match
                                ||||
Any thing that's mended is but patched: virtue that transgresses is
    match                                           match
     ||||                                            ||||
but patched with sin; and sin that amends is but patched with virtue
```

- a *Multiple alignment:* a mutual alignment of many sequences.

```
no.sooner.---met.---------but.they.-look'd
no.sooner.look'd.---------but.they.-lo-v'd
no.sooner.lo-v'd.---------but.they.-sigh'd
no.sooner.sigh'd.---------but.they.--asked.one.another.the.reason
no.sooner.knew.the.reason.but.they.------------sought.the.remedy
no.sooner.                .but.they.
```

The last line shows characters conserved in all sequences in the alignment.

See Chapter 4 for an extended discussion of alignment.

EXAMPLE 1.2

Determine, from the sequences of pancreatic ribonuclease from horse *(Equus caballus)*, minke whale *(Balaenoptera acutorostrata)* and red kangaroo *(Macropus rufus)*, which two of these species are most closely related.

Knowing that horse and whale are placental mammals and kangaroo is a marsupial, we expect horse and whale to be the closest pair. Retrieving the three sequences as in the previous example and pasting the following:

```
>RNP_HORSE
KESPAMKFERQHMDSGSTSSSNPTYCNQMMKRRNMTQGWCKPVNTFVHEP
LADVQAICLQKNITCKNGQSNCYQSSSSMHITDCRLTSGSKYPNCAYQTS
QKERHIIVACEGNPYVPVHFDASVEVST
>RNP_BALAC
RESPAMKFQRQHMDSGNSPGNNPNYCNQMMMRRKMTQGRCKPVNTFVHES
LEDVKAVCSQKNVLCKNGRTNCYESNSTMHITDCRQTGSSKYPNCAYKTS
QKEKHIIVACEGNPYVPVHFDNSV
>RNP_MACRU
ETPAEKFQRQHMDTEHSTASSSNYCNLMMKARDMTSGRCKPLNTFIHEPK
SVVDAVCHQENVTCKNGRTNCYKSNSRLSITNCRQTGASKYPNCQYETSN
LNKQIIVACEGQYVPVHFDAYV
```

into the multiple-sequence alignment program CLUSTAL–W

`http://www.ebi.ac.uk/clustalw/` (or alternatively, T-coffee: `http://www.ch.embnet.org/software/TCoffee.html`)

produces the following:

```
CLUSTAL W (1.8) multiple sequence alignment

RNP_HORSE       KESPAMKFERQHMDSGSTSSSNPTYCNQMMKRRNMTQGWCKPVNTFVHEPLADVQAICLQ 60
RNP_BALAC       RESPAMKFQRQHMDSGNSPGNNPNYCNQMMMRRKMTQGRCKPVNTFVHESLEDVKAVCSQ 60
RNP_MACRU       -ETPAEKFQRQHMDTEHSTASSSNYCNLMMKARDMTSGRCKPLNTFIHEPKSVVDAVCHQ 59
                 *:** **:*****:  :......*** **  *.**.* ***:***:**.   *.*:* *

RNP_HORSE       KNITCKNGQSNCYQSSSSMHITDCRLTSGSKYPNCAYQTSQKERHIIVACEGNPYVPVHF 120
RNP_BALAC       KNVLCKNGRTNCYESNSTMHITDCRQTGSSKYPNCAYKTSQKEKHIIVACEGNPYVPVHF 120
RNP_MACRU       ENVTCKNGRTNCYKSNSRLSITNCRQTGASKYPNCQYETSNLNKQIIVACEG-QYVPVHF 118
                :*: ****::***:*.* : **:** *..****** *:**: :::*******  ******

RNP_HORSE       DASVEVST 128
RNP_BALAC       DNSV---- 124
RNP_MACRU       DAYV---- 122
                *  *
```

In this table, a * under the sequences indicates a position that is conserved (the same in all sequences), and : and . indicate positions at which all sequences contain residues of very similar physicochemical character (:), or somewhat similar physicochemical character (.).

→

EXAMPLE 1.2 *continued*

Large patches of the sequences are identical. There are numerous substitutions but only one internal deletion. By comparing the sequences *in pairs,* the number of identical residues shared among pairs in this alignment (not the same as counting *s) is:

Number of identical residues in aligned Ribonuclease A sequences (out of a total of 122–128 residues)			
Horse	and	Minke whale	95
Minke Whale	and	Red kangaroo	82
Horse	and	Red kangaroo	75

Horse and whale share the most identical residues. The result appears significant, and therefore confirms our expectations. *Warning: Or is the logic really the other way round?*

Let's try a harder one:

EXAMPLE 1.3

The two living genera of elephant are represented by the African elephant *(Loxodonta africana)* and the Indian *(Elephas maximus)*. It has been possible to sequence the mitochondrial cytochrome *b* from a specimen of the Siberian woolly mammoth *(Mammuthus primigenius)* preserved in the Arctic permafrost. To which modern elephant is this mammoth more closely related?

Retrieving the sequences and running CLUSTAL–W:

```
African elephant  MTHIRKSHPLLKIINKSFIDLPTPSNISTWWNFGSLLGACLITQILTGLFLAMHYTPDTM 60
Siberian mammoth  MTHIRKSHPLLKILNKSFIDLPTPSNISTWWNFGSLLGACLITQILTGLFLAMHYTPDTM 60
Indian elephant   MTHTRKSHPLFKIINKSFIDLPTPSNISTWWNFGSLLGACLITQILTGLFLAMHYTPDTM 60
                  *** ******:**:********************************************

African elephant  TAFSSMSHICRDVNYGWIIRQLHSNGASIFFLCLYTHIGRNIYYGSYLYSETWNTGIMLL 120
Siberian mammoth  TAFSSMSHICRDVNYGWIIRQLHSNGASIFFLCLYTHIGRNIYYGSYLYSETWNTGIMLL 120
Indian elephant   TAFSSMSHICRDVNYGWIIRQLHSNGASIFFLCLYTHIGRNIYYGSYLYSETWNTGIMLL 120
                  ************************************************************

African elephant  LITMATAFMGYVLPWGQMSFWGATVITNLFSAIPCIGTNLVEWIWGGFSVDKATLNRFFA 180
Siberian mammoth  LITMATAFMGYVLPWGQMSFWGATVITNLFSAIPYIGTDLVEWIWGGFSVDKATLNRFFA 180
Indian elephant   LITMATAFMGYVLPWGQMSFWGATVITNLFSAIPYIGTNLVEWIWGGFSVDKATLNRFFA 180
                  ******************************** ***:********************

African elephant  LHFILPFTMIALAGVHLTFLHETGSNNPLGLISDSDKIPFHPYYTIKDFLGLLILILLLL 240
Siberian mammoth  LHFILPFTMIALAGVHLTFLHETGSNNPLGLTSDSDKIPFHPYYTIKDFLGLLILILFLL 240
Indian elephant   FHFILPFTMVALAGVHLTFLHETGSNNPLGLTSDSDKIPFHPYYTIKDFLGLLILILLLL 240
                  :********:******************** ************************:**
```

→

EXAMPLE 1.3 *continued*

```
African elephant  LLALLSPDMLGDPDNYMPADPLNTPLHIKPEWYFLFAYAILRSVPNKLGGVLALLLSILI 300

Siberian mammoth  LLALLSPDMLGDPDNYMPADPLNTPLHIKPEWYFLFAYAILRSVPNKLGGVLALLLSILI 300
Indian elephant   LLALLSPDMLGDPDNYMPADPLNTPLHIKPEWYFLFAYAILRSVPNKLGGVLALFLSILI 300
                  **************************************************:*****

African Elephant  LGLMPLLHTSKHRSMMLRPLSQVLFWTLTMDLLTLTWIGSQPVEYPYIIIGQMASILYFS 360
Siberian mammoth  LGIMPLLHTSKHRSMMLRPLSQVLFWTLATDLLMLTWIGSQPVEYPYIIIGQMASILYFS 360
Indian elephant   LGLMPFLHTSKHRSMMLRPLSQVLFWTLTMDLLTLTWIGSQPVEYPYTIIGQMASILYFS 360
                  **:**:**********************: *** ************* ************

