20

Computer Analysis of Cloned Sequences

Paul R. Caron

1. Introduction

1.1. Goals of Analyses

Analysis of the data generated from cDNA sequences, an important step in the final stages of any sequencing project, can provide insights into gene structure and function as well as help to direct the experimental approaches to obtaining the sequence of a full-length clone even when only preliminary data are available. The types of analyses that should be performed will depend on the immediate goal, and may evolve from trying to identify whether a clone is genuine or just an artifact to the stage where one would like to draw conclusions about function and domain structure of a previously unknown gene product encoded by the cDNA clone. This chapter provides an overview of the types of analyses that should be performed to extract the most information from a sequence. No attempt is made to review the current status of sequence analysis software, but particular programs are used to illustrate the general approaches that should be taken.

1.2. Software Choices

There are currently sequence analysis software programs for each of the major computer operating systems. Although many of these software packages have very useful functions, the exponential growth of the sequence databases necessitates the use of centralized computing facilities, which are able to handle the large volume of data associated with the sequence databases as well as provide the computing power necessary to search these data rapidly. These resources may be databases and programs accessible to the users at individual sites or network-based program servers. This chapter discusses both approaches, using the programs that are part of the Genetics Computer Group's (GCG's) (Madi-

From *Methods in Molecular Biology, Vol 69: cDNA Library Protocols*
Edited by I G Cowell and C A Austin Humana Press Inc , Totowa, NJ

son, WI) Wisconsin Sequence Analysis Package™ for most of the examples *(1)*. Section 4. contains pointers for accessing network-based searching services. Although many other implementations of most of the functions in the GCG package exist, this package is widely available and provides a consistent interface to a multitude of sequence analysis programs. This package is available for VMS and a number of UNIX operating systems. This chapter specifically discusses using the UNIX command-line interface to GCG release 8.0 Section 4. contains a summary of the differences between the UNIX and VMS implementations. No discussion of any graphical interface to the GCG programs or any similar programs is presented.

This chapter presents the general methodology needed to characterize a recently cloned sequence. It is impossible in this brief summary to cover thoroughly all of the options for the programs discussed, those interested in more detailed information should consult *Methods in Molecular Biology*, vol. 24 in this series *(2)* as well as the original program documentation. Comparisons between the various databases' searching techniques can be found in Altschul et al. *(3)*. Throughout this chapter, text in **bold** should be entered by the user, <return> is used to denote pressing the "return" or "enter" key; <ctrl-d> signifies pressing "d" while holding down the "control" key; and the user should replace <filename> with the name of his or her file.

2. Materials

Most of the programs discussed in this chapter are part of release 8.0 of the Wisconsin Sequence Analysis Package available from the GCG. The examples were run under IRIX 5.0 on a Silicon Graphics computer. However, the commands should be virtually identical under all of the UNIX operating systems currently supported by GCG. Translation of the commands to VMS is straightforward and is discussed in Section 4. A VT100 terminal connection to the host enables users with a variety of desktop computers and workstations to use the programs similarly. Unless specifically stated, the default values were used for all optional parameters in the examples presented.

Macaw *(4)* and Nentrez are available from the National Center for Biotechnology Information (NCBI) in both Mac and Windows versions. Macaw version 2.0.5 and Nentrez version 3.015 were obtained by anonymous ftp from **ncbi.nlm.nih.gov**. Local copies of the sequence databases were obtained from the NCBI and reformatted with the tools provided with the GCG package. Remote databases were accessed from the NCBI. Direct access to the internet is required to use programs, such as Nentrez or Network Blast, both of which also require preregistration with the NCBI. Access to these programs is also available without preregistration through the World Wide Web (WWW) at **http://www.ncbi.nlm.nih.gov**.

