Volume 10 Number 1 1982 Nucleic Acids Research

SEQ: a nucleotide sequence analysis and recombination system

D.L.Brutlag[1], J.Clayton[2], P.Friedland[3] and L.H.Kedes[4]

Departments of [1]Biochemistry, [3]Computer Science, and [4]Medicine, Stanford University, Stanford, CA 94305, and [2]IntelliGenetics Inc., 124 University Avenue, Palo Alto, CA 94301, USA

Received 15 September 1981

Abstract

SEQ is an interactive, self-documenting computer program that contains procedures for the analysis of nucleotide sequences and the manipulation of such sequences to allow the simulation and prediction of the results of recombinant DNA experiments.

Introduction

The *SEQ* analysis system provides an interactive environment for the analysis of data obtained from nucleotide sequencing and for the simulation of recombinant DNA experiments. The interactive environment and the self-documenting nature of the program make it easy for the non-programmer to use. *SEQ* prompts the user for each piece of information required and the response is most often a single number or a letter. The program is particularly helpful for the inexperienced user in that typing a question mark at any point will provide for the user a description of what the program expects. *SEQ*'s responses to question marks form a short tutorial on the program itself. In addition, all errors made by the user are explained in plain English so that appropriate responses can be made.

While the various procedures of the *SEQ* program are related to many other programs of similar purpose [1]; [2]; [3]; [4]; (and reviewed in [5]), *SEQ* has a rapid and improved homology and dyad symmetry search algorithm which finds many homologies and dyad symmetries that are overlooked by earlier algorithms. *SEQ* also prepares restriction maps with the names and locations of the restriction sites marked on the nucleotide sequence as well as tables containing the length of DNA fragments produced by restriction digests of any known sequence. *SEQ* treats circular sequences properly for all of its search and comparision functions.

SEQ contains 13 primary procedures with over 25 additional suboptions that allow sequence data to be displayed and analyzed in a wide variety of ways. The user can analyze any sequence, any part of a sequence, the inverse-complement of a sequence, or any combination of parts of sequences. In addition, any new sequences generated by the combination of parts of sequences can be saved and can serve as a permanent record for later analysis. It is this capability to combine parts of sequences that allows one to simulate recombinant DNA experiments and predict structures that will be generated in the laboratory.

In addition to many different ways of manipulating sequence information and representing it, the *SEQ* program has many analytical tools that can search for sequence patterns both within and between sequences. These include procedures for finding exact internal repeats, inexact but statistically significant direct repeats, inverted repeats and dyad symmetries within sequences, as well as homologies and potential areas of hybridization between sequences.

The *SEQ* system was written in the SAIL language which currently runs on DEC-10 and DEC-20 computers and is particularly well suited to symbolic manipulation.

Description of the Program

Immediately upon invoking the *SEQ* program the user is requested to provide the name of a file which contains the sequences to be analyzed and the name of a file in which to store the output. The format for sequence storage is particularly simple and flexible. Descriptive information about the nucleotide sequence (such as literature references, locations of gene regions or special sites) can be included in comment lines before the sequence itself. Comment lines begin with a semi-colon and are ignored by all of the procedures except one which specifically prints the comments. The first line without a semi-colon is taken as the name of the sequence and all subsequent lines contain the sequence itself. The sequence can be typed in any format desired. Blanks are ignored and the length of the lines is not important. The sequence is terminated by either the character 1 or 2, where 1 indicates that the sequence is linear and 2 indicates that the sequence is circular or tandemly repeated. Designation of a sequence as circular allows the finding of restriction sites that may be interrupted between the beginning and the end of the sequence, homologies or internal repeats that cross this boundary of a sequence, dinucleotides between the first and last sequence, dyad symmetries that span the discontinuity, etc. Descriptive information about another sequence may follow immediately after the end of the sequence and there may be any arbitrary number of sequences in a file.

Having loaded a specific sequence file (or files), *SEQ* now prompts the user for those analytical procedures he wishes to employ. In addition to the various procedures listed below, the program offers the user the chance to control the execution of the program and the disposition of the output. For instance, at this point one may decide to load additional sequences that were forgotten during the initialization of the program. The user may also request that the program either begin or cease sending the output to a file. He may also modify many internal parameters of *SEQ* which specify such items as the minimum percent of exact match (PERCENTMATCH) required to show an inter- or intra-sequence homology. *SEQ* keeps track of a particular user's personal preference on these parameters by storing a "profile" of that user which is used upon *SEQ* initialization. Finally, by typing Q the user can request that *SEQ* terminate execution altogether. The user may change the nature of the execution of the program after each analysis. This allows errors to be detected and rectified immediately.

The following is a complete list of the various analytical procedures that are available in *SEQ* as of August, 1981:

Options 1 and 2 -- Printing of Sequences and Comments

The first procedure prints the sequence either 50 or 100 characters per line and either one strand or both complementary strands. The sequence is divided by blanks every ten bases and numbered. The numbering gives the relative position from the beginning of the sequence even if only part of the sequence is printed. In addition, option 2 prints the comment lines that appear with the sequence in the sequence file.