African elephant  IILAFLPIAGVIENYLIK 378
Siberian mammoth  IILAFLPIAGMIENYLIK 378
Indian elephant   IILAFLPIAGMIENYLIK 378
                  **********:*******
```

The mammoth and African elephant sequences have 10 mismatches, and the mammoth and Indian elephant sequences have 14 mismatches. It appears that mammoth is more closely related to African elephants. However, this result is less satisfying than the previous one. There are fewer differences. Are they significant? (It is harder to decide whether the differences are significant because we have no preconceived idea of what the answer should be.)

This example raises a number of questions:

1. We 'know' that African and Indian elephants and mammoths must be close relatives – just look at them. But could we tell *from these sequences alone* that they are from closely related species?

2. Given that the differences are small, do they represent evolutionary divergence arising from selection, or merely random noise or drift? We need sensitive statistical criteria for judging the significance of the similarities and differences.

As background to such questions, let us emphasize the distinction between *similarity* and *homology*. *Similarity* is the observation or measurement of resemblance and difference, independent of the source of the resemblance. *Homology* means, specifically, that the sequences and the organisms in which they occur are descended from a common ancestor, with the implication that the similarities are shared ancestral characteristics. Similarity of sequences (or of macroscopic biological characters) is observable in data collectable *now*, and involves no historical hypotheses. In contrast, assertions of homology are statements of historical events that are almost always unobservable. Homology must be an *inference* from observations of similarity. Only in a few special cases is homology directly observable; for instance in family pedigrees showing unusual phenotypes such as the Hapsburg lip, or in laboratory populations, or in clinical studies that follow the course of viral infections at the sequence level in individual patients.

The assertion that the cytochromes *b* from African and Indian elephants and mammoths are homologous *means* that there was a common ancestor, presumably containing a unique cytochrome *b*, that by alternative mutations gave rise to the proteins of mammoths and modern elephants. Does the very high degree of similarity of the sequences justify the conclusion that they are homologous; or are there other explanations?

- It might be that a functional cytochrome *b requires* so many conserved residues that cytochromes *b* from all animals are as similar to one another as the elephant and mammoth proteins are. We can test this by looking at cytochrome *b* sequences from other species. The result is that cytochromes *b* from other animals differ substantially from those of elephants and mammoths.
- A second possibility is that there are special requirements for a cytochrome *b* to function well in an elephant-like animal, that the three cytochrome *b* sequences started out from independent ancestors, and that common selective pressures forced them to become similar. (Remember that we are asking what can be deduced from cytochrome *b* sequences alone.)
- The mammoth may be more closely related to the African elephant, but since the time of the last common ancestor the cytochrome *b* sequence of the Indian elephant has evolved faster than that of the African elephant or the mammoth, accumulating more mutations.
- Still a fourth hypothesis is that all common ancestors of elephants and mammoths had very dissimilar cytochromes *b*, but that living elephants and mammoths gained a common gene by transfer from an unrelated organism via a virus.

Suppose however we conclude that the similarity of the elephant and mammoth sequences is taken to be high enough to be evidence of homology, what then about the ribonuclease sequences in the previous example? Are the *larger* differences among the pancreatic ribonucleases of horse, whale and kangaroo evidence that they are *not* homologues?

How can we answer these questions? Specialists have undertaken careful calibrations of sequence similarities and divergences, among many proteins from many species for which the taxonomic relationships have been worked out by classical methods. In the example of pancreatic ribonucleases, the reasoning from similarity to homology is justified. The question of whether mammoths are closer to African or Indian elephants is still too close to call, even using all available anatomical and sequence evidence. Analysis of sequence similarities are now sufficiently well established that they are considered the most reliable methods for establishing phylogenetic relationships, even though sometimes – as in the elephant example – the results may not be significant, while in other cases they even give incorrect answers. There are a lot of data available, effective tools for retrieving what is necessary to bring to bear on a specific question, and

powerful analytic tools. None of this replaces the need for thoughtful scientific judgement.

Use of SINES and LINES to derive phylogenetic relationships

Major problems with inferring phylogenies from comparisons of gene and protein sequences are (1) the wide range of variation of similarity, which may dip below statistical significance, and (2) the effects of different rates of evolution along different branches of the evolutionary tree. In many cases, even if sequence similarities confidently establish relationships, it may be impossible to decide the *order* in which sets of taxa have split. The phylogeneticist's dream – features that have 'all-or-none' character, and the appearance of which is irreversible so that the order of branching events can be decided – is in some cases afforded by certain non-coding sequences in genomes.

SINES and LINES (Short and Long Interspersed Nuclear Elements) are repetitive non-coding sequences that form large fractions of eukaryotic genomes – at least 30% of human chromosomal DNA, and over 50% of some higher plant genomes. Typically, SINES are ~70–500 base pairs long, and up to 10^6 copies may appear. LINES may be up to 7000 base pairs long, and up to 10^5 copies may appear. SINES enter the genome by reverse transcription of RNA. Most SINES contain a 5′ region homologous to tRNA, a central region unrelated to tRNA, and a 3′ AT-rich region.

Features of SINES that make them useful for phylogenetic studies include:

- A SINE is either present or absent. Presence of a SINE at any particular position is a property that entails no complicated and variable measure of similarity.
- SINES are inserted at random in the non-coding portion of a genome. Therefore, appearance of similar SINES at the same locus in two species implies that the species share a common ancestor in which the insertion event occurred. No analogue of convergent evolution muddies this picture, because there is no selection for the *site* of insertion.
- SINE insertion appears to be irreversible: no mechanism for *loss* of SINES is known, other than rare large-scale deletions that include the SINE. Therefore, if two species share a SINE at a common locus, *absence* of this SINE in a third species implies that the first two species must be more closely related to each other than either is to the third.
- Not only do SINES show relationships, they imply which species came first. The last common ancestor of species containing a common SINE must have come *after* the last common ancestor linking these species and another that lacks this SINE.

N. Okada and colleagues applied SINE sequences to questions of phylogeny.

Whales, like Australians, are mammals that have adopted an aquatic lifestyle. But what – in the case of the whales – are their closest land-based relatives? Classical palaeontology linked the order *Cetacea* – comprising whales, dolphins and porpoises – with the order *Arteriodactyla* – even-toed ungulates (including for instance cattle). Cetaceans were thought to have diverged before the common ancestor of the three extant Arteriodactyl suborders: *Suiformes* (pigs), *Tylopoda* (including camels and llamas), and *Ruminantia* (including deer, cattle, goats, sheep, antelopes, giraffes, etc.). To place cetaceans properly among these groups, several studies were carried out with DNA sequences. Comparisons of mitochondrial DNA, and genes for pancreatic ribonuclease, γ-fibrinogen, and other proteins, suggested that the closest relatives of the whales are hippopotamuses, and that cetaceans and hippopotamuses form a separate group within the arteriodactyls, most closely related to the *Ruminantia* (see Weblem 1.7).

Analysis of SINES confirms this relationship. Several SINES are common to *Ruminantia,* hippopotamuses and cetaceans. Four SINES appear in hippopotamuses and cetaceans only. These observations imply the phylogenetic tree shown in Figure 1.5, in which the SINE insertion events are marked. [Note added in proof: New fossils of land-based ancestors of whales confirm the link between whales and arteriodactyls. This is a good example of the complementarity between molecular and paleontological methods: DNA sequence analysis can

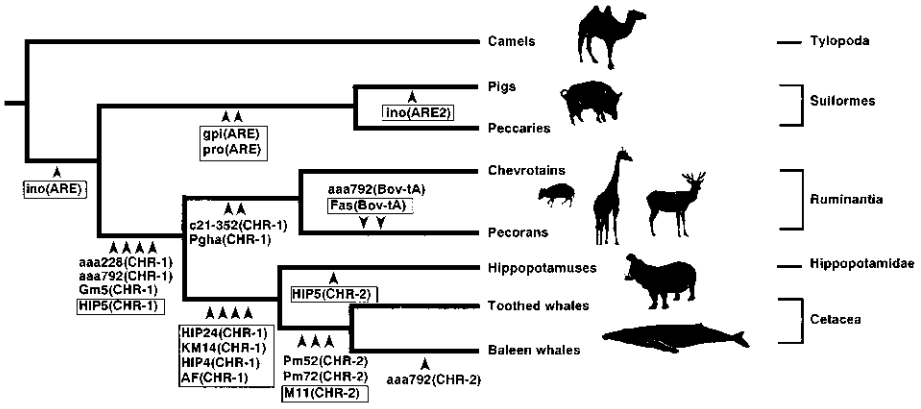


Fig. 1.5 Phylogenetic relationships among cetaceans and other arteriodactyl subgroups, derived from analysis of SINE sequences. Small arrowheads mark insertion events. Each arrowhead indicates the presence of a particular SINE or LINE at a specific locus in all species to the *right* of the arrowhead. Lower case letters identify loci, upper-case letters identify sequence patterns. For instance, the ARE2 pattern sequence appears only in pigs, at the ino locus. The ARE pattern appears twice in the pig genome, at loci gpi and pro, and in the precary genome at the same loci. The ARE insertion occured in a species ancestral to pigs and peccaries but to no other species in the diagram. This implies that pigs and peccaries are more closely related to each other than to any of the other animals studied. (From Nikaido, M., Rooney, A.P., and Okada, N. (1999) 'Phylogenetic relationships among cetartiodactyls based on insertions of short and long interspersed elements: Hippopotamuses are the closest extant relatives of whales', *Proceedings of the National Academy of Sciences USA* 96, 10261–6. (©1999, National Academy of Sciences, USA))

specify relationships among living species quite precisely, but only with fossils can one investigate the relationships among their extinct ancestors.]

Searching for similar sequences in databases: PSI-BLAST

A common theme of the examples we have treated is the search of a database for items similar to a probe. For instance, if you determine the sequence of a new gene, or identify within the human genome a gene responsible for some disease, you will wish to determine whether related genes appear in other species. The ideal method is both *sensitive* – that is, it picks up even very distant relationships – and *selective* – that is, all the relationships that it reports are true.

Database search methods involve a tradeoff between sensitivity and selectivity. Does the method find all or most of the 'hits' that are actually present, or does it miss a large fraction? Conversely, how many of the 'hits' it reports are incorrect? Suppose a database contains 1000 globin sequences. Suppose a search of this database for globins reported 900 results, 700 of which were really globin sequences and 200 of which were not. This result would be said to have 300 false negatives (misses) and 200 false positives. Lowering a tolerance threshold will increase both the number of false negatives and false positives. Often one is willing to work with low thresholds to make sure of not missing anything that might be important; but this requires detailed examination of the results to eliminate the resulting false positives.

A powerful tool for searching sequence databases with a probe sequence is PSI-BLAST, from the US National Center for Biotechnological Information (NCBI). PSI-BLAST stands for 'Position Sensitive Iterated – Basic Linear Alignment Sequence Tool'. A previous program, BLAST, worked by identifying local regions of similarity without gaps and then piecing them together. The PSI in PSI-BLAST refers to enhancements that identify patterns within the sequences at preliminary stages of the database search, and then progressively refine them. Recognition of conserved patterns can sharpen both the selectivity and sensitivity of the search. PSI-BLAST involves a repetitive (or iterative) process, as the emergent pattern becomes better defined in successive stages of the search.

EXAMPLE 1.4

Homologues of the human PAX-6 gene. PAX-6 genes control eye development in a widely divergent set of species (see Box). The human PAX-6 gene encodes the protein appearing in SWISS-PROT entry P26367. To run PSI-BLAST, go to the following URL: `http://www.ncbi.nlm.nih.gov/blast/psiblast.cgi`.

Enter the sequence, and use the default options for selections of the database to search and the similarity matrix used.

→

EXAMPLE 1.4 *continued*

The program returns a list of entries similar to the probe, sorted in decreasing order of statistical significance. (Extracts from the response are shown in the Box, *Results of PSI-BLAST search for human PAX-6 protein*.) A typical line appears as follows:

```
pir||I45557  eyeless, long form - fruit fly (Drosophila melano...   255  7e-67
```

The first item on the line is the database and corresponding entry number (separated by ||) in this case the PIR (Protein Identification Resource) entry I45557. It is the *Drosophila* homologue *eyeless*. The number 255 is a score for the match detected, and the significance of this match is measured by $E = 7 \times 10^{-67}$. E is related to the probability that the observed degree of similarity could have arisen by chance: E is the number of sequences that would be expected to match as well or better than the one being considered, if the same database were probed with random sequences. $E = 7 \times 10^{-67}$ means that it is *extremely* unlikely that even *one* random sequence would match as well as the *Drosophila* homologue. Values of E below about 0.05 would be considered significant; at least they might be worth considering. For borderline cases, you would ask: are the mismatches conservative? Is there any pattern or are the matches and mismatches distributed randomly through the sequences? There is an elusive concept, the *texture* of an alignment, that you will become sensitive to.

Note that if there are many sequences in the databank that are very similar to the probe sequence, they will head the list. In this case, there are many very similar PAX genes in other mammals. You may have to scan far down the list to find a distant relative that you consider interesting.

In fact, the program has matched only a portion of the sequences. The full alignment is shown in the Box, *Complete pairwise sequence alignment of human PAX–6 protein and Drosophila melanogaster eyeless*.

EXAMPLE 1.5

What species contain homologues to human PAX-6 detectable by PSI-BLAST?

PSI-BLAST reports the species in which the identified sequences occur (see Box, Results of PSI-BLAST search for human PAX-6 protein). These appear, embedded in the text of the output, in square brackets; for instance:

```
emb|CAA56038.1| (X79493) transcription factor [Drosophila melanogaster]
```

(In the section reporting *E*-values, the species names may be truncated.)
The following PERL program extracts species names from the PSI-BLAST output.

```
#!/usr/bin/perl
#extract species from psiblast output

# Method:
#   For each line of input, check for a pattern of form [Drosophila melanogaster]
#   Use each pattern found as the index in an associative array
#   The value corresponding to this index is irrelevant
#   By using an associative array, subsequent instances of the same
```

→

Et in terra PAX hominibus, muscisque . . .

The eyes of the human, fly and octopus are very different in structure. Conventional wisdom, noting the immense selective advantage conferred by the ability to see, held that eyes arose independently in different phyla. It therefore came as a great surprise that a gene controlling human eye development has a homologue governing eye development in *Drosophila.*

The PAX-6 gene was first cloned in the mouse and human. It is a master regulatory gene, controlling a complex cascade of events in eye development. Mutations in the human gene cause the clinical condition *aniridia,* a developmental defect in which the iris of the eye is absent or deformed. The PAX-6 homologue in *Drosophila* – called the *eyeless* gene – has a similar function of control over eye development. Flies mutated in this gene develop without eyes; conversely, expression of this gene in a fly's wing, leg, or antenna produces ectopic (= out of place) eyes. (The *Drosophila eyeless* mutant was first described in 1915. Little did anyone then suspect a relation to a mammalian gene.)

Not only are the insect and mammalian genes similar in sequence, they are so closely related that their function crosses species boundaries. Expression of the mouse PAX-6 gene in the fly causes ectopic eye development just as expression of the fly's own *eyeless* gene does.

PAX-6 has homologues in other phyla, including flatworms, ascidians, sea urchins and nematodes. The observation that rhodopsins – a family of proteins containing retinal as a common chromophore – function as light-sensitive pigments in different phyla is supporting evidence for a common origin of different photoreceptor systems. The genuine structural differences in the macroscopic anatomy of different eyes reflect the divergence and independent development of higher-order structure.

EXAMPLE 1.5 *continued*

```
#      species will overwrite the first instance, keeping only a unique set
#   After processing of input complete, sort results and print.


while (<>) {                              # read line of input
    if (/\[([A-Z][a-z]+ [a-z]+)\]/) {     # select lines containing strings of form
                                          #     [Drosophila melanogaster]
     $species{$1} = 1;                    # make or overwrite entry in
   }                                      #         associative array
}

foreach (sort(keys(%species))){           # in alphabetical order,
    print "$_\n";                         #    print species names
}
```

There are 52 species found (see Box: *Species recognized by PSI-BLAST 'hits' to probe sequence human PAX-6*).

→

The program makes use of PERL's rich pattern recognition resources to search for character strings of the *form* `[Drosophila melanogaster]`. We want to specify the following pattern:

- a square bracket,
- followed by a word beginning with an upper case letter followed by a variable number of lower case letters,
- then a space between words,
- then a word all in lower case letters,
- then a closing square bracket.

This kind of pattern is called a *regular expression* and appears in the PERL program in the following form: `[([A-Z][a-z]+ [a-z]+)]`

Building blocks of the pattern specify ranges of characters:

`[A-Z]` = any letter in the range A, B, C, . . . Z

`[a-z]` = any letter in the range a, b, c, . . . z

We can specify repetitions:

`[A-Z]` = *one* upper case letter

`[a-z]+` = *one or more* lower case letters

and combine the results:

`[A-Z][a-z]+ [a-z]+` = an upper case letter followed by one or more lower case letters (the genus name), followed by a blank, followed by one or more lower case letters (the species name).

Enclosing these in parentheses: `([A-Z][a-z]+ [a-z]+)` tells PERL to save the material that matched the pattern for future reference. In PERL this matched material is designated by the variable $*1*. Thus if the input line contained `[Drosophila melanogaster]`, the statement

```
$species{$1} = 1;
```

would effectively be:

```
$species{"Drosophila melanogaster"} = 1;
```

Finally, we want to include the brackets surrounding the genus and species name, but brackets signify character ranges. Therefore, we must precede the brackets by backslashes: `\[...\]`, to give the final pattern: `\[([A-Z][a-z]+ [a-z]+)\]`

The use of the associative array to retain only a unique set of species is another instructive aspect of the program. Recall that an associative array is a generalization of an ordinary array or vector, in which the elements are not indexed by integers but by arbitrary strings. A second reference to an associative array with a previously encountered index string may possibly change the value in the array but not the list of index strings. In this case we do not care about the value but just use the index strings to compile a unique list of species detected. Multiple references to the same species will merely overwrite the first reference, *not* make a repetitive list.

Results of PSI-BLAST search for human PAX-6 protein

```
Reference: Altschul, Stephen F., Thomas L. Madden, Alejandro
A. Schaeffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David
J. Lipman (1997), "Gapped BLAST and PSI-BLAST: a new generation of
protein database search programs", Nucleic Acids Res. 25:3389-3402.