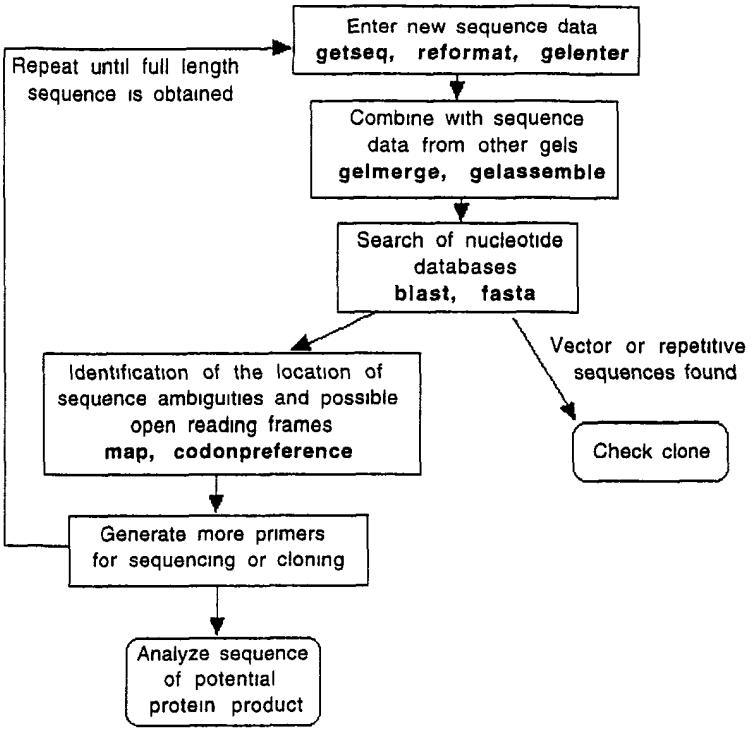

Fig. 1. Initial characterization of nucleotide sequence data.

3. Methods

The sequence analysis problem can be divided into two phases as depicted in Figs. 1 and 2. The first phase involves obtaining a full-length cDNA with high confidence in the reliability of the nucleotide sequence. In the second phase, one attempts to predict what this cDNA encodes, and how much can be extrapolated about the structure and function of the presumed protein product. The flowcharts presented in these two figures display the general approaches to the problems, whereas specific programs are discussed in the text.

3.1. Data Entry/Fragment Assembly

Sequence data can be entered into a computer file either automatically from a DNA sequencing system or manually using custom software or a word processor. The DNA sequence should be saved either directly in a GCG format or as a plain text file. The file containing the sequence data can be transferred directly to the system running GCG and "reformatted," or for short sequences, the sequence data can be "cut and pasted" into GCG-compatible files. Throughout this chapter, it is assumed that files are in a compatible format for subse-

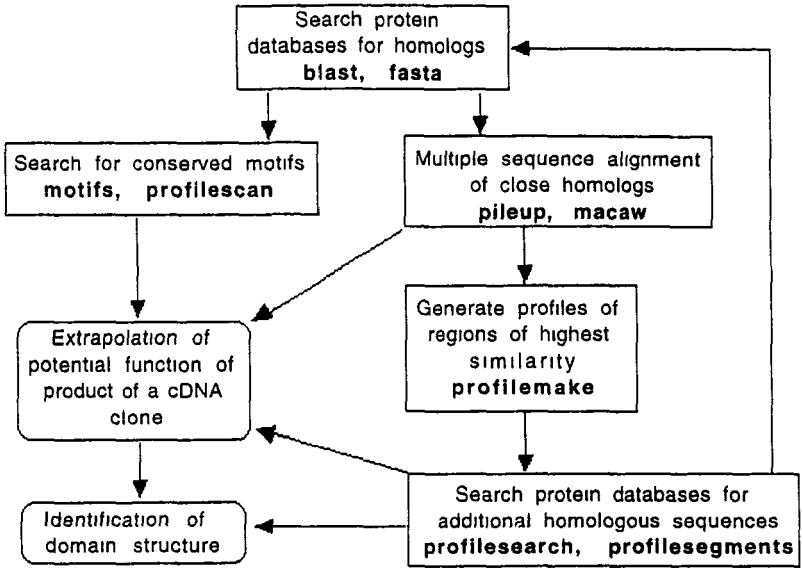

Fig. 2. Identification of structure and function of protein product

quent analysis. Type **reformat** <filename> to format a noncompliant text file into a GCG-compatible format. Tools are also available to convert files directly from GenBank, EMBL, FASTA, PIR, Staden, and IntelliGenetics formats.

3.1.1. Entering a New Sequence Directly into a File

Type **getseq** <return>, followed by the name of a file in which to store the sequence. Paste the sequence, followed by <ctrl-D>. Nucleic acid sequence can be represented by any valid IUPAC symbols.

3.1.2. Assembling Multiple Fragments

Often sequence data are obtained from multiple gels that result from sequencing-related clones and the use of primers originating on both strands of the DNA. Depending on the size and complexity of the sequence project, it may be possible to assemble these data into a single consensus sequence using the gel reading software or a word processor. At other times, it will be necessary to use fragment assembly software to derive a consensus sequence. Details for using the GCG fragment assembly tools can be found in Chapter 2 of *Methods in Molecular Biology*, vol. 24, as well as in the GCG program manual.