An example of double-stranded printing:

```
        10          20          30          40          50
CCACATTTTG CAAATTTTGA TGACCCCCCT CCTTACAAAA AATGCGAAAA
GGTGTAAAAC GTTTAAAACT ACTGGGGGGA GGAATGTTTT TTACGCTTTT

        60          70          80          90         100
TTGATCCAAA AATTAATTTC CCTAAATCCT TCAAAAAGTA ATAGGGATCG
AACTAGGTTT TTAATTAAAG GGATTTAGGA AGTTTTTCAT TATCCCTAGC
```

Option 3 -- Mono-, Di-, and Tri-Nucleotide Tables

This procedure calculates base composition, dinucleotide frequencies, trinucleotide frequencies and codon usage tables. The latter can be used to examine codon usage in different reading frames. By examining only part of a nucleotide sequence one can determine the codon usage within specific exons.

Option 4 -- Enriched Regions

This procedure looks for regions that are particularly enriched for various pairs of nucleotides such as purines or pyrimidines or any other pair of bases one wishes. These regions are marked on the sequence itself.

Option 5 -- Lexicographies

This procedure is adapted from the lexicography procedure described by Korn, Queen and Wegman [1]). It first prepares a set of "words" each of which is of sufficient length to be unique within the sequence and the beginning of which therefore uniquely defines each position in the sequence. These "words" are then alphabetized to allow one to look for exact internal repeats of any specified length. The length of matching before two similar words are printed can be controlled by setting a parameter (MATCHLENGTH).

Option 6 -- General String Searching and Mapping

Option 6 is a general searching procedure that looks for patterns in DNA sequences such as TATA boxes, restriction sites, etc. This procedure can be used to either list the locations of the sites or to display the locations on the sequence itself. The procedure can be used to generate restriction maps of sequences for a limited number of enzymes or subsets of the enzymes (those that leave cohesive ends or those that recognize four base sequences etc.). Another option allows one to determine the

lengths of the DNA fragments that would be generated by cleaving the known sequence with a one or more enzymes and to order those fragments in decreasing size as obtained in a restriction enzyme digest.

Example of a restriction map:

```
      10        20        30        40        50        60        70
CCACATTTTGCAAATTTTGATGACCCCCCTCCTTACAAAAAATGCGAAAATTGATCCAAAAATTAATTTC
          ↑                 ↑                           ↑
          EcoRI*            MnlI                        DpnI

      80        90       100       110       120       130       140
CCTAAATCCTTCAAAAAGTAATAGGGATCGTTAGCACTGGTAATTAGCTGCTCAAAACAGATATTCGTAC
  ↑                         ↑                ↑↑                ↑   ↑
  EcoRI*                    DpnI             AluI                  RsaI
                                             BbvI              HinFIII
                                             Fnu4HI
```

Example of a list of restriction fragments:

Enzyme	Site	Length	Enzyme	Site
AluI	(116)	186	AluI	(302)
MnlI	(28)	88	AluI	(116)
AluI	(302)	85	MnlI	(28)

Notice that the last of these fragments is from a circular sequence and it spans the beginning and end of the sequence as stored in the file.

Option 7 -- Translation

This procedure invokes the genetic code to determine the amino acid sequence of various regions within a nucleotide sequence. The user may choose to use the universal genetic code or that from mammalian mitochondria (Sanger et al., 1981). The translated amino acid sequence can be presented in either the one letter or the three letter amino acid codes and translation can be limited to only specific coding regions (exons) if desired. Methionines are distinguished by capital letters and termination codons by periods:

```
                                  27                                  54
CCA CAT TTT GCA AAT TTT GAT GAC CCC CCT CCT TAC AAA AAA TGC GAA AAT TGA
Pro His Phe Ala Asn Phe Asp Asp Pro Pro Pro Tyr Lys Lys Cys Glu Asn  .
 His Ile Leu Gln Ile Leu MET Thr Pro Leu Leu Thr Lys Asn Ala Lys Ile Asp
  Thr Phe Cys Lys Phe  .   .  Pro Pro Ser Leu Gln Lys MET Arg Lys Leu Ile


                                  81                                 108
TCC AAA AAT TAA TTT CCC TAA ATC CTT CAA AAA GTA ATA GGG ATC GTT AGC ACT
Ser Lys Asn  .  Phe Pro  .  Ile Leu Gln Lys Val Ile Gly Ile Val Ser Thr
 Pro Lys Ile Asn Phe Pro Lys Ser Phe Lys Lys  .   .  Gly Ser Leu Ala Leu
  Gln Lys Leu Ile Ser Leu Asn Pro Ser Lys Ser Asn Arg Asp Arg  .  His Trp
```