Query= sp|P26367|PAX6_HUMAN PAIRED BOX PROTEIN PAX-6
(OCULORHOMBIN) (ANIRIDIA, TYPE II PROTEIN) - Homo sapiens (Human).
         (422 letters)

                Sequences with E-value BETTER than threshold

                                                                    Score  E
Sequences producing significant alignments:                         (bits) Value

 ref|NP_037133.1|  paired box homeotic gene 6 >gi|2495314|sp|P7...   730    0.0
 ref|NP_000271.1|  paired box gene 6, isoform a >gi|417450|sp|P...   730    0.0
 pir||A41644  homeotic protein aniridia - human                      728    0.0
 gb|AAA59962.1|  (M77844) oculorhombin [Homo sapiens] >gi|18935...   728    0.0
 prf||1902328A  PAX6 gene [Homo sapiens]                             724    0.0
 emb|CAB05885.1|  (Z83307) PAX6 [Homo sapiens]                       723    0.0
 ref|NP_001595.2|  paired box gene 6, isoform b                      721    0.0
 ref|NP_038655.1|  paired box gene 6 >gi|543296|pir||S42234 pai...   721    0.0
 dbj|BAA23004.1|  (D87837) PAX6 protein [Gallus gallus]              717    0.0
 gb|AAF73271.1|AF154555_1  (AF154555) paired domain transcripti...   714    0.0
 sp|P55864|PAX6_XENLA  PAIRED BOX PROTEIN PAX-6 >gi|1685056|gb|...   713    0.0
 gb|AAB36681.1|  (U76386) paired-type homeodomain Pax-6 protein...   712    0.0
 gb|AAB05932.1|  (U64513) Xpax6 [Xenopus laevis]                     712    0.0
 sp|P47238|PAX6_COTJA  PAIRED BOX PROTEIN PAX-6 (PAX-QNR) >gi|4...   710    0.0
 dbj|BAA24025.1|  (D88741) PAX6 SL [Cynops pyrrhogaster]             707    0.0
 gb|AAD50903.1|AF169414_1  (AF169414) paired-box transcription ...   706    0.0
 dbj|BAA13680.1|  (D88737) Xenopus Pax-6 long [Xenopus laevis]       703    0.0
 sp|P26630|PAX6_BRARE  PAIRED BOX PROTEIN PAX[ZF-A] (PAX-6) >gi...   699    0.0
 dbj|BAA24024.1|  (D88741) PAX6 LL [Cynops pyrrhogaster]             697    0.0
 gb|AAD50901.1|AF169412_1  (AF169412) paired-box transcription ...   696    0.0
 emb|CAA68835.1|  (Y07546) PAX-6 protein [Astyanax mexicanus] >...   693    0.0
 pir||I50108  paired box transcription factor Pax-6 - zebra fis...   689    0.0
 sp|O73917|PAX6_ORYLA  PAIRED BOX PROTEIN PAX-6 >gi|3115324|emb...   686    0.0
 gb|AAC96095.1|  (AF061252) Pax-family transcription factor 6.2...   684    0.0
 emb|CAA68837.1|  (Y07547) PAX-6 protein [Astyanax mexicanus]        683    0.0
 emb|CAA68836.1|  (Y07547) PAX-6 protein [Astyanax mexicanus]        675    0.0
 emb|CAA68838.1|  (Y07547) PAX-6 protein [Astyanax mexicanus]        675    0.0
 emb|CAA16493.1|  (AL021531) PAX6 [Fugu rubripes]                    646    0.0
 gb|AAF73273.1|AF154557_1  (AF154557) paired domain transcripti...   609    e-173
 dbj|BAA24023.1|  (D88741) PAX6 SS [Cynops pyrrhogaster]             609    e-173
 prf||1717390A  pax gene [Danio rerio]                               609    e-173
 gb|AAF73268.1|AF154552_1  (AF154552) paired domain transcripti...   608    e-173
 gb|AAD50904.1|AF169415_1  (AF169415) paired-box transcription ...   605    e-172
 gb|AAF73269.1|AF154553_1  (AF154553) paired domain transcripti...   604    e-172
 dbj|BAA13681.1|  (D88738) Xenopus Pax-6 short [Xenopus laevis]      600    e-171
 dbj|BAA24022.1|  (D88741) PAX6 LS [Cynops pyrrhogaster]             599    e-170
 gb|AAD50902.1|AF169413_1  (AF169413) paired-box transcription ...   595    e-169
 gb|AAF73270.1|  (AF154554) paired domain transcription factor ...   594    e-169
 gb|AAB07733.1|  (U67887) XLPAX6 [Xenopus laevis]                    592    e-168
 gb|AAA40109.1|  (M77842) oculorhombin [Mus musculus]                455    e-127
 emb|CAA11364.1|  (AJ223440) Pax6 [Branchiostoma floridae]           440    e-122
```

→

Results of PSI-BLAST search for human PAX-6 protein *continued*

```
 emb|CAA11366.1|  (AJ223442) Pax6 [Branchiostoma floridae]          437    e-122
 gb|AAB40616.1|  (U59830) Pax-6 [Loligo opalescens]                 437    e-122
 pir||A57374  paired box transcription factor Pax-6 - sea urchi...  437    e-121
 emb|CAA11368.1|  (AJ223444) Pax6 [Branchiostoma floridae]          435    e-121
 emb|CAA11367.1|  (AJ223443) Pax6 [Branchiostoma floridae]          433    e-120
 emb|CAA11365.1|  (AJ223441) Pax6 [Branchiostoma floridae]          412    e-114
 pir||JC6130  paired box transcription factor Pax-6 - Ribbonwor...  396    e-109
 gb|AAD31712.1|AF134350_1  (AF134350) transcription factor Toy ...  380    e-104
 gb|AAB36534.1|  (U77178) paired box homeodomain protein TPAX6 ...  377    e-104
 emb|CAA71094.1|  (Y09975) Pax-6 [Phallusia mammilata]              342    4e-93
 dbj|BAA20936.1|  (AB002408) mdkPax-6 [Oryzias sp.]                 338    6e-92
 pir||S60252  paired box transcription factor vab-3 - Caenorhab...  336    2e-91
 pir||T20900  hypothetical protein F14F3.1 - Caenorhabditis ele...  336    2e-91
 pir||S36166  paired box transcription factor Pax-6 - rat (frag...  335    5e-91
 sp|P47237|PAX6_CHICK  PAIRED BOX PROTEIN PAX-6 >gi|2147404|pir...  333    2e-90
 dbj|BAA75672.1|  (AB017632) DjPax-6 [Dugesia japonica]             329    4e-89
 gb|AAF64460.1|AF241310_1  (AF241310) transcription factor PaxB...  290    2e-77
 gb|AAF73274.1|  (AF154558) paired domain transcription factor ...  287    1e-76
 pdb|6PAX|A  Chain A, Crystal Structure Of The Human Pax-6 Pair...  264    1e-69
 pir||C41061  paired box homolog Pax6 - mouse (fragment)            261    9e-69
 gb|AAC18658.1|  (U73855) Pax6 [Bos taurus]                         259    4e-68
 pir||I45557  eyeless, long form - fruit fly (Drosophila melano...  255    7e-67
 gb|AAF59318.1|  (AE003843) ey gene product [Drosophila melanog...  255    7e-67
```

. . . many additional "hits" deleted . . .

. . . two selected alignments follow . . .

```
------------------------------------------------------------------------------
                                Alignments

>ref|NP_037133.1| paired box homeotic gene 6
 sp|P70601|PAX6_RAT PAIRED BOX PROTEIN PAX-6
 gb|AAB09042.1| (U69644) paired-box/homeobox protein [Rattus norvegicus]
          Length = 422

 Score =  730 bits (1865), Expect = 0.0
 Identities = 362/422 (85%), Positives = 362/422 (85%)

Query: 1   MQNSHSGVNQLGGVFVNGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVSKILGRY 60
           MQNSHSGVNQLGGVFVNGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVSKILGRY
Sbjct: 1   MQNSHSGVNQLGGVFVNGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVSKILGRY 60

Query: 61  YETGSIRPRAIGGSKPRVATPEVVSKIAQYKRECPSIFAWEIRDRLLSEGVCTNDNIPSV 120
           YETGSIRPRAIGGSKPRVATPEVVSKIAQYKRECPSIFAWEIRDRLLSEGVCTNDNIPSV
Sbjct: 61  YETGSIRPRAIGGSKPRVATPEVVSKIAQYKRECPSIFAWEIRDRLLSEGVCTNDNIPSV 120

Query: 121 SSINRVLRNLASEKQQMGADGMYDKLRMLNGQTGSWGTRPGWYPGTSVPGQPTXXXXXXX 180
           SSINRVLRNLASEKQQMGADGMYDKLRMLNGQTGSWGTRPGWYPGTSVPGQPT
Sbjct: 121 SSINRVLRNLASEKQQMGADGMYDKLRMLNGQTGSWGTRPGWYPGTSVPGQPTQDGCQQQ 180

Query: 181 XXXXXNTNSISSNGEDSDEAQMXXXXXXXXXXNRTSFTQEQIEALEKEFERTHYPDVFAR 240
                 NTNSISSNGEDSDEAQM          NRTSFTQEQIEALEKEFERTHYPDVFAR
Sbjct: 181 EGQGENTNSISSNGEDSDEAQMRLQLKRKLQRNRTSFTQEQIEALEKEFERTHYPDVFAR 240

Query: 241 ERLAAKIDLPEARIQVWFSNRRAKWRREEKLRNQRRQASNXXXXXXXXXXXXXXXVYQPIP 300
           ERLAAKIDLPEARIQVWFSNRRAKWRREEKLRNQRRQASN              VYQPIP
Sbjct: 241 ERLAAKIDLPEARIQVWFSNRRAKWRREEKLRNQRRQASNTPSHIPISSSFSTSVYQPIP 300
```

→

Results of PSI-BLAST search for human PAX-6 protein *Continued*

```
Query: 301 QPTTPVSSFTSGSMLGRTDTALTNTYSALPPMPSFTMANNLPMQPPVPSQTSSYSCMLPT 360
           QPTTPVSSFTSGSMLGRTDTALTNTYSALPPMPSFTMANNLPMQPPVPSQTSSYSCMLPT
Sbjct: 301 QPTTPVSSFTSGSMLGRTDTALTNTYSALPPMPSFTMANNLPMQPPVPSQTSSYSCMLPT 360

Query: 361 SPSVNGRSYDTYTPPHMQTHMNSQPMXXXXXXXXXLIXXXXXXXXXXXXXXXDMSQYWPR 420
           SPSVNGRSYDTYTPPHMQTHMNSQPM         LI               DMSQYWPR
Sbjct: 361 SPSVNGRSYDTYTPPHMQTHMNSQPMGTSGTTSTGLISPGVSVPVQVPGSEPDMSQYWPR 420

Query: 421 LQ 422
           LQ
Sbjct: 421 LQ 422

>pir||I45557 eyeless, long form - fruit fly (Drosophila melanogaster)
 emb|CAA56038.1| (X79493) transcription factor [Drosophila melanogaster]
          Length = 838

 Score =  255 bits (644), Expect = 7e-67
 Identities = 124/132 (93%), Positives = 128/132 (96%)

Query: 5   HSGVNQLGGVFVNGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVSKILGRYYETG 64
           HSGVNQLGGVFV GRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVSKILGRYYETG
Sbjct: 38  HSGVNQLGGVFVGGRPLPDSTRQKIVELAHSGARPCDISRILQVSNGCVSKILGRYYETG 97

Query: 65  SIRPRAIGGSKPRVATPEVVSKIAQYKRECPSIFAWEIRDRLLSEGVCTNDNIPSVSSIN 124
           SIRPRAIGGSKPRVAT EVVSKI+QYKRECPSIFAWEIRDRLL E VCTNDNIPSVSSIN
Sbjct: 98  SIRPRAIGGSKPRVATAEVVSKISQYKRECPSIFAWEIRDRLLQENVCTNDNIPSVSSIN 157

Query: 125 RVLRNLASEKQQ 136
           RVLRNLA++K+Q
Sbjct: 158 RVLRNLAAQKEQ 169
```

Complete pairwise sequence alignment of human PAX-6 protein and *Drosophila melanogaster* eyeless

```
PAX6_human      ---------------------------------MQNSHSGVNQLGGVFVNGRPLPDSTRQ 27
eyeless         MFTLQPTPTAIGTVVPPWSAGTLIERLPSLEDMAHKGHSGVNQLGGVFVGGRPLPDSTRQ 60
                                               ::.************.**********

PAX6_human      KIVELAHSGARPCDISRILQVSNGCVSKILGRYYETGSIRPRAIGGSKPRVATPEVVSKI 87
eyeless         KIVELAHSGARPCDISRILQVSNGCVSKILGRYYETGSIRPRAIGGSKPRVATAEVVSKI 120
                ******************************************************.******

PAX6_human      AQYKRECPSIFAWEIRDRLLSEGVCTNDNIPSVSSINRVLRNLASEKQQ----------- 136
eyeless         SQYKRECPSIFAWEIRDRLLQENVCTNDNIPSVSSINRVLRNLAAQKEQQSTGSGSSSTS 180
                :******************.*.*********************::*:*

PAX6_human      ------------MG-------------------------------------------ADG 141
eyeless         AGNSISAKVSVSIGGNVSNVASGSRGTLSSSTDLMQTATPLNSSESGGATNSGEGSEQEA 240
                           :*                                           :.