3.2. Restriction Site Mapping

The location of commercially available restriction sites within a sequence are valuable landmarks that can be used to compare locations within a sequence

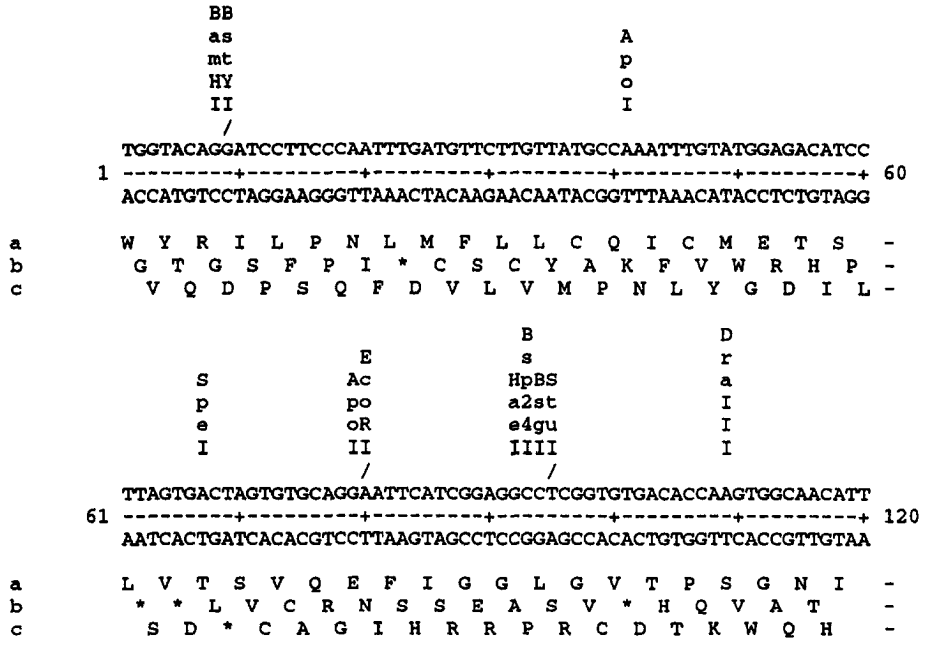

Fig. 3. Example of the output from **map**. The command **map -six** was used to locate the position of restriction enzyme cleavage sites in the sequence. Amino acid translations in the three forward reading frames is shown. Asterisks indicate the location of stop codons.

with DNA fragment sizes obtained while cloning. Identification of new sites in the sequence would provide additional handles that can be used to manipulate the clone further.

3.2.1. Generating a Linear Restriction Map

The **map** command will generate a linear restriction map of a nucleotide sequence and can display the translated amino acid sequence below it. This type of display allows one to visualize the correspondence of the nucleotide sequence with open reading frames and stop codons. An example of the output of this command, in which the restriction enzyme list was limited to enzymes with six or more nucleotides in their recognition site, is shown in Fig. 3. Files other than the default can be used as a source of enzyme or factor recognition specificities and amino acid translation schemes by command line specification.

3.2.2. Generating a List of Restriction Sites

A tabular list of the location of restriction enzyme sites can be generated by the command **mapsort**. Similar to the **map** command, the file that defines cleavage specificity can be defined at the command line.

3.3. Finding Open Reading Frames

Viewing the translated amino acid sequence generated by the **map** command is useful in identifying stop codons in a newly sequenced region and locating regions in the cDNA sequence that contain potential frame shifts. Searching protein sequences against all reading frames of a DNA sequence using programs, such as **blast**, can also identify possible frame shifts. Other statistical analyses of the DNA sequence can also be used to identify the correct translation frame(s).

3.3.1. Codonpreference

Each organism has a set of codons that are used preferentially when expressing proteins of relatively high abundance. This bias can be used to locate the coding vs noncoding reading frames of a sequence. The program **codonpreference** uses the information contained in codon usage tables for a given organism to identify regions of the sequence that most likely encode protein *(5)*. Codon usage tables for a variety of organisms from *Escherichia coli* to human are available in the GCG package. Tables of any specificity can be generated using the program **codonfrequency** *See* Section 4. for details on configuring the GCG package for graphic output before running **codonpreference**.

3.3.2. Testcode

An additional measure of the potential of a region to encode a protein can be obtained by analyzing the distribution of nucleotides in every third position. The degeneracy of the genetic code allows this position to have much more variability than the others while conserving the amino acid sequence. The program **testcode** uses a statistical analysis of the nucleotide sequence to predict the location of coding vs noncoding regions *(6)*.

3.3.3. Translate

The protein sequence can be extracted from the cDNA sequence once the proper open reading frame has been identified using the **translate** program. This program requires that one know the nucleotide number of the starting and ending codon.