Options 8, 9 and 10 -- Intra-Sequence Comparisons

These procedures search for homologies, symmetries, and dyad symmetric regions within a nucleotide sequence allowing for insertion and deletion loops as well as mismatches in the

alignments. The algorithm used is a revision and extension of the very rapid search method described by Korn, Queen and Wegman [1]. Briefly, a homology between two regions of a sequence is extended if the next base in each region match or if, after a limited number of mismatches and/or insertions and deletions, some locally well matched regions can be found. For instance if a mismatch is followed by two out of the next three bases matched in the same register, then the homologous regions are extended to include the mismatch and the matched nucleotides. If there are not two out of three matched nucleotides in the same register, then various sizes of insertion/deletion loops of between one and three nucleotides are chosen to see if there is some other register in which two out of three match. We have chosen to order the search for insertion/deletion and mismatches to obtain further homology such that extension of the homology will usually result in the minimum amount of mismatch while still allowing the algorithm to penetrate through regions that are particularly poorly matched to find a new region of homology. Since all possible insertion/deletion loops and consecutive mismatches up to 3 in a row are tried, homologies and dyad symmetries will be found by this algorithm in which a base on one strand is mismatched with two on another or two versus three etc. This allows the extended dyad symmetry procedure to find RNA stems which have unequal bulge loops and many other structures that are missed by other methods.

An example of a homology between two regions:

```
          *  *    *       **  **    **    *     *
  59    AAAAT TAATTTCCCTAA ATCCTTCA AAAAGTAAT AG      94
 102    AAAATCTGATT CCCTAATTTCGGTCATTAAA TAATCAG     139
```

An example of a dyad symmetry in which G-T pairing was allowed:

```
  36    CAAAAAATGCGAAAATTGATCCAAAAA TT AATTTCCCTAAATCCTTCAA AAAGT      89
        |||||  |||:|||  ||  |||||||  || |||||||   | |  ||  || | |||
  19    GTTTTAAACGTTTTACAC CGGTTTTTTAAATTAAAGG   TAA AAACTTGTGTCA    327
```

Note in both of the above alignments there are single bases mismatched with two bases in addition to both single and double mismatches alone.

The actual algorithm used to discover homologies or dyad symmetries can be modified greatly by the settings of various parameters that specify whether mismatches are allowed or whether insertion/deletion loops are allowed and how well paired the entire structure must be. In addition to these criteria, one may also specify the minimum overall matching as well as criteria for statistical significance of the homologies and symmetries. The statistical significance of each homology is calculated by the methods originally suggested by Korn et. al. [1]. The homology search procedure described above is represented as a Markov process in which the probability of any homology can be readily determined as the stationary state of the Markov process [6], [7]. The *SEQ* program actually calculates these absolute probabilities for homologies up to those 50 nucleotides in length and from 50 to 100% matched for each application of the program. This calculation takes into account the base composition of the sequence and the settings of the various parameters that control the algorithm. Hence if one repeats a homology search and asks for the ten most significant internal homologies with a wide variety of different algorithms (allowing mismatches, allowing

insertions/deletions, or both) or using DNAs of widely varying length or base composition, one obtains between 5 and 20 homologies in most cases. After each homology or dyad symmetry the absolute probability of that homology or one more significant is printed. This absolute probability is multiplied by the total number of alignments considered during the search to obtained the expected number of homologies or dyad symmetries of this degree of matching. Since these expectation values are calculated for each search, they are correct not only relatively, but they can be used as absolute measures of significance. A homology or dyad symmetry with an expectation frequency of 0.05 or less would be considered statistically significant. For dyad symmetries the *SEQ* program also calculates the free energy of formation of the dyad for single-stranded RNA in solution according to the methods of Tinoco [8], [9], [10].

Options 11, 12 and 13 -- Inter-Sequence Comparisons

These procedures search for homologies, symmetries and hybridizations between two different sequences and are in every way analogous to procedures 8, 9 and 10 which search for these patterns within single sequences.

Sequence Specification

After the user has specified which analytical procedures he wishes to perform by typing one or more of the option numbers 1 through 13, he is then prompted for the sequences he wishes to analyze. A question mark at this point will provide a list of the names of all the sequences that were loaded into the *SEQ* program as well as the first comment line and the length of the sequences as determined by *SEQ*. At this point the user has the choice of specifying one or more of those sequences either by name or by number. He may also analyze only part of a sequence by following the name (or number) of the sequence with a starting and an ending nucleotide position on the sequence. The inverse-complement of the sequence that was originally loaded can also be analyzed by typing the sequence name followed by an apostrophe "'". These features can be combined as well:

Sequence: SV40' 1 300

This indicates that one wants to examine the first three hundred nucleotides of the inverse-complement of SV40. Since the SV40 DNA sequence is stored beginning and ending at the origin of replication, this would correspond to the first 300 bases of the early region.