PAX6_human      MYDKLRMLNGQTGS--------------------WGTRP---------------------- 160
eyeless         IYEKLRLLNTQHAAGPGPLEPARAAPLVGQSPNHLGTRSSHPQLVHGNHQALQQHQQQSW 300
```

→

Continued

```
               :*:***:** * .:                    ***.

PAX6_human     -------GWYPG-------TSVP-----------------------------GQP---- 172
eyeless        PPRHYSGSWYPTSLSEIPISSAPNIASVTAYASGPSLAHSLSPPNDIKSLASIGHQRNCP 360
                      .***        :*.*                               *:

PAX6_human     ----------TQDGCQQQEGG---GENTNSISSNGEDSDEAQMRLQLKRKLQRNRTSFTQ 219
eyeless        VATEDIHLKKELDGHQSDETGSGEGENSNGGASNIGNTEDDQARLILKRKLQRNRTSFTN 420
                            ** *.:* *   ***:*. :**  :::: * ** *************:

PAX6_human     EQIEALEKEFERTHYPDVFARERLAAKIDLPEARIQVWFSNRRAKWRREEKLRNQRRQAS 279
eyeless        DQIDSLEKEFERTHYPDVFARERLAGKIGLPEARIQVWFSNRRAKWRREEKLRNQRRTPN 480
               :**::********************.**.*************************** ..

PAX6_human     NTPSHIPISSSFSTSVYQPIPQPTTPVSSFTSGSMLG----------------------- 316
eyeless        STGASATSSSTSATASLTDSPNSLSACSSLLSGSAGGPSVSTINGLSSPSTLSTNVNAPT 540
               .* :  . **: :*:     *:. :. **: ***  *

PAX6_human     ------------------------------------------------------------
eyeless        LGAGIDSSESPTPIPHIRPSCTSDNDNGRQSEDCRRVCSPCPLGVGGHQNTHHIQSNGHA 600


PAX6_human     ----------------------------RTDTALTNTYSALPPMPSFTMANNLPMQPPVP 348
eyeless        QGHALVPAISPRLNFNSGSFGAMYSNMHHTALSMSDSYGAVTPIPSFNHSAVGPLAPPSP 660
                                         :*  ::::::*.*:.*:***. :   *: ** *

PAX6_human     S-------QTSSYSCMLPTSP---------------------------------SVNGRS 368
eyeless        IPQQGDLTPSSLYPCHMTLRPPPMAPAHHHIVPGDGGRPAGVGLGSGQSANLGASCSGSG 720
                        :* *.* :.  *                                  * .* .

PAX6_human     YDTYTP-----------------------------PHMQTHMNSQP----------MGTS 389
eyeless        YEVLSAYALPPPPMASSSAADSSFSAASSASANVTPHHTIAQESCPSPCSSASHFGVAHS 780
               *:. :.                            **     :* *          :. *

PAX6_human     GTTSTGLISPGVS----------------VPVQVPGS----EPDMSQYWPRLQ----- 422
eyeless        SGFSSDPISPAVSSYAHMSYNYASSANTMTPSSASGTSAHVAPGKQQFFASCFYSPWV 838
               .  *:. ***.**                .* ...*:     *. .*::.
```

Species recognized by PSI-BLAST 'hits' to probe sequence human PAX-6

Acropora millepora
Archegozetes longisetosus
Astyanax mexicanus
Bos taurus
Branchiostoma floridae
Branchiostoma lanceolatum
Caenorhabditis elegans
Canis familiaris
Carassius auratus
Chrysaora quinquecirrha
Herdmania curvata
Homo sapiens
Hydra littoralis
Hydra magnipapillata
Hydra vulgaris
Ilyanassa obsoleta
Lampetra japonica
Lineus sanguineus
Loligo opalescens
Mesocricetus auratus