3.4. Homology with Other Proteins/Genes

3.4.1. Local vs Network Searching

Probably the most important question to ask when obtaining a new sequence is "What is this sequence similar to?" One would like to know if this is a homolog of, or identical to, a previously identified sequence. Obviously, one

would like to perform a thorough search using the most recent and complete databases. Unfortunately, with the exponential growth of these data and daily new releases, maintaining an up-to-date local copy of these databases requires much time and storage space. Thus, to make the most efficient use of resources, there has been a shift toward using network servers to perform the searches. Searching local databases has advantages in terms of complete control of data security, response time, and unlimited number of queries.

3.4.2. Blast

Blast searches are very fast searches used to find local segments of homology; no gaps in the aligned segments are allowed *(7)*. The GCG implementation of **blast** allows searching of local databases as well as remote databases using servers at NCBI. This program can be used to query nucleotide sequences against nucleotide sequence databases, or the query sequence can be translated into amino acid sequences to search protein sequence databases. Protein query sequences can likewise be used to search either the protein or nucleotide databases. Starting with a nucleotide query sequence, blast searching of nucleotide sequences will find very closely related sequences, whereas searching of protein sequences will potentially identify more distant homologs. This is apparent in the results of **blast** searches using a sample sequence in Fig. 4. An estimate of the probability of a match is calculated based on the highest scoring match, but caution should be used when interpreting this number owing to repetitive sequences and contamination by vector sequences.

3.4.3. Fasta

The **fasta** program allows sequence searching for regions of local homology; unlike **blast**, gaps are allowed *(8)*. This program is more sensitive than **blast** when searching nucleotide databases. Another advantage is that the GCG version of **fasta** is able to search local sequence databases in the format used by the other GCG programs. The **blast** program requires that local databases be reformatted before searching, which increases the amount of storage space processing time required to maintain up-to-date data.

3.5. Retrieving Homologous Sequences and Descriptions

3.5.1. Fetch

Copies of sequences of interest that are stored in a local copy of a sequence database can be obtained using the **fetch** program. Typing **fetch** followed by the accession number or sequence ID will create a file containing the complete database entry. The command **fetch -outf=term:** or **fetch -outf=gcgstdout** will retrieve the sequence entry to the terminal window.

A

Sequences producing High-scoring Segment Pairs:		High Score	Smallest Sum Probability P(N)	N
gb|U07681|HSU07681	Human NAD(H)-specific isocitrate deh...	383	1.6e-35	2
gb|U07980|BTU07980	Bos taurus NAD+-dependent isocitrate...	329	1.7e-29	2
emb|X67310|TBBISODEH	S.tuberosum mRNA for beta-isopropylm...	179	6.2e-06	1
gb|M33099|BACIPMD	B.coagulans 3-isopropylmalate dehydr...	110	0.9999	1

