Appendix C: Software Papers and Case Studies

GUIDELINES FOR SOFTWARE PAPERS IN *AGRONOMY JOURNAL*

The software section of *Agronomy Journal* (AJ) publishes papers related to computing in agronomy. Papers may focus on measurement or analysis software, monitoring and control software, visualization, specialized databases and data structures, computer utilities for agronomists, comparisons of algorithms or programs, and decision support systems. Software manuscripts may be submitted as articles, Letters to the Editor, Notes, or Forum papers. While AJ encourages authors to make their software available at no or low cost, including source code, it welcomes articles dealing with both freely distributed and commercial software. Papers on simulation modeling are more appropriately submitted to the Agronomic Models section of AJ, and papers focusing on statistics should be submitted to the Statistics section. Papers focusing on educational or extension applications of computers should be submitted to *Natural Sciences Education* (NSE). Field applications of existing software, databases, or decision support systems typically belong in the related subject section (Agroclimatology, Crops, Integrated Agricultural Systems, Integrated Pest Management, or Soils). The editor reserves the right to reassign a paper to another section.

As with other manuscripts submitted to AJ, software manuscripts must adhere to accepted scientific standards in terms of a review of relevant literature, testing, and documentation of methods. The authors must show that the software performs its stated function, which generally requires testing the methodology and presenting one or more sample applications. Because the emphasis is on the software, use of previously published data is acceptable where appropriate. Authors are encouraged to make programs and documentation available at the beginning of the review process and may be required to do so at the editor's discretion.

Manuscripts submitted as software papers to AJ should address the following issues:

1. What are the intended uses and users of the program?
   a. What does the software do? How well does it do it? Does it do it better (faster? At lower cost?) than existing methods or products?
   b. What need does the program address? How is the need currently being addressed, if at all?
   c. Is the program intended for researchers, producers, consultants and extension agents, or some other group?
   d. What are the limitations of the program? Are there assumptions in the program that prohibit its application to apparently related uses?
2. How does the program work?
   a. If the program uses a new algorithm, it should be described; if an established algorithm is central to it, it should be referenced. New algorithms can be described using a flowchart, pseudocode, a short segment of actual code, or a written description.
   b. Where helpful to the reader, longer sections of code can be provided as an appendix. Format code in a monospaced font (e.g., Letter Gothic or Courier) that clearly distinguishes between 1, l, I, O, and 0. Indent lines only where code structure requires it. If there can be any doubt about runover lines vs. new lines, add a marginal or cover note of explanation.

c. If the logical flow of the entire program is of interest, it should be described via a flowchart, pseudocode, or a written description.

3. What is required to run the program?
   a. Is the program commercial, or free? If free, is it executable-only, or is the source code included? If the program is commercial, the seller, price, and user support agreements should be mentioned; if free, the copying policies, programming language, and the specific compiler or interpreter required should be specified.
   b. What operating systems will it run under?
   c. What hardware is required to run the program? Include both minimum and optimal configurations; specifying memory and graphics requirements, and any required or useful peripherals.
   d. How much disk space is required to store the program and associated files? How big are the required input and output files?
   e. Are there any supporting software requirements? For example, does the program require an interpreter or a specific spreadsheet program?
   f. How can the program be obtained? If possible, give both a URL for downloading (or email, for reply with file attachment) and a physical address. For the physical address, include mailing instructions.
4. How does the program operate from the perspective of the user?
   a. Is there a manual available? Is it in the form of a book, a file, or online help? Is any support available from the authors, the seller, or some other party?
   b. How is the program invoked and operated?
   c. Where truly informative, screen shots may be provided as figures. Avoid merely decorative screen shots that largely duplicate the text.
   d. Where sample output is useful, provide it as a table (i.e., a text file, not a graphics file). For publication, this will be formatted as a standard table, close to but not necessarily identical to the appearance of the output submitted.
   e. What inputs are required, and in what format? How are these inputs obtained?
   f. What outputs are produced, and in what format? What does the output mean?

