An Introductory Course in Computational Molecular Biology: Rationale, History, Observations, and Course Description

[†§]Susan J. Johns, [†§]Steven M. Thompson, and [§]A. Keith Dunker

[†]Center for Visualization, Analysis, and Design in the Molecular Sciences,
[§]Department of Biochemistry and Biophysics, Washington State University,

Pullman, WA, USA 99164-4660

{prcadams, thompson}@ribozyme.vadms.wsu.edu, dunker@mail.wsu.edu

Abstract. A course called "Molecular Biology Computer Techniques" was implemented in 1987 and has been evolving ever since. Currently the semester-long three credit course consists of thirty hours of lecture (three hours/week for the first ten weeks of the semester) and a minimum of 45 hours of laboratory instruction (three hours/week). The lectures survey both bioinformatics and structure based methods. The laboratory has two tracks, one that can be described loosely as "sequence analysis" and the other as "molecular modelling." Most students choose one of the two laboratory tracks, although a small number have done both, either simultaneously or in successive years. For each student, the goal of the course is the completion of a student-initiated research project. The culmination of the course is the presentation of the completed projects at a "Poster Session Final." During this final, which is conducted like a poster session at a typical biological science meeting, students are examined, not only by the instructors in the course, but also by a diverse cross-section of the university community at large, including non-scientists (who are specially invited to attend). Questioning by non-scientists provides opportunity for the students to improve their communication skills with the lay public. In this manuscript we discuss our views regarding the rationale for the development of formal courses in computational molecular biology, relate our experiences in the development of our course, and describe the course as it stood the last time it was taught, which was in the Fall of 1994.

Introduction. Computational molecular biology can be divided into two broad areas: 1. methods based on the analysis of sequences of amino acid residues or nucleotides; and 2. methods based on the analysis of molecular structure. The former will be referred to as "sequence analysis," or SA, and the latter as "molecular modelling," or MM.

Continually improving hardware and software, and the development of massive, readily accessible databases has led to an explosion in the use of computational methods by molecular biologists. Computer searches of the Cambridge Life Sciences Collection demonstrate exponential growth for both areas, with SA usage about four times more prevalent than MM.

Not only has there been an exponential increase in computer usage in the biological sciences, there have been qualitative changes as well. In earlier times, computers were typically used near the ends of research projects, primarily as number crunchers. Now, especially because of evolutionary relationships revealed by database searches, computer methods are used at earlier phases of a research project, often providing information that guides the subsequent experimental design. The logical continuation of this trend suggests that computer algorithms may eventually become partners with the scientists in the formation of hypotheses.

There is a natural lag from the time of the inception of a new field of research until the development of formal courses and curricula. However, the rapid commercialization of molecular biology software and the easy dissemination of research software, coupled with the weak backgrounds of many users due to the absence of formal courses, has led to a particularly unfavorable situation. Although computer usage is rapidly increasing, these computer tools are often applied to problems in an inappropriate way or the results are wrongly interpreted, at least in part we believe, from the lack of formal course-work to provide the appropriate background. This problem is exacerbated by a tendency, which we don't fully understand, for biologists (and perhaps others) to blindly accept computer results, whereas the same people would not accept the results of wet-laboratory experiments so uncritically. Thus, it is imperative that increased emphasis be placed on the development of formal courses in computational molecular biology.

Rationale for course and curriculum design. The community of scientists involved in computational molecular biology can be roughly divided into the tool-builders and the tool-users. Completely different educational strategies apply to the two groups.

The tool-builders need to be competent in some domain of molecular biology and also in computer science. In our opinion, future progress in computational molecular biology would be substantially enhanced by the development of truly interdisciplinary graduate training programs covering both molecular biology and computer science. These would be analogous to the inter-disciplinary programs in biophysics that were developed in the 1950's and 1960's. The special relationship that we believe exists between computer science and molecular biology makes the future development of such interdisciplinary programs appear to be pedigologically sound.

While educating new scientists with strong backgrounds in both molecular biology and computer science is important for the improvement of the next generation of tool-builders, appropriate implementation of current and evolving algorithms by current tool-users is no less important.

The many examples of significant insights into molecular biology arising from unexpected sequence matches described in *Of Urfs and Orfs* (Doolittle, 1987) made it clear that ready access to the rapidly growing databases would fundamentally change the way in which molecular biology research would be carried out. These observations led us to develop a course in computational molecular biology, first taught in 1987.