B

sp|P41563|IDH3_BOVIN	ISOCITRATE DEHYDROGENASE (NAD),...	+3	131	3.1e-17	2
gp|U07681|HSU07681_1	NAD(H)-specific isocitrate dehy...	+3	131	3.5e-17	2
sp|P29696|LEU3_SOLTU	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	110	1.2e-12	2
sp|P40495|YIJ4_YEAST	HYPOTHETICAL 40.1 KD PROTEIN IN...	+3	84	4.4e-08	2
pir|S20606|S20606	leuB protein - Salmonella typhi...	+3	98	1.4e-06	1
sp|P30125|LEU3_ECOLI	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	98	1.4e-06	1
sp|P37412|LEU3_SALTY	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	98	1.4e-06	1
pir|A43934|A43934	isocitrate dehydrogenase (NADP+...	+3	71	1.5e-06	2
sp|P33197|IDH_THETH	ISOCITRATE DEHYDROGENASE (NADP)...	+3	71	1.5e-06	2
gp|U07940|LPU07940_3	DlpA [Legionella pneumophila]	+3	72	2.0e-06	2
sp|P41564|IDHG_MACFA	ISOCITRATE DEHYDROGENASE (NAD),...	+3	96	2.7e-06	1
sp|P41565|IDHG_RAT	ISOCITRATE DEHYDROGENASE (NAD),...	+3	96	2.8e-06	1
pir|S39064|S39064	isocitrate dehydrogenase (NAD+)...	+3	96	2.8e-06	1
sp|Q02143|LEU3_LACLA	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	78	1.1e-05	2
gp|M90761|LACLEUILV_2	leuB gene product [Lactococcus ...	+3	78	1.1e-05	2
sp|P28241|IDH2_YEAST	ISOCITRATE DEHYDROGENASE (NAD),...	+3	61	5.8e-05	2
gp|Z46242|CEF35G12_2	F35G12.2, similar to isocitrate...	+3	84	0.00013	1
sp|P39126|IDH_BACSU	ISOCITRATE DEHYDROGENASE (NADP)...	+3	55	0.00045	2
sp|P29102|LEU3_BRANA	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	79	0.00066	1
pdb|4ICD|	Phosphorylated Isocitrate Dehyd...	+3	53	0.00082	2
pdb|3ICD|	Isocitrate Dehydrogenase (E.C.1...	+3	53	0.00082	2
pdb|6ICD|	Isocitrate Dehydrogenase (E.C.1...	+3	53	0.00082	2
pdb|7ICD|	Isocitrate Dehydrogenase (E.C.1...	+3	53	0.00082	2
sp|P31958|LEU3_CLOPA	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	77	0.0012	1
sp|P12010|LEU3_BACCO	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	76	0.0017	1
gp|M33099|BACIPMD_1	B.coagulans 3-isopropylmalate d...	+3	76	0.0017	1
sp|P24404|LEU3_AGRTU	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	76	0.0017	1
pir|A55591|A55591	isocitrate dehydrogenase (NADP+...	+3	53	0.0018	2
sp|P05644|LEU3_BACCA	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	75	0.0023	1
sp|P24098|LEU3_THEAQ	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	74	0.0032	1
pir|JX0286|JX0286	3-isopropylmalate dehydrogenase...	+3	74	0.0032	1
pdb|1IPD|	3-Isopropylmalate Dehydrogenase...	+3	73	0.0044	1
pir|S41223|S41223	3-isopropylmalate dehydrogenase...	+3	73	0.0044	1
gp|U25634|AVU25634_3	ttuC gene product [Agrobacteriu...	+3	52	0.0044	2
sp|P05645|LEU3_BACSU	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	73	0.0044	1
sp|P41560|IDH1_VIBA1	ISOCITRATE DEHYDROGENASE (NADP)...	+3	48	0.0071	2
pir|B49341|B49341	isocitrate dehydrogenase (NADP+...	+3	48	0.0071	2
pir|S43888|S43888	leuB protein - Neisseria lactam...	+3	71	0.0082	1
sp|Q00412|LEU3_SPIPL	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	71	0.0084	1
pir|A44851|A44851	3-isopropylmalate dehydrogenase...	+3	71	0.0084	1
sp|P24015|LEU3_LEPIN	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	71	0.0084	1
sp|P41019|LEU3_BACME	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	71	0.0084	1
gp|U00022|U00022_23	pbpC [Mycobacterium leprae]	-2	32	0.029	3
sp|P28834|IDH1_YEAST	ISOCITRATE DEHYDROGENASE (NAD),...	+3	67	0.030	1
sp|P00351|LEU3_THETH	3-ISOPROPYLMALATE DEHYDROGENASE...	+3	64	0.076	1

Fig. 4. Comparison of the output from **blast** searches of nucleotide vs protein databases. **(A)** Example of the output from a **blast** search using the nucleotide sequence in Figure 1 against a nucleotide database. **(B)** Same sequence used to search a protein database with **blast**.

3.5.2. Retrieve E-mail Server

Complete sequence entries can be retrieved directly from the NCBI by text terms or accession numbers. An example of how to obtain the complete GenBank sequence entry for entry U07681 follows:

mail retrieve@ncbi.nlm.nih.gov
DATALIB genbank
BEGIN
U07681

Send an e-mail message containing the word **help** to **retrieve@ncbi.nlm.nih.gov** to obtain the complete help file.

3.5.3. Entrez

Entrez is a program created at the NCBI that allows query and retrieval of an integrated set of databases consisting of nucleotide, protein, structure, and reference databases. Entries can be retrieved based on a number of criteria, including accession number, text terms, author, taxonomy, and similarity to other entries in the databases. Entries retrieved by entrez are all cross-referenced, such that starting with the accession number of a sequence of interest, one can quickly obtain the corresponding nucleotide and protein sequence entries, as well as those of related entries, literature references, abstracts, and links to structural information when available. The entrez program is available by anonymous ftp from **ncbi.nlm.nih.gov** and is available for most major computing platforms. Use of the network version of this program requires preregistration with the NCBI. A WWW interface to network entrez is accessible to anyone at **http://ncbi.nlm.nih.gov**.

3.6. Multiple Alignment of Homologous Sequences

Once a sequence has been identified as encoding a member of a potential gene family, important information concerning gene structure and function can be extracted from a multiple alignment of related sequences. There are several approaches to multiple-sequence alignment. Only two are presented here: the automated method of GCG's **pileup** and the interactive method of **macaw**.