January 2013

NSE PUBLICATION POLICY: CASE STUDIES AND SOFTWARE

Contemporary interest in providing problem-solving and decision-making experiences in education has prompted the adaptation of decision cases to agricultural, natural resource, and life science situations. The NSE editorial board reviews decision cases suitable for use in classroom or extension education situations. The following guidelines describe the format for publication of decision cases. Prospective authors will find it helpful to consult these guidelines in manuscript preparation to ensure minimal editorial delay.

Decision Case Guidelines in NSE

I. Criteria for Evaluation

Primary consideration is given to *original* cases that describe actual situations (not simulations) requiring a decision. Decision cases should foster integration of concepts, use of problem-solving skills, application of technical information, and/or consideration of human, societal, and ethical factors. Appropriate decision-maker roles for published cases include producers, scientists or other professionals, educators, and policymakers. Criteria for acceptance of decision cases are:

1. Cases must describe *an actual (not simulated) situation* that *advances understanding or teaching of decision making*.
2. Cases must be *thorough and well documented* (e.g., adequate exhibit support).
3. Cases must *address topics and issues of interest* to a broad educational audience.
4. Cases must be *clearly and concisely written*.

II. Format Specifications

Abstract. A clearly worded abstract of the case situation including description of the decision maker, decision focus, key issues, and case objectives/use. The abstract should contain a maximum of 250 words.

The case. The case text should be interesting and easy to read. An introductory paragraph *preceding* the case should set forth the context of the case, including citation of other published cases of relevance to the case being presented. The case description should permit the reader to fully understand the background and specific considerations of the case. The text should allow the reader to readily identify with the decision maker(s) and the decision. The objectives of the decision maker should be evident in the case, either by explicit mention or by inference from other case information. The alternatives or options of the decision maker in dealing with the issues should also be clear to the reader. The concluding paragraph of the case should refocus on the major issue(s). It is convention to write cases in past tense.

Exhibits. Effective cases are usually supported by relevant exhibits. Examples of exhibits include data bearing on the decision, illustrations, background documents, and correspondence, among others. Exhibits should be drawn from actual, unaltered sources (exceptions may be made when confidentiality must be protected) and should be referenced in the appropriate places within the case text. Case exhibits should be well organized and concise and should not contain information that is irrelevant to the case. Exhibit information taken directly from published works should be referenced. Exhibits should be numbered in the same order as they are referenced in the case.

Teaching note. The teaching note describes the objectives of the case and the principle issues considered. This section of the manuscript should provide the reader a concise interpretation of the significance and educational value of the case. The section should also

describe how the case has been or may be used in a classroom or extension education context. If the case has been used, the teaching note may provide a summary of student evaluations of the case. The teaching note may also include the author's analysis of the case, although the detail provided in this analysis may be limited to protect the potential use of the case by readers. Educators interested in teaching the case can usually obtain a full copy of the author's analysis by corresponding directly with the author. The teaching note is particularly important for assisting readers in deciding whether or how to use the case.

References. Citable references in the case text, teaching note, or exhibits should be listed. Use the author–year system for citing references.

Abridged case format. Some cases cannot be published as complete cases due to their length or complexity. Such cases may be published in an *abridged case* format. All abridged cases submitted for publication, regardless of length or complexity, *must be reviewed in their entirety* prior to acceptance. No case will be accepted unless both the complete case and abridged version have been favorably reviewed by the reviewers and editors. Text of abridged cases should be identified as "Case (Abridged)." The text of an abridged case, as well as the teaching note, should be of sufficient length and detail to permit readers to understand the nature of the decision, the identity of the decision maker(s), the principal issues of the case, and the educational value of the case. The abridged text and teaching note should contain sufficient information to allow readers to assess the potential for use of the case. Important exhibits should also be presented whenever possible. As a minimum, abridged cases should contain a complete list and brief description of all exhibits referenced in the complete case. If readers are interested in teaching a case published in abridged format, they should request a copy of the complete case directly from the corresponding author.