One may also respond to the Sequence: prompt with the word COMBINED. The program will then ask for a series of DNA segments to be taken from the sequences that have been loaded into the program. The user may join as many segments from as many different sequences as he wishes to form a recombinant DNA or take all the segments from a single sequence as he might if he were generating a spliced RNA sequence from the complete transcript. As soon as he has specified all of the segments for a combined sequence, *SEQ* asks for the name for the combined sequence and for comments to store with the sequence. This combined sequence can then be analyzed in further

iterations through *SEQ* for its restriction map, translation, codon usage and other properties that might be unique to the combined sequence. This sequence can also be written by *SEQ* into a sequence file for analysis at a later time. COMBINED is an exceptionally powerful and particularly easy way for manipulating and storing nucleotide sequence information.

Suboptions

After specifying the sequences which the user wishes to analyze, the program will then ask him about suboptions to any of the procedures listed above. For instance for sequence printing option 1 he will be asked if he wishes single- or double-stranded printing. For each of the other procedures there are a multiplicity of different suboptions that will control the exact nature of the output and the analysis. One may respond to the request for more information for an option with a carriage return, in which case standard default answers are used. The default answers are those answers most often given to any particular question.

Iteration

After performing the first analysis that was requested, *SEQ* automatically returns to ask the user if he wishes to apply any other procedures. Typing a Q at this point halts the program. Any option may be used repeatedly or new procedures can be tried. This allows one to cycle repeatedly through *SEQ* changing the parameters slightly to see the effect on the analysis. This interactive analysis can also be recorded into a file for reading at some later time.

A Short *SEQ* Example

This short example of *SEQ* is designed to give the flavor of a typical interaction with the program. In the example, the intervening sequences of a a mouse β-globin gene are deleted from the sequence of the genomic clone and the sequence of the resulting "spliced" exons is compared with the sequences of a cloned mouse β-globin cDNA. In the typescript that follows, user responses are printed in small capital letters. We have added some explanatory comments in italics. (Note: <CR> denotes carriage return.)

```
@ SEQ<CR>

                    SEQ - Sequence Analysis System
                          August 11, 1981

          (Copyright 1981 by the Board of Trustees, Stanford University)

Sequence file (name.ext or <CR> when done) <SEQUENCES>MOUSE.LASL<CR>

Sequence file (<CR> when done) <SEQUENCES>GLOBIN.SEQ<CR>

Sequence file (<CR> when done) <CR>
```

The program asks for the names of files where the sequences to be analyzed are stored. In this case they are on "public" directories that all users of the computer system are free to access.

```
File for output? (<CR> for none) <CR>
```

The user has the option of storing in a permanent file all the data generated during the session. Data storage can be turned on and off at any time during the program as well at the beginning.

```
    Enter the option numbers you want, one per line. (<CR> when done)
    Type "Q" to quit.

Option #: ?<CR>

    1  Prints the sequence (single or double stranded)
    2  Prints comments
    3  Nucleotide frequency tables (mono-, di- and/or trinucleotides)
    4  Rich Regions (AG and CT, AT and GC, or AC and GT)
    5  Oligonucleotide dictionaries (lexicography)
    6  Restriction site search, fragment list, and map (general search)
    7  Translation to amino acids (3 frame, 2 strands, one letter AA code)
    8  Homologous regions
    9  Symmetric regions
   10  Regions of dyad symmetry
   11  Intersequence homologies
   12  Intersequence symmetries
   13  Intersequence base-pairing (hybridization)

    P  To change or examine PARAMETERS
    S  To input additional SEQUENCES from a file
    W  To WRITE sequences to a file
    F  To either start or stop output to a FILE
    T  To either start or stop output to the TERMINAL
    Q  To QUIT (to exit the program)
```

User wants to check the comments for special features of each of the the sequences.

```
    Enter the option numbers you want, one per line.  (<CR> when done)

Option #: 2<CR>
Option #: <CR>

    Enter the sequences to be analyzed for Options 1-10. (<CR> when done)

Sequence: 7<CR>
Sequence: 20<CR>
Sequence: <CR>

MOUSHBB

; DEFINITION  MOUSE BETA-GLOBIN MAJOR GENE WITH 2 IVS. 1567BP
; SPECIES     BALB/C
; REFERENCE   (FOR ENTIRE SEQUENCE)
;             KONKEL,D.A., TILGHMAN,S.M., AND LEDER,P.
;             CELL  15, 1125,1132 (1978)
; REFERENCE   (FOR ENTIRE SEQUENCE)
;             VAN OOYEN,A., VAN DEN BERG,J., MANTEI,N., AND WEISSMANN,C.
;             SCIENCE 206, 337 (1979)
; COMMENT     CHECKED AGAINST SUMEX SEQUENCE.
; ORIGIN      5' END OF HAEIII RECOGNITION SITE
```

```
; SITES      KEY        SPAN  DESCRIPTION
;         40 CONFLICT      1   GG (OR GGG?)
;         79 CAP           1  MRNA CAP SITE
;        131 LDR/CDS       0
;        224 CDS/IVS 1     0
;        340 1 IVS/CDS     0
;        562 CDS/IVS 2     0
;        699 CONFLICT      1   G (OR GG?)
;        770 CONFLICT      5   5T (OR 4T?)
;       1007 CONFLICT      2   CT (OR CTCT?)
;       1038 CONFLICT      1   C (OR CC?)
;       1096 CONFLICT      4   GTGG (OR ATAA?)
;       1101 CONFLICT      2   GG (OR AA?)
;       1108 CONFLICT      2   CC (OR C?)
;       1111 CONFLICT      1   G (OR GTAG?)
;       1208 2 IVS/CDS     0
;       1337 CDS/TRL       0
;       1467 POLY A        1  POLY A ADDITION SITE
; Composition:   360 A,  357 C,  356 G,  494 T,  0 N
; Total:   1567 nucleotides


MOBGLOB

; 03.05.79 MOUSE B-MAJ GLOBIN cDNA, KONKEL...CELL,15,1125-1132,1978
; Composition:   145 A,   158 C,   160 G,   163 T,   0 N
; Total:   626  nucleotides
```