An alternative would have been to introduce the basic concepts of computational molecular biology in workshops, which we do periodically offer. However, a formal course seemed to have the advantage over intensive workshops because the longer duration reinforcement of a course setting seems to provide a much better learning experience than short-duration workshops; besides, students earn credits in courses but not (typically) in workshops.

To make this course accessible to as many molecular biologists as possible, a major objective, which we have continually met, is that the course should require zero experience with computers. In order to attain this objective we devote the first two laboratory periods to a very basic introduction to the computing platform used in the course.

A second objective was to make the course as useful as possible to the students. To reach this objective we make student-initiated projects account for most of the grade in the course.

In terms of course content, a debate continuing to the present has been whether to teach two courses, one in sequence analysis and one in molecular modelling, or to combine the two aspects into one course. Our initial decision, which we still follow, was to introduce both sequence analysis and molecular modelling techniques in one course. However, differentiation between sequence analysis and molecular modelling does occur in the laboratory, where each student chooses the track more appropriate for his or her interest, while still being exposed to both areas in lecture.

Three considerations contributed to our decision to teach a single course covering both areas. First, it was felt that typical graduate students in the biological sciences would be unlikely to take two courses devoted to computers (thesis advisors always want to minimize class time and maximize bench time!), yet we felt that students needed an exposure to both areas. Second, one of the central unsolved questions in molecular biology is how an amino acid sequence specifies protein structure. Discussions regarding this topic would clearly involve elements of both sequence analysis and molecular modelling. Finally, all biological sequences exist, not as one-dimensional strings of symbols, but as three-dimensional, physical objects. Thus, it was argued (in 1987) that a major trend in sequence analysis would be the development of methods that utilized information from the 3D structures of molecules having sequences related to the ones under investigation. Indeed, the methods of homology modelling and inverse folding by 3D profiles or threading illustrate this point. Discussion of these topics requires both sequence analysis and molecular modelling awareness.

Molecular Biology Computer Techniques (MBCT): history and experiences. Given the above reasoning, we developed MBCT to cover both sequence analysis and molecular modelling, as a special topics course (i.e. a one-time experimental basis) in the Spring Semester of 1987. Involvement of faculty and staff from Washington State University (WSU) and scientists from the Pacific Northwest Laboratory (PNL - a DOE-funded National Laboratory) for lecturing was made possible, despite the separation of more than 120 miles, by means of the Washington Higher Education Telecommunications System (WHETS). WHETS provides two-way televised interactions whereby the students and lecturer at different locations can see as well as hear each other.

The experimental course was deemed successful and useful so it was elevated to permanent status. At first the course was taught in alternate years until it became unmanageably large (about forty students in 1991). Since that time the course has been taught every year, averaging close to twenty students each time.

Although lectures can be provided to non-WSU campus students via WHETS, laboratory sessions have proven to be much more difficult to provide for such students. Various solutions have been tried over the years, but network communication problems have always reduced the effectiveness of the labs for these students. Non-WSU campus lab sessions are only attempted when off-site student demand requires it (this is not every time the course is offered).

This course is not required by any graduate program. Indeed, for many students, this course is "extra," that is, added on top of their full curriculum. Viewed from this context, the growth of MBCT is quite remarkable.

Since faculty researchers are often loath to allow their students to take extra (i.e. non-required) classes, we have noticed a trend in which one student from a given laboratory would take the course. That student would then become the "local expert" and serve the needs of everyone in the given laboratory, consulting with the Center for Visualization, Analysis, and Design in the Molecular Sciences (VADMS) staff only for the more difficult problems. Once this student nears completion of his or her degree, the faculty researcher would then allow another student to take the course. As an alternative, other faculty have had their technicians take the class, so their students don't need to. We speculate that these gate-keeping processes tend to keep class sizes smaller than they would be otherwise, although hardware limitations also restrict enrollment.

The first time the course was taught, students turned in their final projects as term papers. While grading the papers, it became evident that the students would gain by seeing how their classmates used exactly the same tools in different ways. This stimulated the creation of "The Poster Session Final," which is conducted like a scientific meeting, complete with an abstracts booklet. To encourage the students to visit each other's poster, a contest was initiated whereby the students vote for the "best" poster in the meeting. Such encouragement might in fact be little needed, for the students are generally eager to see what their classmates have done.