```
Ciona intestinalis                  Mus musculus
Coturnix coturnix                   Notophthalmus viridescens
Cynops pyrrhogaster                 Oryzias latipes
Danio rerio                         Paracentrotus lividus
Drosophila mauritiana               Petromyzon marinus
Drosophila melanogaster             Phallusia mammilata
Drosophila sechellia                Podocoryne carnea
Drosophila simulans                 Ptychodera flava
Drosophila virilis                  Rattus norvegicus
Dugesia japonica                    Schistosoma mansoni
Ephydatia fluviatilis               Strongylocentrotus purpuratus
Fugu rubripes                       Sus scrofa
Gallus gallus                       Takifugu rubripes
Girardia tigrina                    Tribolium castaneum
Halocynthia roretzi                 Triturus alpestris
Helobdella triserialis              Xenopus laevis
```

Introduction to protein structure

With protein structures we leave behind the one-dimensional world of nucleotide and amino acid sequences and enter the spatial world of molecular structures. Some of the facilities for archiving and retrieving molecular biological information survive this change pretty well intact, some must be substantially altered, and others do not make it at all.

Biochemically, proteins play a variety of roles in life processes: there are structural proteins (e.g. viral coat proteins, the horny outer layer of human and animal skin, and proteins of the cytoskeleton); proteins that catalyse chemical reactions (the enzymes); transport and storage proteins (haemoglobin); regulatory proteins, including hormones and receptor/signal transduction proteins; proteins that control genetic transcription; and proteins involved in recognition, including cell adhesion molecules, and antibodies and other proteins of the immune system.

Proteins are large molecules. In many cases only a small part of the structure – an *active site* – is functional, the rest existing only to create and fix the spatial relationship among the active site residues. Proteins evolve by structural changes produced by mutations in the amino acid sequence. The primary paradigm of evolution is that changes in DNA generate variability in protein structure and function, which affect the reproductive fitness of the individual, on which natural selection acts.

Approximately 15 000 protein structures are now known. Most were determined by X-ray crystallography or Nuclear Magnetic Resonance (NMR). From

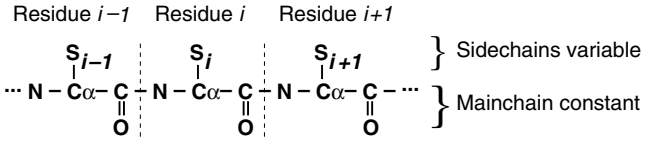


Fig. 1.6 The polypeptide chains of proteins have a mainchain of constant structure and sidechains that vary in sequence. Here S_{i-1}, S_i and S_{i+1} represent sidechains. The sidechains may be chosen, independently, from the set of 20 standard amino acids. It is the sequence of the sidechains that gives each protein its individual structural and functional characteristics.

these we have derived our understanding both of the functions of individual proteins – for example, the chemical explanation of catalytic activity of enzymes – and of the general principles of protein structure and folding.

Chemically, protein molecules are long polymers typically containing several thousand atoms, composed of a uniform repetitive *backbone* (or *main chain*) with a particular *sidechain* attached to each residue (see Fig. 1.6). The amino acid sequence of a protein records the succession of sidechains.

The polypeptide chain folds into a curve in space; the course of the chain defining a 'folding pattern'. Proteins show a great variety of folding patterns. Underlying these are a number of common structural features. These include the recurrence of explicit structural paradigms – for example, α-helices and β-sheets (Fig. 1.7) – and common principles or features such as the dense packing of the atoms in protein interiors. Folding may be thought of as a kind of intramolecular condensation or crystallization. (See Chapter 5.)

The hierarchical nature of protein architecture

The Danish protein chemist K.U. Linderstrøm-Lang described the following levels of protein structure: The amino acid sequence – the set of primary chemical bonds – is called the *primary structure*. The assignment of helices and sheets – the hydrogen-bonding pattern of the mainchain – is called the *secondary structure*. The assembly and interactions of the helices and sheets is called the *tertiary structure*. For proteins composed of more than one subunit, J.D. Bernal called the assembly of the monomers the *quaternary structure*. In some cases, evolution can merge proteins – changing quaternary to tertiary structure. For example, five separate enzymes in the bacterium *E. coli,* that catalyse successive steps in the pathway of biosynthesis of aromatic amino acids, correspond to five regions of a single protein in the fungus *Aspergillus nidulans*. Sometimes homologous monomers form oligomers in different ways; for instance, globins form tetramers in mammalian haemoglobins, and dimers – using a different interface – in the ark clam *Scapharca inaequivalvis*.

It has proved useful to add additional levels to the hierarchy:

- *Supersecondary structures*. Proteins show recurrent patterns of interaction between helices and sheets close together in the sequence. These

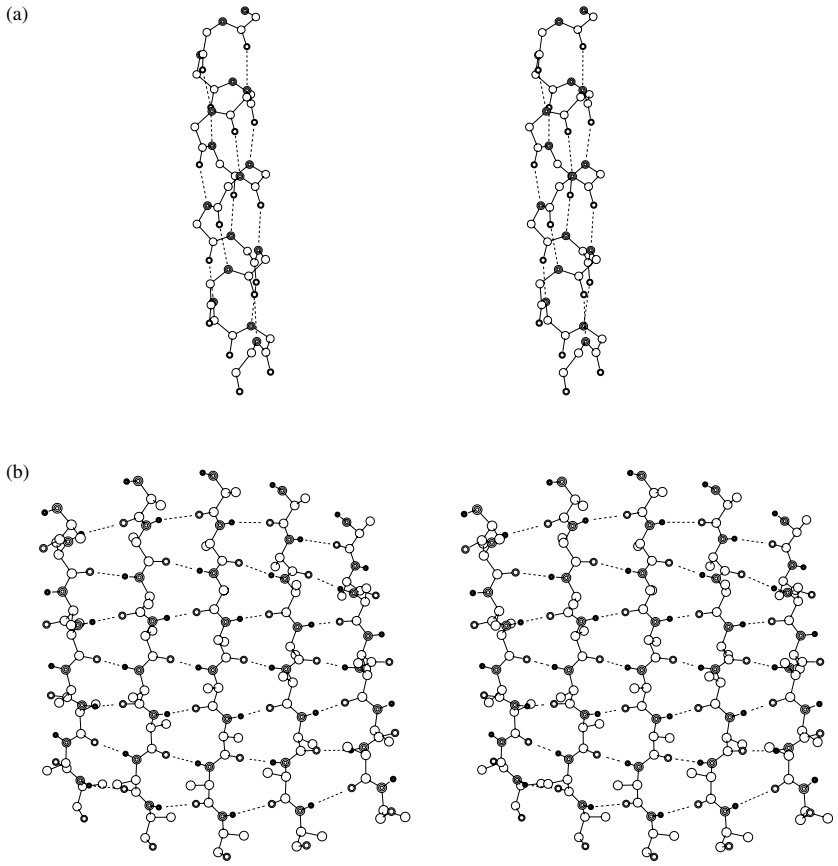


Fig. 1.7 Standard secondary structures of proteins. (a) α-helix. (b) β-sheet. (b) illustrates a parallel β-sheet, in which all strands pointing in the same direction. Antiparallel β-sheets, in which all pairs of adjacent strands point in opposite directions, are also common. In fact, β-sheets can be formed by any combination of parallel and antiparallel strands.

supersecondary structures include the α-helix hairpin, the β-hairpin, and the β-α-β unit (Fig. 1.8).

- *Domains*. Many proteins contain compact units within the folding pattern of a single chain, that look as if they should have independent stability. These are called domains. (Do not confuse domains as substructures of proteins with domains as general classes of living things: archaea, bacteria and eukaryotes.) The RNA-binding protein L1 has feature typical of multidomain proteins: the binding site appears in a cleft between the two domains, and the relative geometry of the two domains is flexible, allowing for ligand-induced conformational changes (Fig. 1.9). In the hierarchy, domains fall between supersecondary structures and the tertiary structure of a complete monomer.
- *Modular proteins*. Modular proteins are multidomain proteins which often contain many copies of closely related domains. Domains recur in many

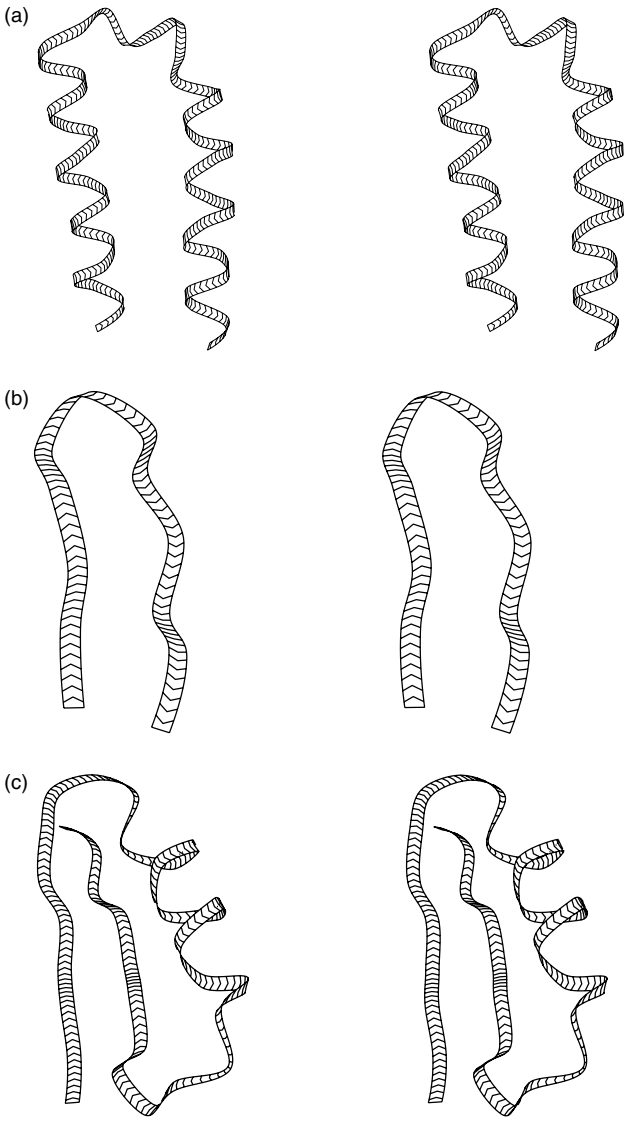


Fig. 1.8 Common supersecondary structures. top: α-helix hairpin, centre: β-hairpin, bottom: β-α-β unit. The chevrons indicate the direction of the chain.

proteins in different structural contexts; that is, different modular proteins can 'mix and match' sets of domains. For example, fibronectin, a large extracellular protein involved in cell adhesion and migration, contains 29 domains including multiple tandem repeats of three types of domains called F1, F2, and F3. It is a linear array of the form: $(F1)_6(F2)_2(F1)_3(F3)_{15}(F1)_3$. Fibronectin domains also appear in other modular proteins.

(See `http://www.bork.embl-heidelberg.de/Modules/` for pictures and nomenclature.)

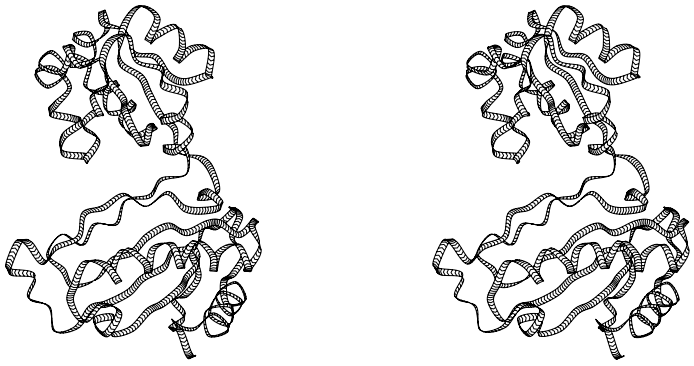


Fig. 1.9 Ribosomal protein L1 from *Methanococcus jannaschii* [1CJS]. ([1CJS] is the Protein Data Bank identification code of the entry.)

Classification of protein structures

The most general classification of families of protein structures is based on the secondary and tertiary structures of proteins.

Class	Characteristic
α-helical	secondary structure exclusively or almost exclusively α-helical
β-sheet	secondary structure exclusively or almost exclusively β-sheet
$\alpha + \beta$	α-helices and β-sheets separated in different parts of the molecule; absence of β-α-β supersecondary structure
α/β	helices and sheets assembled from β-α-β units
α/β-linear	line through centres of strands of sheet roughly linear
α/β-barrels	line through centres of strands of sheet roughly circular
little or no secondary structure	

Within these broad categories, protein structures show a variety of folding patterns. Among proteins with similar folding patterns, there are families that share enough features of structure, sequence and function to suggest evolutionary relationship. However, unrelated proteins often show similar structural themes.

Classification of protein structures occupies a key position in bioinformatics, not least as a bridge between sequence and function. We shall return to this theme, to describe results and relevant web sites. Meanwhile, the following album of small structures provides opportunities for practicing visual analysis and recognition of the important spatial patterns (Fig. 1.10). Trace the chains visually, picking out helices and sheets. (The chevrons indicate the direction of the chain.) Can you see supersecondary structures? Into which general classes do these structures fall? (See Exercises 1.12 and 1.13, and Problem 1.7.) Many other examples appear in *Introduction to Protein Architecture: The Structural Biology of Proteins*.

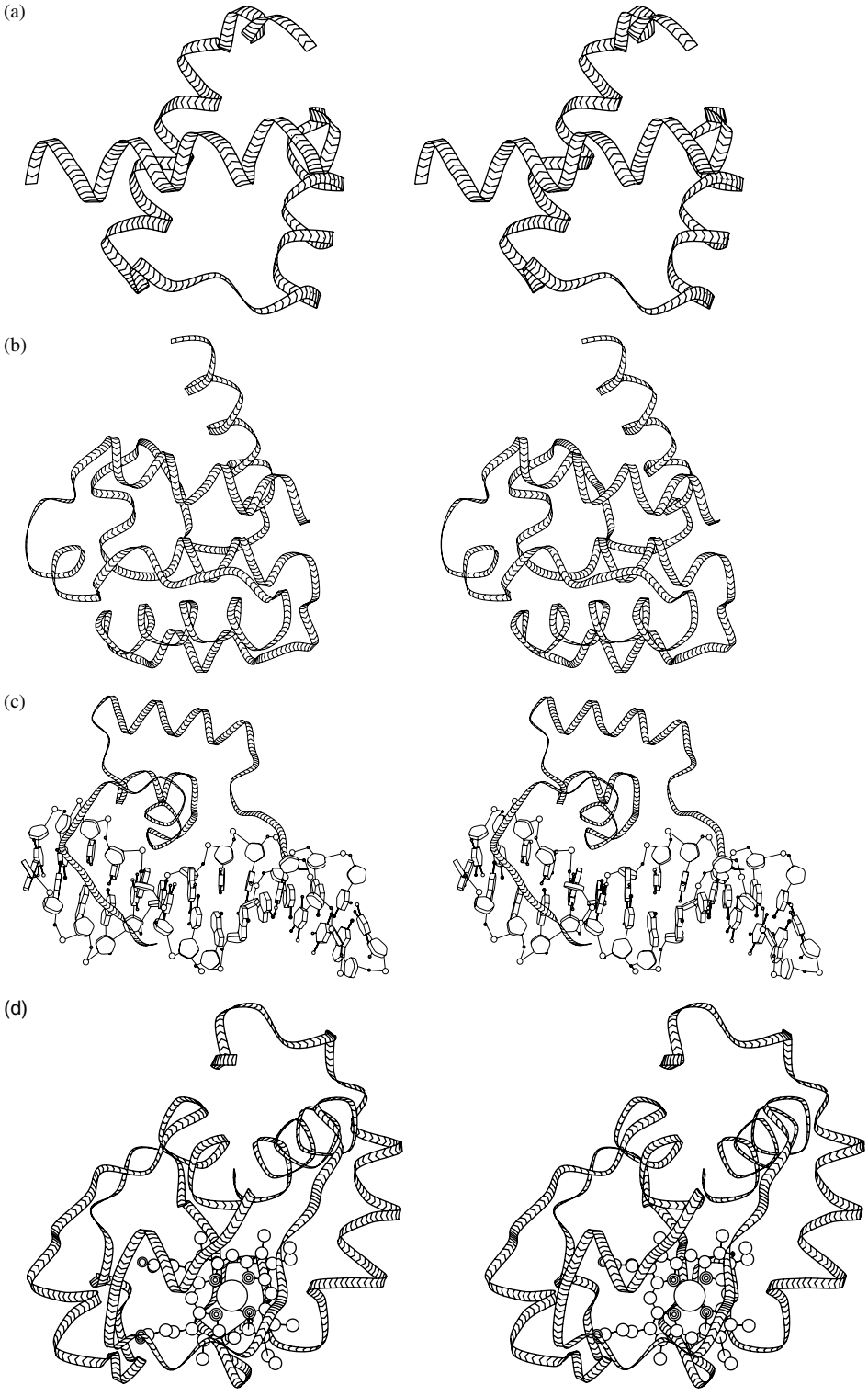


Fig. 1.10 *Continued*

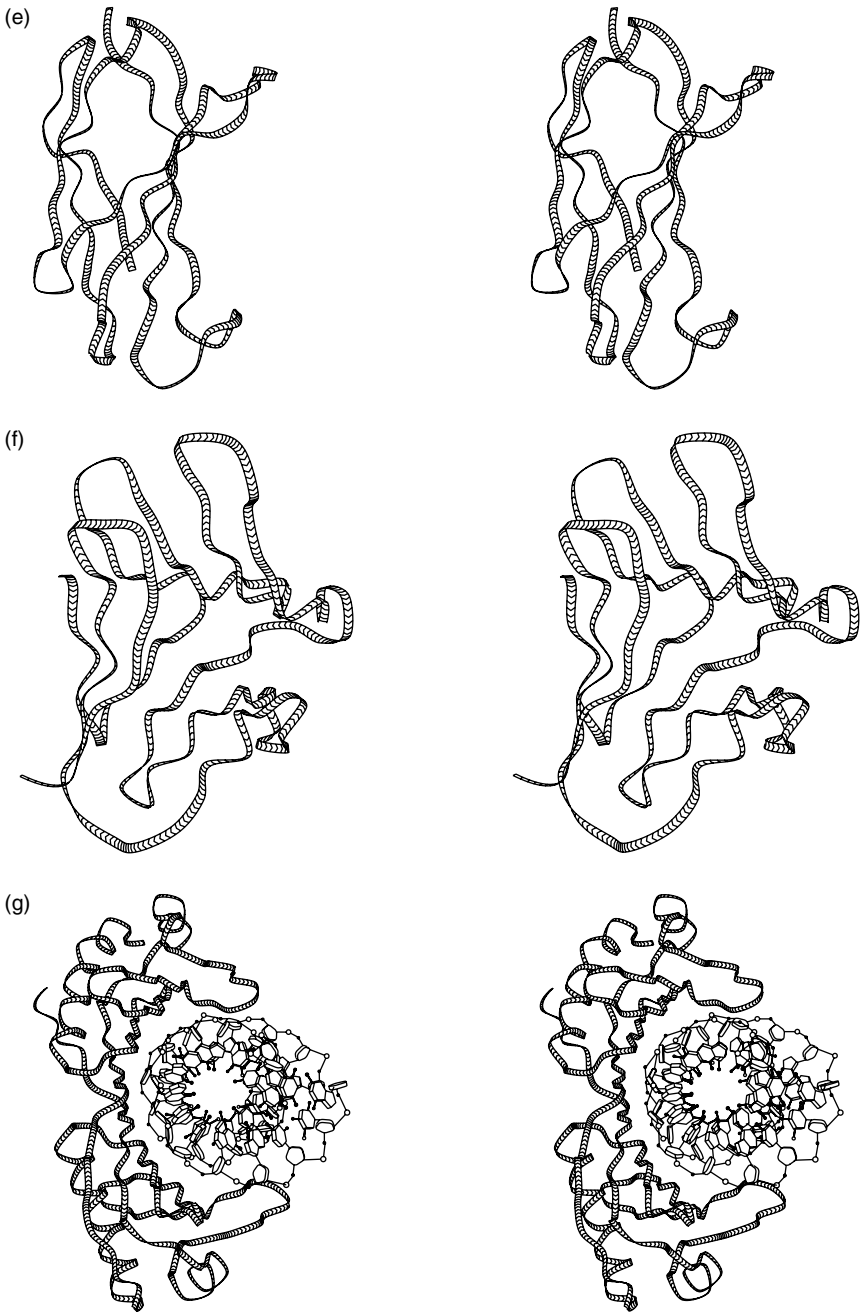


Fig. 1.10 *Continued*

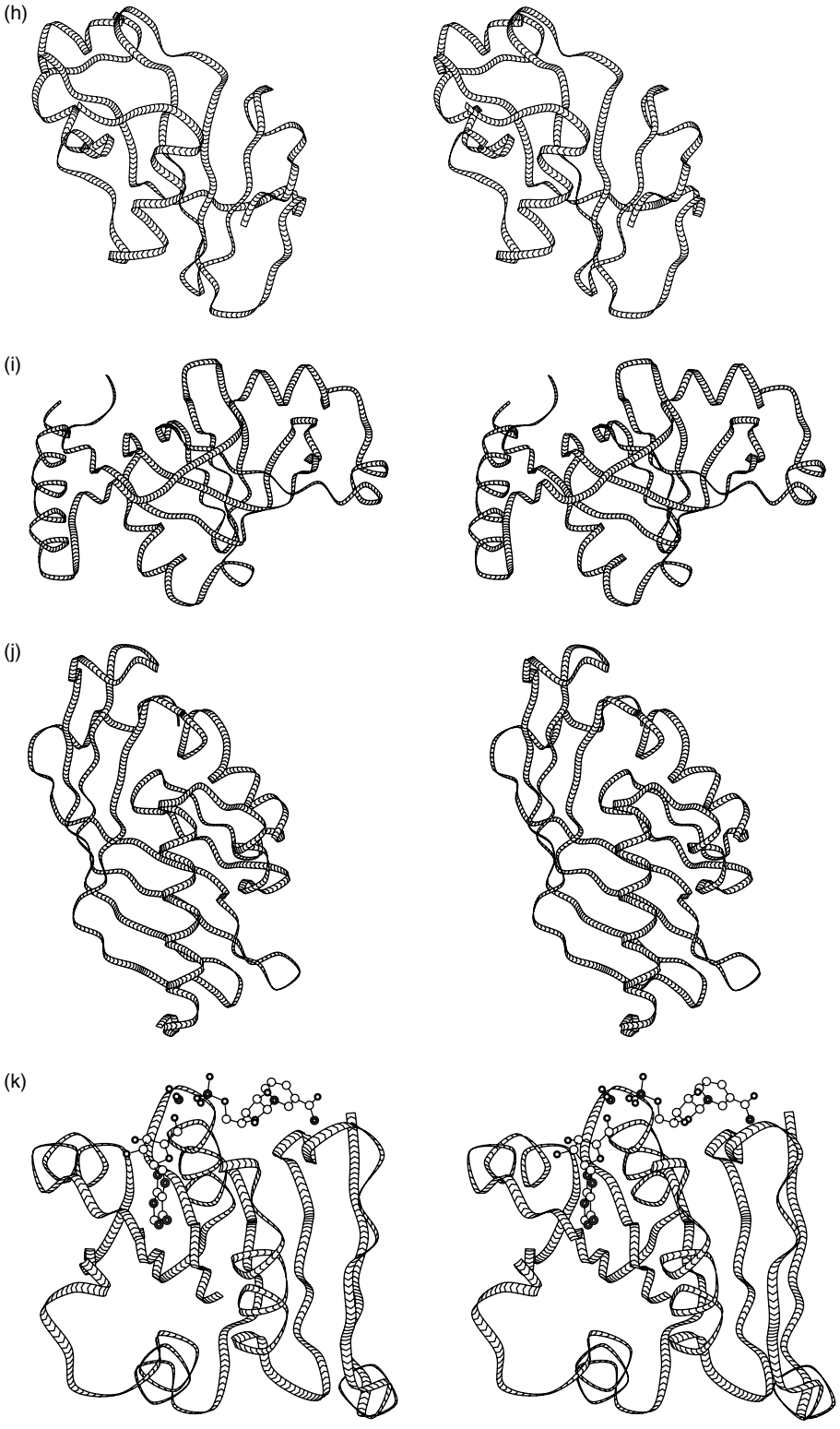
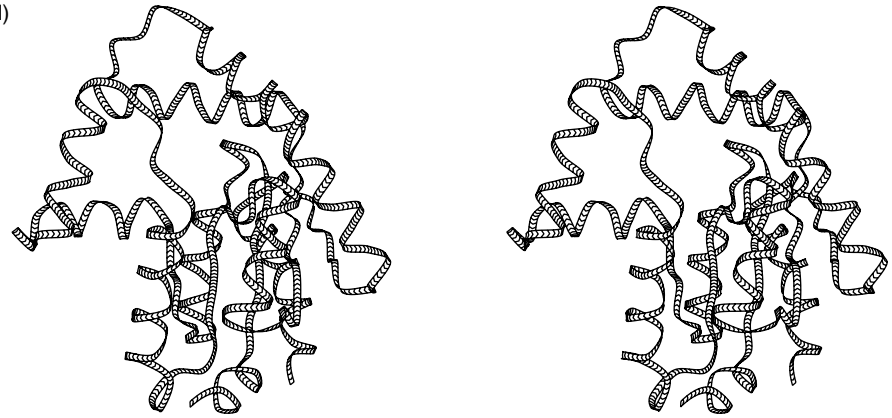
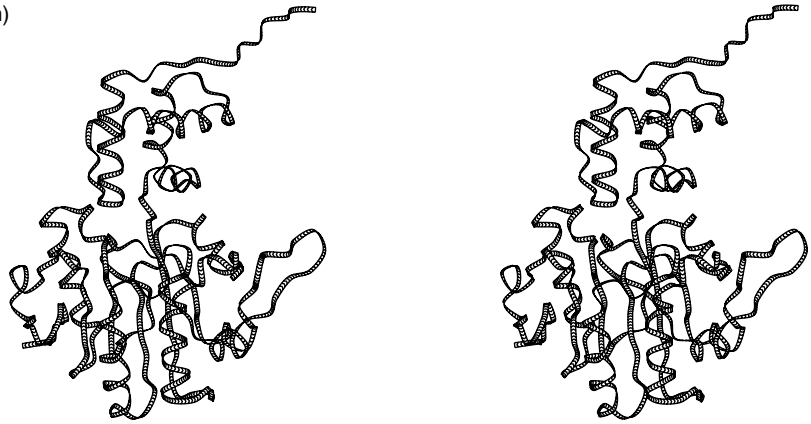


Fig. 1.10 *Continued*

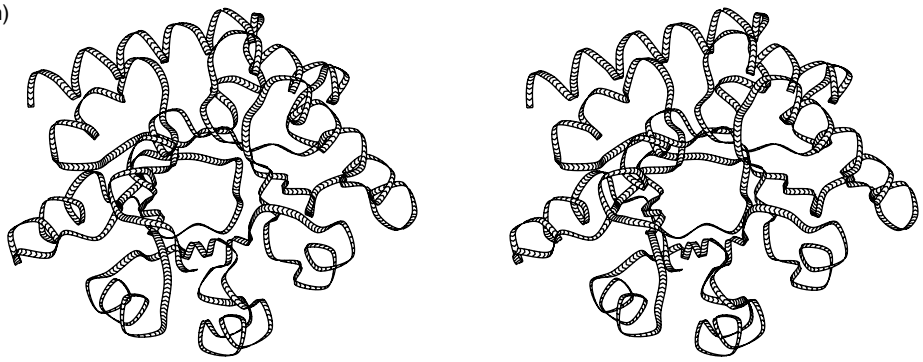
(l)



(m)



(n)



(o)

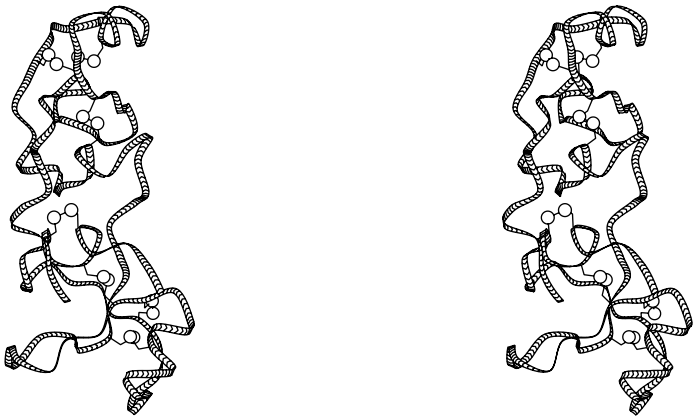


Fig. 1.10 An album of protein structures. (a) engrailed homeodomain [1ENH]; (b) second calponin homology domain from utrophin [1BHD]; (c) HIN recombinase, DNA-binding domain [1HCR]; (d) rice embryo cytochrome c [1CCR]; (e) fibronectin cell-adhesion module type III-10 [1FNA]; (f) mannose-specific agglutinin (lectin) [1NPL]; (g) TATA-box-binding protein core domain [1CDW]; (h) barnase [1BRN]; (i) lysyl-tRNA synthetase [1BBW]; (j) scytalone dehydratase [3STD]; (k) alcohol dehydrogenase, NAD-binding domain [[1EE2].]; (l) adenylate kinase [3ADK]; (m) chemotaxis receptor methyltransferase [1AF7]; (n) thiamin phosphate synthase [2TPS]; (o) porcine pancreatic spasmolytic polypeptide [2PSP].

Protein structure prediction and engineering

The amino acid sequence of a protein dictates its three-dimensional structure. When placed in a medium of suitable solvent and temperature conditions, such as provided by a cell interior, proteins fold spontaneously to their nature active states. Some proteins require chaperones to fold, but these catalyze the process rather than direct it.

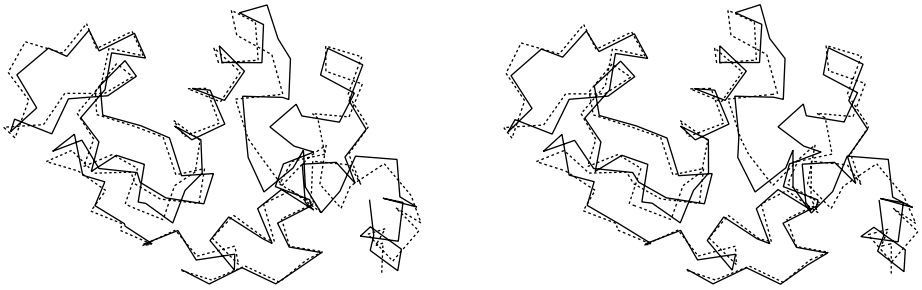
If amino acid sequences contain sufficient information to specify three-dimensional structures of proteins, it should be possible to devise an algorithm to predict protein structure from amino acid sequence. This has proved elusive. In consequence, in addition to pursuing the fundamental problem of a priori prediction of protein structure from amino acid sequence, scientists have defined less-ambitious goals:

1. *Secondary structure prediction:* Which segments of the sequence form helices and which form strands of sheet?
2. *Fold recognition:* Given a library of known protein structures and their amino acid sequences, and the amino acid sequence of a protein of unknown structure, can we find the structure in the library that is most likely to have a folding pattern similar to that of the protein of unknown structure?
3. *Homology modelling:* Suppose a target protein, of known amino acid sequence but unknown structure, is homologous to one or more proteins of known structure. Then we expect that much of the structure of the target

protein will resemble that of the known protein, and it can serve as a basis for a model of the target structure. The completeness and quality of the result depend crucially on how similar the sequences are. As a rule of thumb, if the sequences of two related proteins have 50% or more identical residues in an optimal alignment, the structures are likely to have similar conformations over more than 90% of the model. (This is a conservative estimate, as the following illustration shows.)

Here are the aligned sequences, and superposed structures, of two related proteins, hen egg white lysozyme and baboon α-lactalbumin. The sequences are closely related (37% identical residues in the aligned sequences), and the structures are very similar. Each protein could serve as a good model for the other, at least as far as the course of the mainchain is concerned.

Chicken lysozyme	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRNTDGS
Baboon α-lactalbumin	KQFTKCELSQNLY--DIDGYGRIALPELICTMFHTSGYDTQAIVEND-ES
Chicken lysozyme	TDYGILQINSRWWCNDGRTPGSRNLCNIPCSALLSSDITASVNCAKKIVS
Baboon α-lactalbumin	TEYGLFQISNALWCKSSQSPQSRNICDITCDKFLDDDITDDIMCAKKILD
Chicken lysozyme	DGN-GMNAWVAWRNRCKGTDVQA-WIRGCRL-
Baboon α-lactalbumin	I--KGIDYWIAHKALC-TEKL-EQWL--CE-K



Critical Assessment of Structure Prediction (CASP)

Judging of techniques for predicting protein structures requires blind tests. To this end, J. Moult initiated biennial CASP (Critical Assessment of Structure Prediction) programmes. Crystallographers and NMR spectroscopists in the process of determining a protein structure are invited to (1) publish the amino acid sequence several months before the expected date of completion of their experiment, and (2) commit themselves to keeping the results secret until an agreed date. Predictors submit models, which are held until the deadline for release of the experimental structure. Then the predictions and experiments are compared – to the delight of a few and the chagrin of most.

The results of CASP evaluations record progress in the effectiveness of predictions, which has occurred partly because of the growth of the databanks but also because of improvements in the methods. We shall discuss protein structure prediction in Chapter 5.

Protein engineering

Molecular biologists used to be like astronomers – we could observe our subjects but not modify them. This is no longer true. In the laboratory we can modify nucleic acids and proteins at will. We can probe them by exhaustive mutation to see the effects on function. We can endow old proteins with new functions, as in the development of catalytic antibodies. We can even try to create new ones.

Many rules about protein structure were derived from observations of natural proteins. These rules do not *necessarily* apply to engineered proteins. Natural proteins have features required by general principles of physical chemistry, and by the mechanism of protein evolution. Engineered proteins must obey the laws of physical chemistry but not the constraints of evolution. Engineered proteins can explore new territory.

Clinical implications

There is consensus that the sequencing of the human and other genomes will lead to improvements in the health of mankind. Even discounting some of the more outrageous claims – hype springs eternal – categories of applications include the following.

1. *Diagnosis of disease and disease risks*. DNA sequencing can detect the absence of a particular gene, or a mutation. Identification of specific gene sequences associated with diseases will permit fast and reliable diagnosis of conditions (a) when a patient presents with symptoms, (b) in advance of appearance of symptoms, as in tests for inherited late-onset conditions such as Huntington disease (see Box), (c) for *in utero* diagnosis of potential abnormalities such as cystic fibrosis, and (d) for genetic counselling of couples contemplating having children.

   In many cases our genes do not irrevocably condemn us to contract a disease, but raise the probability that we will. An example of a risk factor detectable at the genetic level involves α_1-antitrypsin, a protein that normally functions to inhibit elastase in the alveoli of the lung. People homozygous for the Z mutant of α_1-antitrypsin (342Glu→Lys), express only a dysfunctional protein. They are at risk of emphysema, because of damage to the lungs from endogenous elastase unchecked by normal inhibitory activity, and also of liver disease, because of accumulation of a polymeric form of α_1-antitrypsin in hepatocytes where it is synthesized. Smoking makes the development of

emphysema all but certain. In these cases the disease is brought on by a *combination* of genetic and environmental factors.

Often the relationship between genotype and disease risk is much more difficult to pin down. Some diseases such as asthma depend on interactions of many genes, as well as environmental factors. In other cases a gene may be all present and correct, but a mutation elsewhere may alter its level of expression or distribution among tissues. Such abnormalities must be detected by measurements of protein activity. Analysis of protein expression patterns is also an important way to measure response to treatment.

2. *Genetics of responses to therapy – customized treatment*. Because people differ in their ability to metabolize drugs, different patients with the same condition may require different dosages. Sequence analysis permits selecting drugs and dosages optimal for individual patients, a fast-growing field called *pharmacogenomics*. Physicians can thereby avoid experimenting with different therapies, a procedure that is dangerous in terms of side effects – often even fatal – and in any case is expensive. Treatment of patients for adverse reactions to prescribed drugs consumes billions of dollars in health care costs.

Huntington disease

Huntington disease is an inherited neurodegenerative disorder affecting approximately 30 000 people in the USA. Its symptoms are quite severe, including uncontrollable dance-like (choreatic) movements, mental disturbance, personality changes, and intellectual impairment. Death usually follows within 10–15 years after the onset of symptoms. The gene arrived in New England during the colonial period, in the seventeenth century. It may have been responsible for some accusations of witchcraft. The gene has not been eliminated from the population, because the age of onset – 30–50 years – is after the typical reproductive period.

Formerly, members of affected families had no alternative but to face the uncertainty and fear, during youth and early adulthood, of not knowing whether they had inherited the disease. The discovery of the gene for Huntington disease in 1993 made it possible to identify affected individuals. The gene contains expanded repeats of the trinucleotide CAG, corresponding to polyglutamine blocks in the corresponding protein, *huntingtin*. (Huntington disease is one of a family of neurodegenerative conditions resulting from trinucleotide repeats.) The larger the block of CAGs, the earlier the onset and more severe the symptoms. The normal gene contains 11–28 CAG repeats. People with 29–34 repeats are unlikely to develop the disease, and those with 35–41 repeats may develop only relatively mild symptoms. However people with >41 repeats are almost certain to suffer full Huntington disease.

The inheritance is marked by a phenomenon called *anticipation:* the repeats grow longer in successive generations, progressively increasing the severity of the disease and reducing the age of onset. For some reason this effect is greater in paternal than in maternal genes. Therefore, even people in the borderline region, who might bear a gene containing 29–41 repeats, should be counselled about the risks to their offspring.

For example, the very toxic drug 6-mercaptopurine is used in the treatment of childhood leukaemia. A small fraction of patients used to die from the treatment, because they lack the enzyme thiopurine methyltransferase, needed to metabolize the drug. Testing of patients for this enzyme identifies those at risk.

Conversely, it may become possible to use drugs that are safe and effective in a minority of patients, but which have been rejected before or during clinical trials because of inefficacy or severe side effects in the majority of patients.

3. *Identification of drug targets.* A *target* is a protein the function of which can be selectively modified by interaction by a drug, to affect the symptoms or underlying causes of a disease. Identification of a target provides the focus for subsequent steps in the drug design process. Among drugs now in use, the targets of about half are receptors, about a quarter enzymes, and about a quarter hormones. Approximately 7% act on unknown targets.

   The growth in bacterial resistance to antibiotics is creating a crisis in disease control. There is a very real possibility that our descendants will look back at the second half of the twentieth century as a narrow window during which bacterial infections could be controlled, and before and after which they could not.

   The urgency of finding new drugs is mitigated by the availability of data on which to base their development. Genomics can suggest targets. *Differential* genomics, and comparison of protein expression patterns, between drug-sensitive and resistant strains of pathogenic bacteria can pinpoint the proteins responsible for drug resistance. The study of genetic variation between tumour and normal cells can, it is hoped, identify differentially expressed proteins as potential targets for anticancer drugs.

4. *Gene therapy.* If a gene is missing or defective, we would like to replace it or at least supply its product. If a gene is overactive, we would like to turn it off.

   Direct supply of proteins is possible for many diseases, of which insulin replacement for diabetes and Factor VIII for a common form of haemophilia are perhaps the best known.

   Gene transfer has succeeded in animals, for production of human proteins in the milk of sheep and cows. In human patients, gene replacement therapy for cystic fibrosis using adenovirus has shown encouraging results.

   One approach to blocking genes is called 'antisense therapy'. The idea is to introduce a short stretch of DNA or RNA that binds in a sequence-specific manner to a region of a gene. Binding to endogenous DNA can interfere with transcription; binding to mRNA can interfere with translation. Antisense therapy has shown some efficacy against cytomegalovirus and Crohn disease.

   Antisense therapy is very attractive, because going directly from target sequence to blocker short-circuits many stages of the drug-design process.

The future

The new century will see a revolution in healthcare development and delivery. Barriers between 'blue sky' research and clinical practice are tumbling down. It is possible that a reader of this book will discover a cure for a disease that would otherwise kill him or her. Indeed it is extremely likely that Szent-Gyorgi's quip, 'Cancer supports more people than it kills' will come true. One hopes that this happens because the research establishment has succeeded in developing therapeutic or preventative measures against tumours rather than merely by imitating their uncontrolled growth.

WEB RESOURCE:

For general background:

D. Casey of Oak Ridge National Laboratory has written two extremely useful *compact* introductions to molecular biology providing essential background for bioinformatics: *Primer on Molecular Genetics* (1992). Washington, DC: Human Genome Program, US Department of Energy.

http://www.bis.med.jhmi.edu/Dan/DOE/intro.html

Human Genome Project Information:

http://www.ornl.gov/hgmis/project/info.html

Genome statistics:

http://bioinformatics.weizmann.ac.il/mb/statistics.html

Taxonomy sites:

Species 2000 – a comprehensive index of all known plants, animals, fungi and microorganisms: http://www.sp2000.org

Tree of life – phylogeny and biodiversity: http://phylogeny.arizona.edu/tree

Databases of genetics of disease:

http://www.ncbi.nlm.nih.gov/omim/
http://www.geneclinics.org/profiles/all.html

Lists of databases:

http://www.infobiogen.fr/services/dbcat/
http://www.ebi.ac.uk/biocat/

List of tools for analysis:

http://www.ebi.ac.uk/Tools/index.html

Debate on electronic access to the scientific literature:

http://www.nature.com/nature/debates/e-access/)

Recommended reading

A glimpse of the future?

Blumberg, B.S. (1996) 'Medical research for the next millenium', *The Cambridge Review* 117, 3–8. [A fascinating prediction of things to come, some of which are already here.]

The transition to electronic publishing

Lesk, M. (1997) *Practical Digital Libraries: Books, Bytes and Bucks* (San Francisco: Morgan Kaufmann). [Introduction to the transition from traditional libraries to information provision by computer.]

Berners-Lee, T and Hendler, J. (2001) 'Publishing on the semantic web', *Nature* 410, 1023–4. [Comments from the inventor of the web.]

Butler, D. and Campbell, P. (2001) 'Future e-access to the primary literature', *Nature* 410, 613. [Describes developments in electronic publishing of scientific journals.]

Doolittle, W.F. (2000) 'Uprooting the tree of life', *Scientific American* 282(2), 90–5. [Implications of analysis of sequences for our understanding of the relationships between living things.]

Genomic sequence determination

Green, E.D. (2001) 'Strategies for systematic sequencing of complex organisms', *Nature Reviews (Genetics)* 2, 573–83. [A clear discussion of possible approaches to large-scale sequencing projects. Includes list of and links to ongoing projects for sequencing of multicellular organisms.]

Sulston, J. and Ferry, G. (2002) *The common thread: a story of science, politics, ethics, and the human genome* (New York: Bantam). [A first hand account.]

More about protein structure

Branden, C.-I. and Tooze, J. (1999). *Introduction to Protein Structure,* 2nd. ed. (New York: Garland). [A fine introductory text.]

Lesk, A.M. (2000) *Introduction to Protein Architecture: The Structural Biology of Proteins* (Oxford: Oxford University Press). [Companion volume to Introduction to Bioinformatics, with a focus on protein structure and evolution.]

Discussions of databases

Frishman, D., Heumann, K., Lesk, A, and Mewes, H.-W. (1998) 'Comprehensive, comprehensible, distributed and intelligent databases: current status', *Bioinformatics* 14, 551–61. [Status and problems of organization of information in molecular biology.]

Lesk, A.M. and 25 co-authors. (2001) 'Quality control in databanks for molecular biology', *BioEssays* 22, 1024–34. [Treats the problems and possible developments in assuring adequate quality in the data archives on which we all depend.]

Stein, L. (2001) 'Genome annotation: from sequence to biology', *Nature Reviews (Genetics)* 2, 493–503. [Also emphasizes the importance of annotation.]

Legal aspects of patenting

Human Genome Project Information Website: Genetics and Patenting
`http://www.ornl.gov/hgmis/elsi/patents.html`

Maschio, T. and Kowalski, T. (2001) 'Bioinformatics – a patenting view', *Trends in Biotechnology* 19, 334–9.

Caulfield, T., Gold, E.R., and Cho, M.K. (2000) 'Patenting human genetic material: refocusing the debate', *Nature Reviews (Genetics)* 1, 227–31. [Discussions of legal aspects of genomics and bioinformatics. (1) genes, (2) algorithms – computer methods, and (3) computer programs are subject to patent or copyright.]

Exercises, Problems, and Weblems

Exercise 1.1 (a) The Sloan Digital Sky Survey is a mapping of the Northern sky over a 5-year period. The total raw data will exceed 40 terabytes (1 byte = 1 character; 1TB = 10^{12} bytes). How many human genome equivalents does this amount to? (b) The Earth Observing System/Data Information System (EOS/DIS) – a series of long-term global observations of the Earth – is estimated to require 15 petabytes of storage (1 petabyte = 10^{15} bytes.) How many human genome equivalents will this amount to? (c) Compare the data storage required for EOS/DIS with that required to store the complete DNA sequences of every inhabitant of the USA. (Ignore savings available using various kinds of storage compression techniques. Assume that each person's DNA sequence requires 1 byte/nucleotide.)

Exercise 1.2 (a) How many floppy disks would be required to store the entire human genome? (b) How many CDs would be required to store the entire human genome? (c) How many DVDs would be required to store the entire human genome? (In all cases assume that the sequence is stored as 1 byte/per character, uncompressed.)

Exercise 1.3 Suppose you were going to prepare the Box on Huntington disease (page 51) for a web site. For which words or phrases would you provide links?

Exercise 1.4 The end of the human β-haemoglobin gene has the nucleotide sequence:

```
... ctg gcc cac aag tat cac taa
```

(a) What is the translation of this sequence into an amino acid sequence? (b) Write the nucleotide sequence of a single base change producing a silent mutation in this region. (A silent mutation is one that leaves the amino acid sequence unchanged.) (c) Write the nucleotide sequence, and the translation to an amino acid sequence, of a single base change producing a missense mutation in this region. (d) Write the nucleotide sequence, and the translation to an amino acid sequence, of a single base change producing a mutation in this region that would lead to premature truncation of the protein. (e) Write the nucleotide sequence of a single base change producing a mutation in this region that would lead to improper chain termination resulting in extension of the protein.

Exercise 1.5 On a photocopy of the Box *Complete pairwise sequence alignment of human PAX-6 protein and* Drosophila melanogaster *eyeless*, indicate with a highlighter the regions aligned by PSI-BLAST.

Exercise 1.6 (a) What cutoff value of *E* would you use in a PSI-BLAST search if all you want to know is whether your sequence is already in a databank? (b) What cutoff

value of E would you use in a PSI-BLAST search if you want to locate distant homologues of your sequence?

Exercise 1.7 In designing an antisense sequence, estimate the minimum length required to avoid exact complementarity to many random regions of the human genome.

Exercise 1.8 It is suggested that all living humans are descended from a common ancestor called Eve, who lived approximately 140 000–200 000 years ago. (a) Assuming six generations per century, how many generations have there been between Eve and the present? (b) If a bacterial cell divides every 20 min, how long would be required for the bacterium to go through that number of generations?

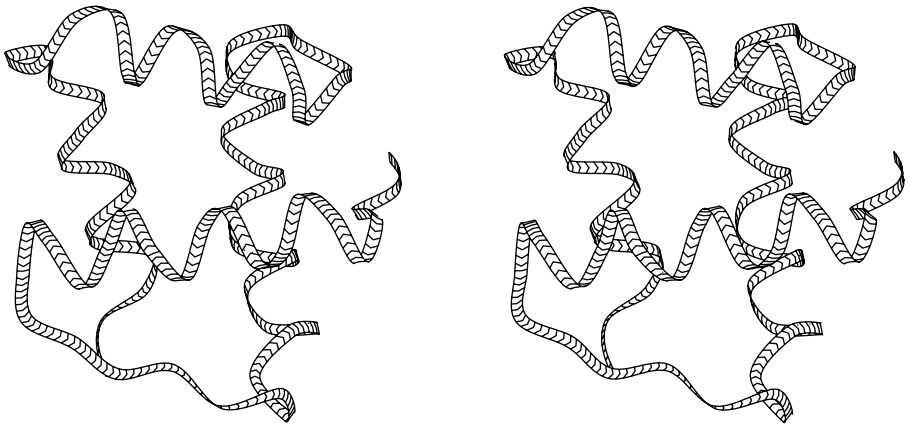
Exercise 1.9 Name an amino acid that has physicochemical properties similar to (a) leucine, (b) aspartic acid, (c) threonine? We expect that such substitutions would in most cases have relatively little effect on the structure and function of a protein. Name an amino acid that has physicochemical properties very different from (d) leucine, (e) aspartic acid, (f) threonine? Such substitutions might have severe effects on the structure and function of a protein, especially if they occur in the interior of the protein structure.

Exercise 1.10 In Fig. 1.7(a), does the direction of the chain from N-terminus to C-terminus point up the page or down the page? In Fig. 1.7(b), does the direction of the chain from N-terminus to C-terminus point up the page or down the page?

Exercise 1.11 From inspection of Fig. 1.9, how many times does the chain pass between the domains of *M. jannaschii* ribosomal protein L1?

Exercise 1.12 On a photocopy of Fig. 1.10(k and l), indicate with highlighter the helices (in red) and strands of sheet (in blue). On a photocopy of Fig. 1.10(g and m), divide the protein into domains.

Exercise 1.13 Which of the structures shown in Fig. 1.10 contains the following domain?



Exercise 1.14 On a photocopy of the superposition of Chicken lysozyme and Baboon α-lactalbumin structures, indicate with a highlighter two regions in which the conformation of the mainchain is different.

Exercise 1.15 In the PERL program on page 18, estimate the fraction of the text of the program that contains comment material. (Count full lines and half lines.)

Exercise 1.16 Modify the PERL program that extracts species names from PSI-BLAST output so that it would also accept names given in the form `[D. melanogaster]`

Exercise 1.17 What is the nucleotide sequence of the molecule shown in Plate I?

Problem 1.1 The following table contains a multiple alignment of partial sequences from a family of proteins called ETS domains. Each line corresponds to the amino acid sequence from one protein, specified as a sequence of letters each specifying one amino acid. Looking down any column shows the amino acids that appear at that position in each of the proteins in the family. In this way patterns of preference are made visible.