Examples of complete and abridged cases are published in JNRLSE (21:9–14, 15–19, 20–26; 22:134–138, 138–144; 23:98–103, 103–108; 24:169–172, 173–178, 178–184). Prospective authors may reference these for guidance on format and style. Refer to the instructions to authors for contribution guidelines and style information.

COMPUTER SOFTWARE PUBLICATION POLICY

As a natural consequence of increased classroom computerization, NSE authors are submitting greater numbers of articles describing computer software. Previously applicable formats do not have provisions for ensuring that journal readers always receive sufficient information about software to evaluate its utility. At the 1983 ASA annual meetings, the Editorial Board approved the following guidelines concerning publication of computer software articles. Prospective authors will find it helpful to consult this checklist early in their manuscript preparation to ensure compliance and minimize editorial delays. In most cases it would be useful to provide copies of the software along with the manuscript for the review process.

Guidelines for Publishing Computer Software in NSE

I. Pedagogy

The primary thrust of computer software manuscripts must be on pedagogical applications. For papers stressing developmental and/or modeling theory of the software itself, authors will be asked to adapt the paper to a pedagogical theme or to publish in a more appropriate journal.

II. Specifications

Reviewers should check manuscripts for inclusion of the following software specifications:

Encoding Language. C^{++}, Java, VB, PHP, etc.
Minimum Memory Requirements. For example, 128 K.
Minimum Free Hard Drive Space. For example, 1 GB.
Additional Software Programs Required for Use. Microsoft Word, Microsoft PowerPoint, etc.

III. Documentation

Manuscripts should indicate the availability of documentation about the software. Such user aides are frequently printed manuals or instructional files located on the program.

Reference to start-up and ending procedures should also be included in this documentation. In addition, the author is expected to adequately describe input requirements as well as output information and format somewhere in the manuscript.

IV. Availability

In all cases, authors should make some statement about the availability of the software to readers. Options on availability are many, but should be clearly stated. For example, if the software is declared to be public domain material, it may be supplied free upon request, or upon receipt of a USB memory stick. In the case of software offered for sale, the source, proposed fee, and major user-agreement specifications should be included. Also, educators will want to know whether multiple copies for classroom use are allowed.

V. Example

Conforming to these criteria will enable readers to more informatively evaluate the utility of software for their own systems. Item I should represent the major message of the author. Items II, III, and IV should be packaged into a minor, unobtrusive section of the manuscript entitled *Software Specifications*. Two older samples follow:

Software Specifications

The final version of the 2000 Arkansas Envirothon educational resource material CD-ROM contains directions and files totaling approximately 540 Mb. Directions for accessing the CD-ROM are stored in both in a MS-DOS text file (README.TXT) and in rich text format (README.RTF). The files may be accessed by any web browser capable of reading HTML version 3.0 and above and Adobe Acrobat Reader version 3.0 and above compatible with your operating system. Netscape Communicator version 4.07 and Adobe Acrobat Reader version 4.0 for Windows 95 or 98 operating system are included on the CD-ROM with instructions for installing the software. Those wishing to obtain a copy of the CD may contact the current president of the Arkansas Envirothon Steering Committee. Current email addresses are provided on the Arkansas Envirothon website (http://www.uaex.edu/envirothon/).

Software Specifications

The Darcy applets were written in Java 2.0. They can be run from web browsers supporting Java applets. Best results are obtained when used with a free plug-in for the browser. The plug-in for various operating systems can be downloaded automatically from Sun Microsystems the first time it is needed. We have tested the applets with Netscape 4.0 or higher and MS Explorer 4.0 or higher. The computer must have at least

32 MB of random access memory (64 MB recommended), and 15 MB of free fixed disk space. The software can be accessed at http://soilphysics.okstate.edu/toolkit/. The compiled software can be downloaded and stored on a local machine so it can be used without network access. The software can also be loaded on a local web server and incorporated into other web pages. Contact the senior author for details.