The Poster Session Final serves a number purposes. The graduate students want to make a favorable impression on the faculty in general and on their advisors in particular, and so the poster session provides an incentive to put in extra effort, which is often considerable (indeed, this extra effort gives rise to faculty complaints as the students tend to disappear from their respective laboratories, but it is not uncommon for such complaints to be muted when the usefulness of the completed project becomes apparent). The poster session broadcasts to the entire university the computational biology tools that are currently available in the VADMS suite. The students gain the experience of preparing a poster before going to a national meeting (in fact, several posters from MBCT have been taken directly to national meetings with little or no alteration and publications have resulted from the class projects). By having the students questioned by interested, non-scientists, the importance of communicating with the general public is emphasized and practice at this skill gained. Finally, the students really become enthusiastic about their projects; for many, it is the first time they have presented the results of their own work.

Because of the nature of MBCT, it is possible for students to complete a reasonable research project during one semester. Completing a project to the stage of being able to prepare a poster would be much more problematic for wet-laboratory experiments; yet seldom has any student failed to reach this stage in MBCT. For this reason, courses in computational molecular biology seem ideal for developing this alternative approach. Although organizing such a poster session requires much time and effort, the benefits seem well worth the costs.

For reasons we don't understand, students who take MBCT tend to be too gullible with regard to the results generated by computers. We have tried to counteract this tendency in several ways. In the lectures, we point out mistakes that have been made in the past, including misalignments of sequence and false identification of folding motifs, both of which became evident when the 3D structures were determined. In the laboratory, we have developed exercises demonstrating that different algorithms currently in use give different answers to the same problem; for example, the DSSP program by Kabsch and Sander (1983) and the Define_Structure program by Richards and Kundrot (1988) often give quite different secondary structure assignments for the same 3D coordinates. We even have a warning that is repeated at every opportunity and prominently displayed on the annual class T-shirt: "Don't expect your computer to tell you the truth" (von Heijne, 1987).

On the positive side, we emphasize that one can do true experiments, with variables (such as changing the default parameters) and controls (testing the results using sequences or structures known to fit the category under investigation and known to not fit the given category). Just as for wet-laboratory experiments, the computer experiments serve to test the validity of a given result. Despite the negative warnings and positive examples, many of the students remain insufficiently critical of the computer out-puts. This is exasperating and suggests a fundamental problem that we just don't grasp.

MBCT: some details. The lecture portion of MBCT has been given by a collection of individuals, including faculty from several departments at WSU, scientists from PNL (via WHETS), and staff of the VADMS Center, which was started in 1986 to bring the methods of computational molecular biology to the WSU campus. The laboratory portion of MBCT has been developed and operated entirely by the staff of the VADMS Center.

MBCT is taken by a broad spectrum of graduate students, technicians, and an occasional senior-level undergraduate in the biological sciences. A recently implemented undergraduate course will better serve the needs of the interested student at this level. The majors of the students span the biological spectrum, from agronomy to zoology. Clearly, the tools of molecular biology are being applied across the biological domain.

To tailor the course to the individual needs of this diverse group of students, the student projects are emphasized. The Poster Session Final accounts for 50% of the course grade, and the (closely related) laboratory exercises for another 30%. The single examination covering the lecture material accounts for only 20%.

Students from three sources typically take the course: those from WSU, the University of Idaho (UI), and WSU-Tri-Cities (usually PNL employees). Most are first or second year graduate students (UI and WSU) or research staff (PNL, WSU, and UI). Most have little or no prior computer experience, save word processing.

As we have not found a suitable text for MBCT, we simply provide a list of pertinent references and put extra copies of these books and papers on reserve in the library. Emphasis is placed on several monographs that are more comprehensive and are generally at a more appropriate level for the introductory nature of the class as compared to recently published research papers. These include: Brooks, III, C.L., Karplus, M. and Pettitt, B.M. *Proteins: A Theoretical Perspective of Dynamics, Structure and Thermodynamics* (1988); Doolittle, R.F. *Of Urfs and Orfs* (1987); Doolittle, R.F.(editor) Molecular Evolution: Computer Analysis of Protein and Nucleotide Sequences, *Methods in Enzymology* (1990); von Heijne, G. *Sequence Analysis in Molecular Biology* (1987); and Gribskov, M. and Devereaux, J. *Sequence Analysis Primer* (1991). In addition, various lecturers do include optional references to recently published papers for those students who want to dig deeper into particular (often project-related) topics.