The first example above is represenative of the elegantly and carefully annotated sequences being collected by Walter Goad and colleagues at Los Alamos Scientific Laboratories. The information is enough to provide a complete description of locations of intron/exon boundaries in the genomic clone. The second sequence is less well annotated and does not contain information about the start of translation in the cDNA clone.

```
    Enter the option numbers you want, one per line. (<CR> when done)
    Type "Q" to quit.

Option #: 7<CR>
Option #: <CR>

    Enter the sequences to be analyzed for Options 1-10. (<CR> when done)

Sequence: ?<CR>

    At this point you can type a sequence name, a sequence number,
    the word ALL for all the sequences individually, the word COMBINED
    to combine any set of sequences, or follow any of the above with "'"
    for the inverse complement.  You can also follow a sequence name or
    number by a lower and (optionally) upper limit.
```

A list of sequences available would normally follow at this point but are deleted from the typescript for simplicity.

```
Sequence: combined<CR>
```

```
    Specify the sequences you wish to combine (ALL, sequence name or
    sequence number; <CR> when done)

Sequence to combine: 7 131-223<CR>
Sequence to combine: 7 340-561<CR>
Sequence to combine: 7 1208-1336<CR>
Sequence to combine: <CR>

Should this COMBINED sequence be circular? (Y or N) n<CR>

    Please specify a name for this sequence combination for future
    reference.  (Please do not use "ALL" or "COMBINED")

Combination name: spliced globin codons$<CR>
```

The combined sequence, named "spliced globin codons" by the user, has now been appended to the list of all sequences currently loaded in the program and can thus be referred to at a later point in the program for additional analytical procedures.

```
Sequence: 20<CR>
Sequence: <CR>

Translation option? (M, 1, 2, 3, P, F, W, <CR>, or ?) ?<CR>

        Appropriate responses are :

            M - translation using mitochondrial code
            1 - one letter amino acid code
            2 - for double stranded translation (inverse complement is
                created and translated)
            3 - for translation in all three frames
            P - to translate only specified sections of the sequence
            F - Full translation (for options 2 and 3)
            W - Write a translation file in PEP format (not to be used with
                sub-options 2, 3, P, or F);
            <CR> when done
```