```
TYLWEFLLKLLQDR.EYCPRFIKWTNREKGVFKLV..DSKAVSRLWGMHKN.KPD
VQLWQFLLEILTD..CEHTDVIEWVG.TEGEFKLT..DPDRVARLWGEKKN.KPA
IQLWQFLLELLTD..KDARDCISWVG.DEGEFKLN..QPELVAQKWGQRKN.KPT
IQLWQFLLELLSD..SSNSSCITWEG.TNGEFKMT..DPDEVARRWGERKS.KPN
IQLWQFLLELLTD..KSCQSFISWTG.DGWEFKLS..DPDEVARRWGKRKN.KPK
IQLWQFLLELLQD..GARSSCIRWTG.NSREFQLC..DPKEVARLWGERKR.KPG
IQLWHFILELLQK..EEFRHVIAWQQGEYGEFVIK..DPDEVARLWGRRKC.KPQ
VTLWQFLLQLLRE..QGNGHIISWTSRDGGEFKLV..DAEEVARLWGLRKN.KTN
ITLWQFLLHLLLD..QKHEHLICWTS.NDGEFKLL..KAEEVAKLWGLRKN.KTN
LQLWQFLVALLDD..PTNAHFIAWTG.RGMEFKLI..EPEEVARLWGIQKN.RPA
IHLWQFLKELLASP.QVNGTAIRWIDRSKGIFKIE..DSVRVAKLWGRRKN.RPA
RLLWDFLQQLLNDRNQKYSDLIAWKCRDTGVFKIV..DPAGLAKLWGIQKN.HLS
RLLWDYVYQLLSD..SRYENFIRWEDKESKIFRIV..DPNGLARLWGNHKN.RTN
IRLYQFLLDLLRS..GDMKDSIWWVDKDKGTFQFSSKHKEALAHRWGIQKGNRKK
LRLYQFLLGLLTR..GDMRECVWWVEPGAGVFQFSSKHKELLARRWGQQKGNRKR
```

In your personal copy of this book:

(a) Using coloured highlighter, mark, in each sequence, the residues in different classes in different colours:

small residues	G A S T
medium-sized nonpolar residues	C P V I L
large nonpolar residues	F Y M W
polar residues	H N Q
positively-charged residues	K R
negatively-charged residues	D E

(b) For each position containing the same amino acid in every sequence, write the letter symbolizing the common residue in upper case below the column. For each position containing the same amino acid in all but one of the sequences, write the letter symbolizing the preferred residue in lower case below the column.

(c) What patterns of periodicity of conserved residues suggest themselves?

(d) What secondary structure do these patterns suggest in certain regions?

(e) What distribution of conservation of charged residues do you observe? Propose a reasonable guess about what kind of molecule these domains interact with.

Problem 1.2 Classify the structures appearing in Fig. 1.10 in the following categories: α-helical, β-sheet, $\alpha + \beta$, α/β linear, α/β-barrels, little or no secondary structure.

Problem 1.3 Generalize the PERL program on page 16 to print the translations of a DNA sequence in all six possible reading frames.

Problem 1.4 For what of following sets of fragment strings does the PERL program on page 18 work correctly?

(a) Would it correctly recover:

Kate, when France is mine and I am
yours, then yours is France and you are mine.

from:

Kate, when France
France is mine
is mine and
and I am\nyours
yours then
then yours is France
France and you are mine\n

(b) Would it correctly recover:

One woman is fair, yet I am well; another is wise, yet I am well; another virtuous, yet I am well; but till all graces be in one woman, one woman shall not come in my grace.

from:

One woman is
woman is fair,
is fair, yet I am
yet I am well;
I am well; another
another is wise, yet I am well;
yet I am well; another virtuous,
another virtuous, yet I am well;
well; but till all
all graces be
be in one woman,
one woman, one
one woman shall
shal not come in my grace.

(c) Would it correctly recover:

That he is mad, 'tis true: 'tis true 'tis pity;
And pity 'tis 'tis true.

from:

That he is
is mad, 'tis
'tis true
true: 'tis true 'tis
true 'tis
'tis pity;\n
pity;\nAnd pity
pity 'tis
'tis 'tis
'tis true.\n

In (c), would it work if you deleted all punctuation marks from these strings?

Problem 1.5 Generalize the PERL program on page 18 so that it will correctly assemble all the fragments in the previous problem. (Warning – this is not an easy problem.)

Problem 1.6 Write a PERL program to find motif matches as illustrated in the Box on page 24. (a) Demand exact matches. (b) Allowing one mismatch, not necessarily at the first position as in the examples, but no insertions or deletions.

Problem 1.7 PERL is capable of great concision. Here is an alternative version of the program to assemble overlapping fragments (see page 18):