Upon entering the course, students typically have only vague ideas regarding their intended projects, and lack the information or understanding to know whether they should pursue the sequence analysis track or the molecular modelling track in the laboratory. Thus, the first lecture and the first laboratory strive to present an overview of the capabilities and uses of the current software available to students in the course. At the end of the second week, each student meets with one or more of the instructors, where the student describes his or her intended project. At this meeting, advice is given (typically to scale back the project, with the possibility of expanding it later if time permits) and definite steps are laid out. This meeting also serves to identify the few students who entered the course with no project in mind. For these students, a second meeting is scheduled and a project is suggested. All students are required to turn-in a one page summary of their intended project at the end of week seven. This forces the students to think more seriously about their project in advance of actually doing it and identifies students who are still having problems.

Of course, the nature of a student's project determines which laboratory track, sequence analysis or molecular modelling, that the student will follow. To accommodate the time required for project and computer skills development, the first two weeks of the laboratory are the same for all students. The two tracks remain very similar to each other during the third and fourth weeks, with only differences in emphasis. This similarity gives the student extra time to switch tracks if necessary. At the fifth week the divergence between the two tracks becomes very pronounced, so a student should have decided on which track to follow by this point in the course. Although rare, students have switched tracks later than the fifth week. One remarkable student obtained the coordinates of a protein with high similarity to his sequence of interest directly from the crystallographer by electronic mail before it was deposited in the Protein Data Bank, but this occurred late in the semester. Therefore, the student backtracked to finish the modelling track in addition to sequence analysis and, for his class project, built a model of his protein based on the 3D structure of the similar one.

All the class requirements, except the poster, are completed by the beginning of the eleventh week. The last lecture is on Monday of the eleventh week. This means that all the lecture material is covered before the students begin serious, full-time work on their projects. The only examination in the course, a take-home, is given out at the end of the penultimate lecture and returned after the week-end at the last lecture. If students have kept up, the laboratory exercises are completed by the end of the tenth week as well. Thus, the last five weeks of the semester (six weeks including WSU's week-long holiday for Thanksgiving) are used by the students for a concentrated effort to complete their class projects. An unintended benefit from this structure is that this course could be easily adapted to the quarter system, with the lectures and laboratory in one quarter and the project in the next.

To increase campus attendance at the Poster Session Final, a seminar is presented one hour before the poster session is to begin. Given the time of day (3:10 PM), most of the audience choose to go to the Poster Session Final, which opens at 4:00 PM, rather than to return to their laboratories. Speakers are invited who are well-known and who have made interesting uses of computational methods in the study of proteins and/or nucleic acids and their sequences. Past speakers include Frederick Richards, Barry Honig, Mitchell Sogin, and Garland Marshall; the next speaker is scheduled to be George Rose.

Typically, the Poster Session Final lasts until 7:00 PM or 8:00 PM. This gives enough time for the general public and the graders to view the posters and ask questions of the students. After that, the students have time to look at all the other posters and talk to their fellow students about what they did and why. Then they vote for the best poster of the session, after which the session is over. Due to the long duration of the actual poster session itself, pizza, and beverages are served about 6:00 PM. The following day the posters are removed to another site for more careful grading by the class instructors. After one week, the posters become available for pick-up by the students.

MBCT 1994 - a specific example. The materials from MBCT when it was last taught in the Fall Semester of 1994 are presented. Following are the lecture titles, brief descriptions of the laboratory exercises, and some information regarding the Poster Session Final, including titles and authors of the posters displayed in 1994:

Lecture Schedule for MBCT, Fall 1994.
Note: Lectures were given MWF for the first ten weeks of the semester

#	Date	Topic
1	8/22	Introduction: Computers, Structures, and DNA
2	8/24	Sequence Databases: Content and Organization
3	8/26	Structural Databases: Content and Organization
4	8/29	String Searches and Their Uses
5	8/31	Dot Matrix Methods
6	9/2	Alignments and Substitution Matrices
Friday	*9/2*	*Project Conferences - To Be Arranged*
Monday	*9/5*	*Labor Day Holiday*
7	9/7	Multiple Sequence Alignments
8	9/9	Database Searching: Old and New Methods
9	9/12	Nucleic Acid Sequence Characterization
10	9/14	Finding Remote Relationships: Profiles
11	9/16	Sequence Alignments and Molecular Evolution
12	9/19	Sequence Alignments and Ancient DNA
13	9/21	Trends in Scientific Visualization
14	9/23	Visualization of Molecular Structures
15	9/26	Calculation of Structure and Properties: Overview
16	9/28	Molecular Mechanics: Energy Minimizations
17	9/30	Molecular Mechanics: Future Direction
18	10/3	Molecular Dynamics: Background