3.6.1. Pileup

A multiple-sequence alignment is produced by **pileup** using a progressive alignment algorithm in which the sequences are clustered by pairwise comparison. This alignment method works best if the sequences are very similar and because this method requires that all the sequences be aligned; it fails when one or more of the sequences are not true homologs. The alignment produced

```
           1                                                    50
 IDH_BACSU  .......... .......MAQ GEKITVSNGV LNVPNNPIIP FIEGDGTGPD
 IDH_ECOLI  .......... MESKVVVPAQ GKKITLQNGK LNVPENPIIP YIEGDGIGVD
IDH1_YEAST  ....MLNRTI AKRTLATAAQ AE....RTLP KKYGGRFTVT LIPGDGVGKE
IDH2_YEAST  MLRNTFFRNT SRRFLATVKQ PSIGRYTGKP NPSTGKYTVS FIEGDGIGPE
 IDH_THETH  .......... .MPLITTETG ......KKMH VLEDGRKLIT VIPGDGIGPE

           51                                                   100
 IDH_BACSU  IWNAASKVLE AAVEKAYKGE KKITWKEVYA GEKAYNKTGE ..WLPAETLD
 IDH_ECOLI  VTPAMLKVVD AAVEKAYKGE RKISWMEIYT GEKSTQVYGQ DVWLPAETLD
IDH1_YEAST  ITDSVRTIFE AE. .. ..N IPIDW.E..T ININKQ...TD HKEGVYEAVE
IDH2_YEAST  ISKSVKKIFS AA... ...N VPIEW.E..S CDVSPIFVNG LTTIPDPAVQ
 IDH_THETH  CVEATLKVLE AA. .....K APLAY EVRE AGASVFRRGI ASGVPQETIE

           101                                                  150
 IDH_BACSU  VIREYFIAIK GPLTTPVGG. GIRSLNVALR QELDLFVCLR PVRYFTGVPS
 IDH_ECOLI  LIREYRVAIK GPLTTPVGG. GIRSLNVALR QELDLYICLR PVRYYQGTPS
IDH1_YEAST  SLKRNKIGLK GLWHTPADQT GHGSLNVALR KQLDIYANVA LFKSLKGVKT
IDH2_YEAST  SITKNLVALK GPLATPIGK. GHRSLNLTLR KTFGLFANVR PAKSIEGFKT
 IDH_THETH  SIRKTRVVLK GPLETPVG.Y GEKSANVTLR KLFETYANVR PVREFPNVPT

           151                                                  200
 IDH_BACSU  PVKRPEDTDM VIFRENTEDI YAGIEYAKGS EEVQKLISFL QNELNVNKIR
 IDH_ECOLI  PVKHPELTDM VIFRENSEDI YAGIEWKADS ADAEKVIKFL REEMGVKKIR
IDH1_YEAST  RIP..DI.DL IVIRENTEGE FSGLEHESVP GVVE...... ..........
IDH2_YEAST  TYE..NV.DL VLIRENTEGE YSGIEHIVCP GVVQ... .. ..........
 IDH_THETH  PYAGRGI.DL VVVRENVEDL YAGIEHMQTP SVAQ...... ..........

           201                                                  250
 IDH_BACSU  FPETSGIGIK PVSEEGTSRL VRAAIDYAIE HGRKSVTLVH KGNIMKFTEG
 IDH_ECOLI  FPEHCGIGIK PCSEEGTKRL VRAAIEYAIA NDRDSVTLVH KGNIMKFTEG
IDH1_YEAST  .......SLK VMTRPKTERI ARFAFDFAKK YNRKSVTAVH KANIMKLGDG
IDH2_YEAST  .......SIK LITRDASERV IRYAFEYARA IGRPRVIVVH KSTIQRLADG
 IDH_THETH  .......TLK LISWKGSEKI VRFAFELARA EGRKKVHCAT KSNIMKLAEG

           251                                                  300
 IDH_BACSU  AFKNWGYELA EKEYGDKVFT WAQYDRIAEE QGKDAANKAQ SEAEAAGKII
 IDH_ECOLI  AFKDWGYQLA REEFGGELID GGPWLKV... .......... KNPNTGKEIV
IDH1_YEAST  LFRNIITEIG QKEYPD.... .......... .......... ........ID
IDH2_YEAST  LFVNVAKELS .KEYPD...  .......... ..... .... ........LT
 IDH_THETH  ..PKRAFEQV AQEYPD.... .......... .......... ........IE

           301                                                  350
 IDH_BACSU  IKDSIADIFL QQILTRPNEF ..DVVATMNL NGDYISDALA AQVGG.IGIA
 IDH_ECOLI  IKDVIADAFL QQILLRPAEY ..DVIACMNL NGDYISDALA AQVGG.IGIA
IDH1_YEAST  VSSIIVDNAS MQAVAKPHQF ..DVLVTPSM YGTILGNIGA ALIGGP.GLV
IDH2_YEAST  LETELIDNSV LKVVTNPSAY TDAVSVCPNL YGDILSDLNS GLSAGSLGLT
 IDH_THETH  AVHIIVDNAA HQLVKRPEQF ..EVIVTTNM NGDILSDLTS GLIGG.LGFA

           351                                                  400
 IDH_BACSU  PGANINYETG HAIFEAT.HG TAPKYAGLDK VNPSSVILSG VLLLEHLGWN
 IDH_ECOLI  PGANIGDEC. .ALFEAT.HG TAPKYAGQDK VNPGSIILSA EMMLRHMGWT
IDH1_YEAST  AGANFGRDY. .AVFEPGSRH VGLDIKGQNV ANPTAMILSS TLMLNHLGLN
IDH2_YEAST  PSANIGHKI. .SIFEAV.HG SAPDIAGQDK ANPTALLLSS VMMLNHMGLT
 IDH_THETH  PSANIGNEV. .AIFEAV.HG SAPKYAGKNV INPTAVLLSA VMMLRYLEEF
```