PEP *is the name of a MOLGEN program that analyzes peptide sequences similar to the way in which* SEQ *analyzes nucleic acid sequences.*

```
Translation option? (M, 1, 2, 3, P, F, <CR>, or ?) 3<CR>

Translation option? (M, 1, 2, 3, P, F, <CR>, or ?) <CR>


GLOBIN GENE


 The translated sequence is:


                                  27                                  54
ATG GTG CAC CTG ACT GAT GCT GAG AAG GCT GCT GTC TCT TGC CTG TGG GGA AAG
MET Val His Leu Thr Asp Ala Glu Lys Ala Ala Val Ser Cys Leu Trp Gly Lys
 Trp Cys Thr  .  Leu MET Leu Arg Arg Leu Leu Ser Leu Ala Cys Gly Glu Arg
  Gly Ala Pro Asp  .  Cys  .  Glu Gly Cys Cys Leu Leu Pro Val Gly Lys Gly
```

```
                                    81                                  108
GTG AAC TCC GAT GAA GTT GGT GGT GAG GCC CTG GGC AGG CTG CTG GTT GTC TAC
Val Asn Ser Asp Glu Val Gly Gly Glu Ala Leu Gly Arg Leu Leu Val Val Tyr
  .  Thr Pro MET Lys Leu Val Val Arg Pro Trp Ala Gly Cys Trp Leu Ser Thr
  Glu Leu Arg  .  Ser Trp Trp  .  Gly Pro Gly Gln Ala Ala Gly Cys Leu Pro


 . . .


                                   405                                  432
GCA CAG GCT GCC TTC CAG AAG GTG GTG GCT GGA GTG GCC ACT GCC TTG GCT CAC
Ala Gln Ala Ala Phe Gln Lys Val Val Ala Gly Val Ala Thr Ala Leu Ala His
 His Arg Leu Pro Ser Arg Arg Trp Trp Leu Glu Trp Pro Leu Pro Trp Leu Thr
  Thr Gly Cys Leu Pro Glu Gly Gly Gly Trp Ser Gly His Cys Leu Gly Ser Gln


                                   459
AAG TAC CAC TAA
Lys Tyr His  .
 Ser Thr Thr
  Val Pro Leu


MOBGLOB


 The translated sequence is:


                                    27                                   54
ACA TTT GCT TCT GAC ATA GTT GTG TTG ACT CAC AAC CCC AGA AAC AGA CAT CAT
Thr Phe Ala Ser Asp Ile Val Val Leu Thr His Asn Pro Arg Asn Arg His His
 His Leu Leu Leu Thr  .  Leu Cys  .  Leu Thr Thr Pro Glu Thr Asp Ile MET
  Ile Cys Phe  .  His Ser Cys Val Asp Ser Gln Pro Gln Lys Gln Thr Ser Trp


                                    81                                  108
GGT GCA CCT GAC TGA TGC TGA GAA GGC TGC TGT CTC TTG CCT GTG GGG AAA GGT
Gly Ala Pro Asp  .  Cys  .  Glu Gly Cys Cys Leu Leu Pro Val Gly Lys Gly
 Val His Leu Thr Asp Ala Glu Lys Ala Ala Val Ser Cys Leu Trp Gly Lys Val
  Cys Thr  .  Leu MET Leu Arg Arg Leu Leu Ser Leu Ala Cys Gly Glu Arg  .


                                   135                                  162
GAA CTC CGA TGA AGT TGG TGG TGA GGC CCT GGG CAG GCT GCT GGT TGT CTA CCC
Glu Leu Arg  .  Ser Trp Trp  .  Gly Pro Gly Gln Ala Ala Gly Cys Leu Pro
 Asn Ser Asp Glu Val Gly Gly Glu Ala Leu Gly Arg Leu Leu Val Val Tyr Pro
  Thr Pro MET Lys Leu Val Val Arg Pro Trp Ala Gly Cys Trp Leu Ser Thr Leu


                                   189                                  216
TTG GAC CCA GCG GTA CTT TGA TAG CTT TGG AGA CCT ATC CTC TGC CTC TGC TAT
Leu Asp Pro Ala Val Leu  .  .  Leu Trp Arg Pro Ile Leu Cys Leu Cys Tyr
```

```
Trp Thr Gln Arg Tyr Phe Asp Ser Phe Gly Asp Leu Ser Ser Ala Ser Ala Ile
 Gly Pro Ser Gly Thr Leu Ile Ala Leu Glu Thr Tyr Pro Leu Pro Leu Leu Ser


                                    243                                 270
CAT GGG TAA TGC CAA AGT GAA GGC CCA TGG CAA GAA GGT GAT AAC TGC CTT TAA
His Gly  .  Cys Gln Ser Glu Gly Pro Trp Gln Glu Gly Asp Asn Cys Leu  .
 MET Gly Asn Ala Lys Val Lys Ala His Gly Lys Lys Val Ile Thr Ala Phe Asn
  Trp Val MET Pro Lys  .  Arg Pro MET Ala Arg Arg  .  .  Leu Pro Leu Thr


                                    297                                 324
CGA TGG CCT GAA TCA CTT GGA CAG CCT CAA GGG CAC CTT TGC CAG CCT CAG TGA
Arg Trp Pro Glu Ser Leu Gly Gln Pro Gln Gly His Leu Cys Gln Pro Gln  .
 Asp Gly Leu Asn His Leu Asp Ser Leu Lys Gly Thr Phe Ala Ser Leu Ser Glu
  MET Ala  .  Ile Thr Trp Thr Ala Ser Arg Ala Pro Leu Pro Ala Ser Val Ser


                                    351                                 378
GCT CCA CTG TGA CAA GCT GCA TGT GGA TCC TGA GAA CTT CAG GCT CCT GGG CAA
Ala Pro Leu  .  Gln Ala Ala Cys Gly Ser  .  Glu Leu Gln Ala Pro Gly Gln
 Leu His Cys Asp Lys Leu His Val Asp Pro Glu Asn Phe Arg Leu Leu Gly Asn
  Ser Thr Val Thr Ser Cys MET Trp Ile Leu Arg Thr Ser Gly Ser Trp Ala Ile


                                    405                                 432
TAT GAT CGT GAT TGT GCT GGG CCA CCA CCT TGG CAA GGA TTT CAC CCC CGC TGC
Tyr Asp Arg Asp Cys Ala Gly Pro Pro Pro Trp Gln Gly Phe His Pro Arg Cys
 MET Ile Val Ile Val Leu Gly His His Leu Gly Lys Asp Phe Thr Pro Ala Ala
  .  Ser  .  Leu Cys Trp Ala Thr Thr Leu Ala Arg Ile Ser Pro Pro Leu His


                                    459                                 486
ACA GGC TGC CTT CCA GAA GGT GGT GGC TGG AGT GGC CAC TGC CTT GGC TCA CAA
Thr Gly Cys Leu Pro Glu Gly Gly Gly Trp Ser Gly His Cys Leu Gly Ser Gln
 Gln Ala Ala Phe Gln Lys Val Val Ala Gly Val Ala Thr Ala Leu Ala His Lys
  Arg Leu Pro Ser Arg Arg Trp Trp Leu Glu Trp Pro Leu Pro Trp Leu Thr Ser


                                    513                                 540
GTA CCA CTA AAC CCC CTT TCC TGC TCT TGC CTG TGA ACA ATG GTT AAT TGT TCC
Val Pro Leu Asn Pro Leu Ser Cys Ser Cys Leu  .  Thr MET Val Asn Cys Ser
 Tyr His  .  Thr Pro Phe Pro Ala Leu Ala Cys Glu Gln Trp Leu Ile Val Pro
  Thr Thr Lys Pro Pro Phe Leu Leu Leu Pro Val Asn Asn Gly  .  Leu Phe Pro


                                    567                                 594
CAA GAG AGC ATC TGT CAG TTG TTG GCA AAA TGA TAG ACA TTT GAA AAT CTG TCT
Gln Glu Ser Ile Cys Gln Leu Leu Ala Lys  .  .  Thr Phe Glu Asn Leu Ser
 Lys Arg Ala Ser Val Ser Cys Trp Gln Asn Asp Arg His Leu Lys Ile Cys Leu
  Arg Glu His Leu Ser Val Val Gly Lys MET Ile Asp Ile  .  Lys Ser Val Phe
```

```
                                    621
TCT GAC AAA TAA AAA GCA TTT ATG TTC ACT GC
Ser Asp Lys  .  Lys Ala Phe MET Phe Thr
 Leu Thr Asn Lys Lys His Leu Cys Ser Leu
   .  Gln Ile Lys Ser Ile Tyr Val His Cys
```