19	10/5	Molecular Dynamics: Applications
20	10/7	Amino Acid Sequences and Their Attributes
Friday	*10/7*	*Project Statements Due*
21	10/10	Secondary Structure Prediction I
22	10/12	Secondary Structure Prediction II
23	10/14	Secondary Structure Prediction III
24	10/17	Neural Networks
25	10/19	Prediction of Protein Tertiary Structure I
26	10/21	Prediction of Protein Tertiary Structure II
27	10/24	Docking and Drug Design
28	10/26	Homology Modelling I
29	10/28	Homology Modelling II
Friday	*10/28*	*Take Home Midterms Given Out at End of Class Period*
Monday	*10/31*	*Take Home Midterms Due at Beginning of Class Period*
30	10/31	Implications of the Human Genome Project

MBCT Laboratory Exercise Outline, Fall 1994.

Note: Majority of software from the Genetics Computer Group (GCG, v. 8.0) and Columbia University (MacroModel, v. 3.0).

Exercise #1: Introduction to Computing

MM and SA Tracks: A basic introduction to computers and the computing platform from which this course is taught is provided. Specific topics include: 1. an introduction to computers; 2. equipment used in this course; 3. VAX background information; 4. DCL; 5. MAIL; and 6. A demonstration tour of the software available in the VADMS Center.

Exercise #2: Practicing the Basics

MM and SA Tracks: Further work using DCL commands and VAX utilities is carried out. Specific topics include: 1. keyboard mapping and editing; 2. using the EVE editor; 3. more practice with DCL and various utilities; 4. expanded communications procedures, such as SEND, SMTP with MAIL, and PHONE; and 5. command file background information and batch processing.

Exercise #3: Molecular Biology Databases (emphasis varies with track)

MM and SA Tracks: The purpose of this exercise is provide an introduction to the content and structure of several of the databases used by molecular biologists. Specific topics include: 1. background information on ascii and non-ascii databases; 2. simple ascii databases; 3. complex ascii and mixed ascii/binary databases - Brookhaven's ProteinDataBank (PDB) and GCG format GenBank, EMBL, PIR, and SwissProtein; and 4. a non-ascii database - Cambridge Structural Database and QUEST (Allen, et al., 1983).

Exercise #4: Entering Data

In this exercise all students explore the background and detail of specific databases that relate to the track they have chosen with an overall emphasis on data entry. Both tracks learn sequence data format including GCG and PIR (Protein Identification Resource), reformatting and data conversion. Molecular modelling formats for small and large molecules are also reviewed by both tracks.

MM Track: Specific topics include: 1. entering structural data via the EVE editor and via MacroModel; 2. data conversion between binary and ascii formats; and 3. GOPHER access to PDB database.

SA Track: Specific topics include: 1. entering sequence data via the EVE editor and via GCG's SeqEd and SetKeys; and 2. sample gel entry using a lightbox and sample autorad.

Exercise #5:

MM Track - Protein secondary structure: This exercise on protein secondary structure prediction from amino acid sequence has the overall objective of pointing out the limitations of current methods. Specific topics include: 1. protein secondary structure information and prediction; 2. prediction reliability; 3. circular dichroism - background information and actual fitting of CD spectral data; and 4. locating secondary structure information in PDB files and comparing authors' assignments with those from the programs Define_Structure and DSSP, and from secondary structure predictions of primary sequences.

SA Track - Probe design, the "guessmer:" This exercise focuses on what are often the first steps in many molecular biology projects. Typically these involve either probing genomic digests, shotgun clones or cDNA libraries, or using PCR to amplify some desired stretch of DNA. All require suitable oligonucleotide probes. Specific topics include: 1. finding consensus elements using PileUp and PlotSimilarity; 2. creating a consensus sequence with ProfileMake; 3. creating potential probes using SeqEd and BackTranslate; and 4. testing the probe with StemLoop, Gap, Prime, and FindPatterns.

Exercise #6:

MM Track - Advanced data entry techniques: Advanced data entry techniques are the focus of this exercise, including the problems of recognizing and correcting data input errors. Also covered is the modification of data files for specialized needs. Specific topics include: 1. MacroModel its features and limitations; 2. growing a protein; 3. converting data files to be used with MacroModel; 4. customizing a data file; 5. color coding protein backbones; and 6. creating a small peptide model and exploring solvent effects on its conformation.

SA Track - Contig assembly: In this exercise, GCG's Fragment Assembly System (FAS) is used to reconstruct a complete gene sequence from fragment data sets provided. Specific topics include: 1. using GelStart to initialize the system; 2. entering fragment data with GelEnter; 3. aligning fragments to create contigs with GelMerge; 4. checking out the resultant alignment and editing it with GelAssemble; 5. visualizing the contigs with GelView; and 6. building larger and larger contig alignments through FAS's iterative nature.