```
#!/usr/bin/perl

$/ = "";
@fragments = split("\n",<DATA>);

foreach (@fragments) { $firstfragment{$_} = $_; }

foreach $i (@fragments) {
    foreach $j (@fragments) { unless ($i eq $j) {
        ($combine = $i . "XXX" . $j) =~ /([\S ]{2,})XXX\1/;
        (length($1) <= length($successor{$i})) || { $successor{$i} = $j };
    }                          }
    undef $firstfragment{$successor{$i}};
}

$test = $outstring = join "", values(%firstfragment);
while ($test = $successor{$test}) { ($outstring .= "XXX" . $test) =~ s/([\S ]+)XXX\1/\1/; }

$outstring =~ s/\\n/\n/g; print "$outstring\n";

__END__
the men and women merely players;\n
one man in his time
All the world's
their entrances,\nand one man
stage,\nAnd all the men and women
They have their exits and their entrances,\n
world's a stage,\nAnd all
their entrances,\nand one man
in his time plays many parts.
merely players;\nThey have
```

(This is a good example of what to avoid. Anyone who produces code like this should be fired immediately. The absence of comments, and the tricky coding

and useless brevity, make it difficult to understand what the program is doing. A program written in this way is difficult to debug and virtually impossible to maintain. Someday you may succeed someone in a job and be presented with such a program to work on. You will have my sympathy.)

(a) Photocopy the concise program listed in this problem and the original version on page 18 so that they appear side-by-side on a page. Wherever possible, map each line of the concise program into the corresponding set of lines of the long one.

(b) Prepare a version of the concise program with enough comments to clarify what it is doing (for this you could consider adapting the comments from the original program) and how it is doing it. Do not change any of the executable statements (back to the original version or to anything else); just add comments.

Weblem 1.1 Identify the source of all quotes from Shakespeare's plays in the Box on alignment.

Weblem 1.2 Identify web sites that give *elementary* tutorial explanations and/or on-line demonstrations of (a) the Polymerase Chain Reaction (PCR), (b) Southern blotting, (c) restriction maps, (d) cache memory, (e) suffix tree. Write a one-paragraph explanation of these terms based on these sites.

Weblem 1.3 To which phyla do the following species belong? (a) Starfish. (b) Lamprey (c) Tapeworm (d) Ginko tree (e) Scorpion. (f) Jellyfish. (g) Sea anemone.

Weblem 1.4 What are the common names of the following species? (a) *Acer rubrum*. (b) *Orycteropus afer*. (c) *Beta vulgaris*. (d) *Pyractomena borealis*. (e) *Macrocystis pyrifera*.

Weblem 1.5 A typical British breakfast consists of: eggs (from chickens) fried in lard, bacon, kippered herrings, grilled cup mushrooms, fried potatoes, grilled tomatoes, baked beans, toast, and tea with milk. Write the complete taxonomic classification of the organisms from which these are derived.

Weblem 1.6 Recover and align the mitochondrial cytochrome *b* sequences from horse, whale and kangaroo. (a) Compare the degree of similarity of each pair of sequences with the results from comparison of the pancreatic ribonuclease sequences from these species in Example 1.2. Are the conclusions from the analysis of mitochondrial cytochromes *b* sequences consistent with those from analysis of the pancreatic ribonucleases? (b) Compare the *relative* similarity of these sequences with the results from comparison of the pancreatic ribonuclease sequences from these species in Example 1.2. Are the conclusions from the analysis of mitochondrial cytochromes *b* sequences consistent with those from analysis of the pancreatic ribonucleases?

Weblem 1.7 Recover and align the pancreatic ribonuclease sequences from sperm whale, horse, and hippopotamus. Are the results consistent with the relationships shown by the SINES?

Weblem 1.8 We observed that the amino acid sequences of cytochrome *b* from elephants and the mammoth are very similar. One hypothesis to explain this observation is that a functional cytochrome *b* might *require* so many conserved residues that

cytochromes *b* from all animals are as similar to one another as the elephant and mammoth proteins are. Test this hypothesis by retrieving cytochrome *b* sequences from other mammalian species, and check whether the cytochrome *b* amino acid sequences from more distantly-related species are as similar as the elephant and mammoth sequences.

Weblem 1.9 Recover and align the cytochrome *c* sequences of human, rattlesnake, and monitor lizard. Which pair appears to be the most closely related? Is this surprising to you? Why or why not?

Weblem 1.10 Send the sequences of pancreatic ribonucleases from horse, minke whale, and red kangaroo (Example 1.2) to the T-coffee multiple-alignment server: `http://www.ch.embnet.org/software/TCoffee.html` Is the resulting alignment the same as that shown in Example 1.2 produced by CLUSTAL–W?

Weblem 1.11 Linnaeus divided the animal kingdom into six classes: mammals, birds, amphibia (including reptiles), fishes, insects and worms. This implies, for instance, that he considered crocodiles and salamanders more closely related than crocodiles and birds. Thomas Huxley, on the other hand, in the nineteenth century, grouped reptiles and birds together. For three suitable proteins with homologues in crocodiles, salamanders and birds, determine the similarity between the homologous sequences. Which pair of animal groups appears most closely related? Who was right, Linnaeus or Huxley?

Weblem 1.12 When was the last new species of primate discovered?

Weblem 1.13 In how many more species have PAX-6 homologues been discovered since the table on pages 38–9 was compiled?

Weblem 1.14 Identify three modular proteins in addition to fibronectin itself that contain fibronectin III domains.

Weblem 1.15 Find six examples of diseases other than diabetes and haemophilia that are treatable by administering missing protein directly. In each case, what protein is administered?

Weblem 1.16 To what late-onset disease are carriers of a variant apolipoprotein E gene at unusually high risk? What variant carries the highest risk? What is known about the mechanism by which this variant influences the development of the disease?

Weblem 1.17 For approximately 10% of Europeans, the painkiller codeine is ineffective because the patients lack the enzyme that converts codeine into the active molecule, morphine. What is the most common mutation that causes this condition?