The spliced globin gene that was constructed translates without stop codons in frame 1 as expected. Inspection of the translation of the cDNA sequence, MOBGLOB, suggests that frame 2 is the correct reading frame from from base 53 to base 496. The user now writes the translated sequences out to a file to be analyzed by the polypeptide analysis program PEP.

```
    Enter the option numbers you want, one per line. (<CR> when done)
    Type "Q" to quit.

Option #: 7<CR>
Option #: <CR>

    Enter the sequences to be analyzed for Options 1-10. (<CR> when done)

Sequence: 20 53-496<CR>
Sequence: 24<CR>
Sequence: <CR>

Translation option? (M, 1, 2, 3, P, F, W, <CR>, or ?) W<CR>

Translation option? (M, 1, 2, 3, P, F, <CR>, or ?) <CR>


MOBGLOB
Limits:      53   496


    Compiling Translation information for MOBGLOB

Write sequence to file []: MOUSEGLOBIN.PEP<CR>

Name for the sequence: MGLOBIN-MRNA<CR>

Include comments from the original sequence? (Y or N) Y<CR>

Additional comments: (carriage return when done)
; <CR>


    Compiling Translation information for GLOBIN GENE

Write sequence to file [MOUSEGLOBIN.PEP]: <CR>

Name for the sequence: MGLOBIN-DNA<CR>

Include comments from the original sequence? (Y or N) Y<CR>

Additional comments: (carriage return when done)
; CONSTRUCTED FROM CODING REGIONS<CR>
; <CR>
```

```
    Enter the option numbers you want, one per line. (<CR> when done)
    Type "Q" to quit.

Option #: 0<CR>

  Wednesday, September 2, 1981 3:14PM-PDT
```