Exercise #7:

MM Track - Physical characteristics of molecules: The focus of this exercise is the determination of the physical characteristics of proteins Creating and using alpha carbon traces and the superpositioning of molecules is also covered. Specific topics include: 1. estimating surface areas and volumes; 2. visualizing exposed charged groups; 3. displaying hydrogen bonding; 4. coloring the determined secondary structure of proteins; 5. doing superpositioning on simple and complex molecules; and 6. creating alpha carbon protein traces and superpositioning them.

SA Track - Gene finding strategies: This exercise gives an introduction to methods used in the recognition of coding sequences. Specific topics include: 1. defining urfs and orfs; 2. translating frames using Map; 3. identifying potential genes using signal methods - FindPatterns, Terminator, Repeat, StemLoop, and weight matrices in FitConsensus - to locate promoters, terminators, repeat regions, and splice junctions; 4. finding genes with content methods, what the sequence "looks" like and "nonrandomness" techniques - TestCode, Frames, and CodonPreference; 5. translation issues - where to start and stop, and exons and introns; and 6. network methods such as GRAIL, NetGene, and GeneID.

Exercise #8:

MM Track - Molecular measurements and docking: This exercise covers methods for molecular measurement and docking procedures. Specific topics include: 1. making molecular measurements; 2. measuring distances on Van der Waals structures; 3. exploring atom distances; and 4. docking molecules.

SA Track - Database searching, multiple sequence alignment, and profiles: These methods provide insight into the mechanism, structure/function relationships, and evolution of biological molecules. Specific topics include: 1. the advantages and disadvantages of protein versus DNA searching; 2. the algorithms - FastA, TFastA, WordSearch, and BLAST; 3. understanding limitations, interpreting results and significance, similarity and homology; and 4. multiple sequence alignment analysis using PileUp and the Profile suite.

Exercise #9:

MM Track - Homology modelling: This exercise provides an introduction to the homology modelling of proteins. Specific topics include: 1. collecting data on which to make alignments; 2. creating alignments; 3. refining collected data into a usable form; and 4. overlaying a sequence upon another's coordinates.

SA Track - Display and prediction of protein attributes: In this exercise the analysis, display, and comparison of amino acid sequences using properties such as hydrophobicity, antigenicity, CD, and secondary structure prediction are presented. Specific topics include: 1. mapping physical characteristics using PeptideMap, PeptideSort, and IsoElectric; 2. hydrophobicity profiles using the locally developed programs PK23 and GES; 3. hydrophobic moment analysis using HelicalWheel and Moment; 4. antigenicity prediction using AMPHI; 5. secondary structure predictions using PIR's Cho-Fas and GCG's PepPlot and PeptideStructure/PlotStructure combination; 6. network predictions using PredictProtein and NNPredict; and 7. experimental estimates of secondary structure by CD using the local programs ENTERCD and CDFIT. Customizing run parameters and interpretations to the particular situation at hand, e.g. with window sizes and globular versus membrane proteins, is stressed in this exercise.

Exercise #10:

SA and MM Tracks - General review and poster preparation: This exercise involves a discussion of the strength and weakness of the various algorithms the students have used. The general discussion is followed by instructions with regard to the development of the posters for the Poster Session

Final. A required tutorial chosen from the Optional Exercises booklet is used to produce part of an informative and visually interesting poster.

Poster session final. About one week before the Poster Session Final, a seminar announcement is distributed. In 1994, the pre-poster session seminar was given by Garland Marshall of the Center for Molecular Design, Washington University. The title of his seminar was "Computer-Aided Molecular Design."

Abstracts describing the projects are due a week before the date of the poster session to allow time to assemble an abstract booklet. The abstract booklets are available to everyone that comes to the poster session. Following are the titles and authors taken from the 1994 abstract booklet:

1. Profile Analysis of Isocitrate Lyase Family, by Abdur Rehman.
2. Characterization of a Novel Lipoxygenase cDNA in Soybean: Response to Nitrogen and Methyl Jasmonate, by Lowry C. Stephenson.
3. A Putative Protein from a Newly Isolated cDNA from *Arabidopsis thaliana* is Strongly Related to Iron-Containing Fatty Acid Desaturases, by Marlyse Peyou Ndi.
4. Sequence Analysis of the Gene for Component A of NTA Monooxygenase, by Yonguri Xu.
5. Characterization of plasmid pCC5.2 from *Synechocystis* PCC 6803, by Weidong Xu.
6. Analysis of a Sec-Independent Secretion Pathway in *Yersinia entrocolitica*, by Michael J. Smith.
7. Sequence Analysis of a Plasmid Gene Encoding a Potential Membrane Protein, by Dong Han.
8. Variable Surface Protein of *Giardia lambila,* by Kirsten Bengston.
9. Design of Probes for Chitinase Gene in *S. lydicus* WYEC108 and Construction of Codon Usage Table for *Streptomyces sp*. Using GCG programs, by Brinda Mahadevan.
10. Immunological and Functional Relevance of Conserved Regions of Heat Shock Proteins of *Mycobacterium sp*, by Carlene Emerson.
11. Evolutionary Analysis of the G-protein Coupled Receptor Subfamily Glycoprotein Hormone Receptors: FSHR, TSHR, and LHR, by Tracy Lloyd.
12. Phage Display of the A/T-DNA-binding Domain of Random Oligopeptides, by Jeong S. Oh.
13. Homology Modelling of Pentachlorophenol 4-Monooxygenase Reductase (Pcp D) Using Phthalate Dioxygenase Reductase(PDR) as a Model, by Suchart Chanama.
14. Specificity of Hydroxylation of (-)Limonene in Peppermint *(Mentha piperita)*, Spearmint *(Mentha spicata)*, and Perilla (*Perilla frutescens),* by Marie Rufener.
15. Primers to Use in Very Long Amplification Methodology, by Bernard Miller.
16. Neural Net Analysis of Zinc Finger Motifs, by Mark Lambert.
17. Modification to CAGE/GEM Using Remote Access to GenBank Sequence Databases, by Ron Suguitan.

Grading sheets with instructions are provided for individuals chosen to participate in the examination process. Volunteers typically grade only a few posters each, but, in aggregate, most posters are graded by at least one scientist who is not an instructor for the class.

Concluding remarks. Sequence- and structure-based computational approaches for understanding the structure/function relationships of the informational biomolecules have been combined into a single course, Molecular Biology Computer Techniques. From our discussions with others, the more usual approach is to teach separate courses in sequence- and structure-based approaches. However, given the interplay between sequence and structure, we believe that the synthesis of both topics into a single *introductory* course has merit. A major limitation with our one semester course, however, is the lack of time devoted to each topic, which necessarily constrains the lecture portion of this course to be more of a survey. A possible alternative would be a course that spans the entire academic year, with integration of the sequence- and structure-based approaches and with considerable time devoted to the interplay between the two. However, such a course probably lies several years in the future, after the faculty in the various departments in the biological sciences have come to more fully appreciate that these emerging computational methods are, indeed, indispensable.

References.

Allen, F.H., Kennard, O., and Taylor, R. (1983) *Systematic Analysis of Structural Data a a Research Technique in Organic Chemistry.* Acc. Chem. Res. **16**: 146-153.

Brooks, III, C.L., Karplus, M., and Pettitt, B.M. (1988) *Proteins: A Theoretical Perspective of Dynamics, Structure and Thermodynamics*. Ad. Chem Physics, LXXI, Wiley and Sons, New York, NY.

Doolittle, R. F. (1987) *Of Urfs and Orfs, A Primer on How to Analyze Derived Amino Acid Sequences*, University Science Books, Mill Valley, CA.

Doolittle, R.F., editor (1990) Molecular Evolution: Computer Analysis of Protein and Nucleotide Sequences, *Methods in Enzymology*, **183**, Academic Press, New York, NY.

Genetics Computer Group (1994) *Program Manual for the Wisconsin Package*, Version 8, 575 Science Drive, Madison, Wisconsin, USA 53711.

Gribskov, M. and Devereaux, J. (1991) *Sequence Analysis Primer,* W.H. Freeman Press, Salt Lake City, UT.

Kabsch, W. and Sander, C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* **22**: 2577-2637.

Richards, F. M. and Kundrot, C.E. (1988) Identification of structural motifs from protein coordinate data: Secondary structure and first-level supersecondary structure. *Proteins* **3**: 71-84.

Still, W.C., Richards, N.G.J., Guida, W.C., Lipton, M., Liskamp, R., Chang, G., and Hendrickson, T. (1990) MacroModel Version 3.0, Department of Chemistry, Columbia University, New York, NY, USA 10027.

von Heijne, G. (1987) *Sequence Analysis in Molecular Biology, Treasure Trove or Trivial Pursuit,* Academic Press, New York, NY.