Fig. 5. Example of an alignment produced by **pileup**

by **pileup** should be inspected and manually edited, if necessary, before proceeding to use this alignment in any further analysis. Figure 5 demonstrates a portion of an alignment produced by **pileup**.

3.6.2. Macaw

The **macaw** program offers a different approach to multiple sequence *(4)*. This program runs on both Macintosh and Windows operating systems, and allows interactive alignment of multiple sequences. Regions of high similarity can be aligned with high stringency and locked into place. Then intervening regions can be aligned with reduced stringency, until the entire sequence is aligned. Sequences that do not have similarity in multiple segments are quickly identified, allowing the alignment to be properly interpreted.

3.7. Identification of Functional Domains

3.7.1. Motifs

Conserved protein sequence motifs, such as nucleotide binding sites and protein modification sites, which have been identified in the PROSITE database, can be identified using the **motifs** program in the GCG package. The PROSITE database also contains detailed documentation about the proteins containing each motif and a list of known false-positive and false-negative sequences.

3.7.2. ProfileScan

Multiple alignments of related protein sequences have been used to calculate tables in which a score is given for every amino acid at each position based on its frequency of occurrence *(9)*. Sets of these profile tables can be used by **profilescan** to search a protein sequence for regions of high similarity with these protein families.

3.7.3. ProfileSearch

Profiles can also be created from multiple-sequence alignments produced by **pileup** or **macaw**, and used to search the protein sequence databases for more homologs. These searches have the potential to be much more sensitive than **blast** or **fasta** searches, but take much longer to run. Ideally, a profile should be created using only the most conserved regions of the alignment. This profile can then be used by **profilesearch** to search the database, and the results can be aligned with the query profile by the program **profilesegments**. As with any of these searching techniques, this can be an iterative process, with more sequences added to the alignment with each pass.

3.8. Submission of Completed Sequences to Databanks

New sequence data should be submitted to the genetic databanks when it is complete. There is rapid sharing of the data by the International Collaboration of Nucleotide Sequence Databases, and it is therefore only necessary to submit

to one database: GenBank, EMBL, or DDBJ. Deposition of sequence data is preferably done either through WWW servers: **http://www.ncbi.nlm.nih.gov**, **http://www.ebi.ac.uk**, or e-mailing a completed submission form to **gb-sub@ncbi.nlm.nih.gov**, **datasubs@ebi.ac.uk**, or **ddbjsub@ddbj.nig.ac.jp**. Most journals require that sequence data be deposited and an accession number be obtained before publication. It is possible to request that the data not be released until the article is published in the scientific press.

4. Notes

4.1. GCG Introduction

1 Logging on and starting GCG. Users should have an account established by their system administrator on a machine that is running the GCG software The exact instructions for starting GCG vary by system and should be obtained from the administrator.

2 GCG help: There are many more programs and program options than can be presented in this limited format. Details of these can be found in the printed GCG manual or found online by typing **genmanual** Alternatively, entering **genhelp** will bring up a help file in which each program is listed in alphabetical order A list of command line options for each program can also be obtained by entering the program name followed by **-check**.

3. Program prompting: Most GCG programs need, and ask for, at least one input file name. In addition, many programs write their output into a file and ask for an output file name. Usually the program suggests a default answer by presenting it within parentheses and asterisks, for example (* filename.out *) One can accept the default answer by pressing return or type another value then press return Most programs are insensitive to unexpected responses and simply repeat the question if something unacceptable is entered

4. Command line structure: There are three types of parameters that can be specified on the command line when executing a GCG command.
   a. Required parameters: Items such as input sequence name, beginning and ending positions, and output file name must be supplied or else the program will pause to allow input of any missing parameters. Use of **-default** disables this interactive input and forces the use of the default program values for any option that is not specified.
   b. Local data files: There is the option of changing the data file that the program will use by specifying a new file on the command line or by having a file of the same name as the default file in the local directory. These include translation tables, sequence comparison tables, restriction enzyme lists, and so forth.
   c Optional parameters: Most programs have several options that can only be set on the command line. These include such options as the output line width, limits on the lengths of results, and so on. A list of optional parameters for each program can be found by using **genhelp** or by using the **-check** command line qualifier.