PEP can now be used to look for homologies between the two translated sequences.

```
@PEP

                    PEP - Polypeptide Analysis System
                              August 3, 1981

        (Copyright 1981 by the Board of Trustees, Stanford University)

Polypeptide file (CR when done) MOUSEGLOBIN.PEP<CR>

Polypeptide file (CR when done) <CR>

File for output? (CR for none) <CR>

    Enter the option numbers you want, one per line.  End with an extra <CR>
Type "Q" to quit.

Option #: c[?]


    1  Prints the sequence
    2  Prints comments
    3  Amino Acid composition
    5  String search, proteolysis fragment list, and map (general search)
    6  Hydrophobic/Hydrophilic/Aromatic
    7  Reverse Translation
    8  Write Reverse Translation to a File
    9  Homologous regions
    10 Interpeptide homologies

    P  To examine or change PARAMETERS
    S  To input additional PEPTIDES from a file
    F  To either start or stop output to a FILE
    T  To either start or stop output to the TERMINAL
    Q  To QUIT (to exit the program)

    Enter the option numbers you want, one per line.  End with an extra <CR>

Option #: 10<CR>
Option #: <CR>

    Give two sequence identifiers for Option 10 (<CR> when done)
Sequence 1: ?<CR>

    At this point you can type a peptide name, a peptide number, or
    the word ALL for all the peptides individually.  You can also follow
    a peptide name or number by a lower and (optionally) upper limit.

    The peptides are:
```

```
1 MGLOBIN-RNA ; 03.05.79 MOUSE B-MAJ GLOBIN MRNA,
KONKEL...CELL,15,1125-1132,1978

2  MGLOBIN-GENE          ; CONSTRUCTED FROM CODING REGIONS


Sequence 1: 1<CR>

Sequence 2: 2<CR>

Sequence 1: <CR>


MGLOBIN-MRNA -  MGLOBIN-GENE


 The regions of homology in the two sequences are:


     1     MVHLTDAEKAAVSCLWGKVNSDEVGGEALGRLLVVYPWTQRYFDSFGDLSSASAIMGNAK
     1     MVHLTDAEKAAVSCLWGKVNSDEVGGEALGRLLVVYPWTQRYFDSFGDLSSASAIMGNAK


           VKAHGKKVITAFNDGLNHLDSLKGTFASLSELHCDKLHVDPENFRLLGNMIVIVLGHHLG
           VKAHGKKVITAFNDGLNHLDSLKGTFASLSELHCDKLHVDPENFRLLGNMIVIVLGHHLG


           KDFTPAAQAAFQKVVAGVATALAHKYH.     148
           KDFTPAAQAAFQKVVAGVATALAHKYH.     148

 % =  100.
 P( 148, 148) =  .000          E =  .000


 The number of matches is 1

Expect =  100.000   MinMatch = 2        PercentMatch = 70

AfterDis = 2        LoopOut = 3         MaxDist = 148

    Enter the option numbers you want, one per line.  End with an extra <CR>
Type "Q" to quit.

Option #: Q<CR>
```

SEQ and the *SUMEX* Resource

The *SEQ* program has been available, free of charge, for trial evaluation and guest usage by academic colleagues on the SUMEX-AIM computer facility during the last year (Feb 1980 until the present). Those who wish to avail themselves of this opportunity may write to the authors for guest privileges. The program has been available for the commercial molecular biology community through IntelliGenetics Inc.

Acknowledgments

This work is part of the MOLGEN project, a joint research effort among the Departments of Computer Science, Medicine, and Biochemistry at Stanford University. The research has been supported under NSF grant MCS80-16247. Computational resources have been provided by the SUMEX-AIM Natinal Biomedical Research Resource, NIH grant RR-00785-08, and by the Department of Computer Science.

MOLGEN, *SEQ* and *PEP* are trademarks of the Board of Trustess of Stanford University.

Address all correspondence to: Dr. L.H.Kedes, 151M, VA Hospital, Miranda Avenue, Palo Alto, CA 94304, USA

References

1. Korn, L. J., Queen, C. L. and Wegman, M. N. , "Computer analysis of nucleic acid regulatory sequences," Proc. Nat. Acad. Sci. USA, Vol. 74, 1977, pp. 4401-4405.
2. Staden, R., "Sequence data handling by computer," Nuc. Acids. Res. , Vol. 4, 1977, pp. 4037-4051.
3. Staden, R., "Further procedures for sequence analysis by computer," Nuc. Acids. Res. , Vol. 5, 1978, pp. 1013-1015.
4. Sege,R., Soll, D., Ruddle, F. H. and Queen, C. , "A conversational system for the computer analysis of nucleic acid sequences," Nuc. Acids. Res., Vol. 9, 1981, pp. 437- 444.
5. Gingeras, T. R. and Roberts, R. J. , "Steps toward computer analysis of nucleotide sequences," Science, Vol. 209, 1980, pp. 1322-1328.
6. Feller, W., An Introduction to Probability Theory and its Applications, John Wiley and Sons, Inc., New York, 1968, pp.372-424
7. Karlin, S. and Taylor, H. M., A First Course in Stochastic Processes, Academic Press, New York, 2nd Edition, 1975.
8. Tinoco, I., Uhlenbeck, O.D., and Levine, M.D., , "Estimation of Secondary Structure in Ribonucleic Acids," Nature, Vol. 230, 1971, pp. 5293.
9. Tinoco, I., "Improved Estimation of Secondary Structure in Ribonucleic Acids," Nature New Biol. , Vol. 246, 1973, pp. 40.
10. Borer, P. N., Dengler, B. and Tinoco, I. Jr., "Stability of ribonucleic double-stranded helices," J. Mol. Biol., Vol. 86, 1974, pp. 843-853.