Educational Issues in Biocomputing: Introduction to Session

[†§]Steven M. Thompson, [†§]Susan J. Johns, and [§]A. Keith Dunker
[†]Center for Visualization, Analysis, and Design in the Molecular Sciences, and
[§]Department of Biochemistry and Biophysics, Washington State University,

Pullman, WA, USA 99164-4660

{prcadams, thompson}@ribozyme.vadms.wsu.edu; dunker@mail.wsu.edu

Biocomputing, especially as applied to molecular biology, has two main historical branches: 1. biomolecule crystallographers and their cohorts began to develop computational approaches for gaining additional insights and understanding regarding the structures they had determined; and 2. with the advent of the massive databases of sequences of nucleotides and amino acids, molecular biologists, mathematicians, and computer sciences began to explore ways of discovering the biologically relevant organizations within these data fields. These two branches have not yet coalesced into a single field of study, though they are intimately intertwined and may someday merge; departments of biocomputing are not yet common. Nevertheless, a few scientists and university faculty, who are scattered across various departments, colleges, computer centers, and private companies, have begun to develop courses and workshops to teach, or to provide learning environments for, the use of the new tools of biocomputing.

Without a defined field or a formal departmental structure, and given the very brief history of biocomputing, special problems are encountered in the development of course and other educational materials. Even the most basic questions such as content and organization are unclear.

In our informal (often e-mail) discussions with various individuals who, like us, have been involved with the development of these educational materials, we have discovered a common thread of problems and experiences, repeated individually many times over. For this reason, we proposed that the Pacific Symposium on Biocomputing host an educational forum so new entrants into the educational aspects of this field could gain from all of our prior experiences and could also gain access to course materials that have been previously developed.

Because biocomputing is progressing so rapidly and is in such a state of flux, it is our belief that educational forums in this area should be tied closely to vigorous research meetings. Every effort should be made to maintain a close connection between teaching and discovery.

Educational issues at the Pacific Symposium on Biocomputing will be presented and discussed in three formats: oral presentation of papers, a poster session, and a workshop/panel discussion. The workshop will focus on several specific questions: 1. which students need formal training in biocomputing; 2. when should such courses be inserted into their curricula; 3. what core skills, and in what order, should be taught; 4. how can educators keep abreast of the rapid changes in this field; and 5. visualization, is it a necessary tool or merely window dressing? We intend to record and transcribe the discussions on these topics and to make them available to those interested on the World Wide Web at:

http://ribozyme.vadms.wsu.edu/~VADMS/PSB-edu.html

The papers to be presented orally or in the poster session span the range of current options for providing instruction in biocomputing. Each of the papers will be described in turn as follows:

The paper by Swanson and Lybrand is especially interesting because it presents a course in biocomputing implemented for students with backgrounds in physics and engineering. The course uses a more quantitative and physics-based approach and assumes a high computer literacy for the students.

At the other end of the computer-skills spectrum, our paper (Johns, Thompson, and Dunker) describes the history of, experiences with, and rationale for a course designed primarily for molecular biology students with little or no prior computing experience. This course provides a hands-on, “real-life” approach in which student-initiated projects are emphasized.

Whereas these first two papers focus on intelligent and correct usage of currently available algorithms in computational molecular biology, the paper by Altman and Koza describes a higher level course that strives to present a technical introduction for students in computer and medical information science. This programming course initially focuses on fundamental representations and algorithms already in use in biocomputing, and then on current research in these areas.

The paper by Gaëta, Bucholtz, Campbell, Huynh, Kim, and Reisner describes the Australian National Genomic Information Service (ANGIS) and their approach for training in biocomputing. The fact that they address the needs of all Australia is particularly impressive. Details are given on a set of self-paced tutorials, a system access recipe book, and various workshops provided by ANGIS.

A prototype course using the World Wide Web by the Biocomputing Division of the Virtual School of Natural Sciences is outlined in the paper by De La Vega, Giegerich, and Fuellen. Offered in the summer of 1995, this course reached 34 students in 13 countries. This paper provides an illuminating and interesting discussion on the use of the World Wide Web to teach a course in computational molecular biology to the entire world.

We hope that the set of papers just described and the workshop/panel discussion that will occur will be found to be useful to the molecular biology community. We welcome your comments and long range participation, either through future meetings, that we hope will be a continual part of the Pacific Symposium on Biocomputing, or by comments sent to the WWW site given above.