Table 1
General File Utility Commands in VMS and UNIX Operating Systems

Function	VMS command	UNIX command
List a file	**type** <filename>	**cat** <filename>
Delete a file	**delete** <filename>	**rm** <filename>
Copy a file	**copy** <filename1> <filename2>	**cp** <filename1> <filename2>
Rename a file	**rename** <filename1> <filename2>	**mv** <filename1> <filename2>

5 Graphic output: GCG programs support an array of graphic output drivers, including postscript, HPGL, tektronics, and XWindows. Users should see Appendix C of the Wisconsin Sequence Analysis Package for details.
6. Customizing GCG data files: GCG data files including restriction enzyme lists, and translation tables can be retrieved and edited. Type **fetch** <filename>, and a local copy of the file will be retrieved to the local directory. The local copies of these files will then be used as default.

4.2. UNIX vs VMS

7 GCG command differences: There are several major differences between the VMS and UNIX operating systems that are of concern to users of the GCG programs The GCG command names are identical in both systems, but commands may be abbreviated and file names are case insensitive in VMS, whereas not in UNIX. A slash, "/" is used to identify a command-line option in VMS, whereas in UNIX, a dash, "–", is used. Commands may be extended over multiple lines by entering a "–" at the end of each line in VMS or a "\" in UNIX
8 File utilities: File names in VMS follow the pattern DISK:[Dir]Filename.txt, whereas in UNIX, the pattern is /dir/filename.txt. *See* Table 1 for other general file utility commands.

4.3. Sequence Contamination

9. Vector sequence contamination Unfortunately, a large number of sequences that are deposited into sequence databases either represent portions of the cloning vector or are contaminated with portions of the cloning vector sequence. A blast search performed at the earliest stages of the sequence will usually indicate whether the sequence is from a potentially interesting clone or from a cloning artifact. Precautions should be taken in the final stages of preparing the sequence to eliminate the deposition of any extraneous sequences into the databases.
10. Degenerate primer sequence contamination: The sequences that are complementary to degenerate primers used to amplify the DNA prior to cloning should be removed from the sequence prior to performing any sequence comparison search. These sequences, which often represent conserved regions and thus have a high likelihood of finding a match in the database, are not necessarily part of the origi-

nal coding sequence. These primer sequences should also be removed from the sequence before deposition into the sequence databases.

4.4. Sequence Searching Over the Internet

11. A number of resources are available for searching the sequence databases over the internet. Gateways to various search engines can be found at **http://www.ncbi.nlm.nih.gov**, **http://www.ebi.ac.uk**, **http://www.blocks.fhcrc.org**, **http://dot.imgen.bcm.tmc.edu:9331**

References

1 Genetics Computer Group (1994) *Program Manual for the Wisconsin Package, Version 8*. Madison, WI.

2 Griffin, H G. and Griffin, A M , eds (1994) *Methods in Molecular Biology, vol 24, Computer Analysis of Sequence Data, parts I and II*, Humana Press, Totowa, NJ.

3 Altschul, S. F , Boguski, M S., Gish, W., and Wootton, J C (1994) Issues in searching molecular sequence databases. *Nat. Genet.* **6,** 119–129

4 Schuler, G. D , Altschul, S. F , and Lipman, D. J (1991) A workbench for multiple alignment construction and analysis *Proteins. Struct Funct , and Genetics* **9,** 180–190.

5. Gribskov, M , Devereux, J., and Burgess, R R. (1984) The codon preference plot. graphic analysis of protein coding sequences and prediction of gene expression *Nucleic Acids Res.* **12,** 539–549

6. Fickett, J. W. (1982) Recognition of protein coding regions in DNA sequences *Nucleic Acids Res.* **10,** 5303–5318.

7 Altschul, S. F., Gish, W., Miller, W., Myers, E W , and Lipman, D. J (1990) Basic local alignment search tool *J. Mol Biol* **213,** 403–410.

8. Pearson, W. R. and Miller, W. (1992) Dynamic programming algorithms for biological sequence comparison. *Methods Enzymol* **210,** 575–601

9. Gribskov, M., Luethy, R., and Eisenberg, D (1989) Profile analysis *Methods Enzymol.* **183,** 146–159.