Mini-Project: Tool or Analysis Practicum

17-654/17-754: Analysis of Software Artifacts
Jonathan Aldrich (jonathan.aldrich@cs.cmu.edu)

Project Bids due Friday, March 7 at 8pm (for first choice of tool projects) or 5:30pm March 18th at the latest
Interim Report due Thursday, March 20 at 5:30pm
Masters Presentations and Reports due Tuesday, March 25 at 5:30pm
Ph.D. Presentations due Tuesday, April 22 at 5:30pm (may be extended somewhat)
Ph.D. Final Report due Thursday, May 8 at 5:30pm
Masters Projects: 200 points total
Ph.D. Projects: 400 points total

The goals of this mini-project, for Masters students, are to gain an indepth practical experience with an analysis tool or technique and reflect on the experience. For Ph.D. students, the goal is to gain a deeper understanding of the analysis research literature.

The expected scope of effort for this project is around 18 hours per Master's student (36 hours per Ph.D. student); plan your project accordingly.

Groups. 654 students may work on this project in groups of 4-6 (smaller groups are permitted, with scaled expectations, but should discuss this with the instructor). Group projects will be given a single grade. You are free to choose your own groups, subject to the constraint that the instructors reserve the right to assign groups in the case that some students are unable to find partners. Expectations for the project will be scaled (within reason) to the size of the group.

Collaboration Policy. Since different groups will be working on different projects, the only collaboration policy is that your work must be your own (as always).

1 Project Bids

For those choosing tool evaluation projects (typically most groups) please bid by Friday, March 7 at 8pm, for the best choice of tool, or by March 18th at 5:30pm at the latest. Bids after March 7th may be less likely to get their first choice if there are conflicts (this is less of an issue if you want to do a non-tool evaluation project). Your bid should be emailed to the instructor (jonathan.aldrich@cs.cmu.edu) and must contain:

1. The names of the members of your group

2. Choose ONE of the project types below (from Section 5,6, or 7)

3. For tool evaluation projects, give your ordered preference for at least 3 tools. At least 3 of the tools you bid for must be chosen from the commercial tools listed below, for which we have special permission to evaluate in this course, however, you may bid for as many other tools as you like, and place them however you like in the order. If any of the tools you are bidding for are not listed on the course web page, please include a URL for the tool, together with a 2-3 sentence summary of what the tool does and why you believe it is relevant to software analysis.

4. For manual analysis application projects, give a reference (web, paper, or both) for the analysis technique, and if it was not discussed in the class, give a 2 paragraph description of the technique and why you believe it is relevant to software analysis.

5. For development of new analysis techniques, briefly describe the technique you plan to investigate and how you will evaluate whether the new technique is useful.

6. For Ph.D. literature reviews, describe the topic, list 2-3 papers from which you will begin the literature review, and how you will limit the scope.

   The instructor will provide feedback on all bids by Saturday, March 8; some projects may need to be adjusted to fulfill the pedagogical goals of the course. Full credit (10 points) will be given to all bids received on time and in accordance with the specifications above, regardless of any other feedback given.

   For tool evaluations, teams will be assigned tools to ensure that (a) each team evaluates a different tool, (b) some team is evaluating each of the 5 commercial tools we have received special permission to evaluate, (c) as many teams get their first choices, subject to the constraints above. Earlier bids will, all other things being equal, be given higher preference.

2 Interim Report

Your interim report, turned in on Blackboard in PDF format by Thursday, March 20 at 5:30pm, should be about 1 page and contain:

1. The title of your proposed project

2. The names of the members of your group

3. Describe what you intend to do in one-half to three-fourths of a page. For tool evaluations and analysis applications, name which tool or analysis technique you are applying, what artifact(s) you intend to apply it to, and what qualitative and quantitative data you intend to collect. For development of new analysis techniques, give a summary of your ideas so far along with a short worked example. For Ph.D. literature reviews, provide a preliminary outline of your paper and your current bibliography.

4. Describe what you have done so far on the project. For tool projects, a general guideline is that you should have at least run the tool successfully on a small example. For other Masters projects, you should have at least prototype-sized portion of the work done.

Grading for the interim report will be worth 20 points; you will get full credit as long as you turn in a report that demonstrates you have done a reasonable amount of initial work towards the project. The instructor or TA will provide feedback on the interim report by Friday, March 21.

3 Presentations

Each Master's group will prepare a presentation approximately 15 minutes long, for presentation the week of March 25-27. For tool or analysis projects, the presentation should describe the tool or analysis technique if it was not presented in class; describe how the tool or technique was applied, describe qualitative and quantitative data gathered, give a summary of lessons learned, including the benefits, drawbacks, and scope of applicability of the tool or technique. For development of new analysis techniques, describe the new technique in technical detail, and describe your evaluation of the technique.

Presentations will be worth 50 points. You will be graded both on content (what you did) and clarity of presentation. You may use the Powerpoint template provided on the course website, or your own.

4 Project Reports

The final 120 points will be based on the project report. Grading will be based both on content and clarity of communication. The paper should be around 12-15 pages (single-spaced 12 point) for literature reviews, or 5-6 pages of text (diagrams, screenshots, examples will likely take extra space) for other project types, but these are general guidelines only; clarity is more important than length. The contents of the report are discussed in each section, below.

5 Tool or Analysis Application and Evaluation

Choose an analysis tool from the list available on the course website, or an alternative tool or relevant manual analysis technique by agreement with the course instructor. Apply the tool or technique to at least two realistic programs–one that we give you, and at least one that you find on your own. Focus on assessing the strengths and weaknesses of the tool or technique, both in quantitative and qualitative terms.

The assessment must be written with other members of the class as the intended audience. The writeup should briefly describe the tool or technique, describe the experimental setup (for example, how was the tool or technique applied and to what subject), and describe both qualitative and quantitative data gathered in the experiment (some of the requirements for reporting are listed below, but you should be creative and report additional data that you see relevant as well). Based on your experience, discuss the lessons you learned, including the benefits, drawbacks, and scope of applicability of the tool or technique.

The course website includes examples of model tool evaluations done in the past, to give you an idea of what is expected. MSE students are encouraged, but not required, to apply an analysis tool or technique with relevance to their studio project.

You must apply the tool to one standard project that we provide (one in C/C++ and one in Java–you choose which) in addition to one or more projects of your choice (if the tool or technique is not applicable to the projects we provide, bring this issue up with the instructor). You must report the following:

1. How you customized the tool (or would recommend customizing it based on your experience). For example, if your tool reports different categories of warnings, which categories would you or did you turn on or off? Justify your choices.

2. For each project, how many true positives the tool found which were relevant to the project (ideally after customization). Across all projects, give two concrete examples of the warning and relevant code, and an explanation of what the error was and why it was an issue (i.e. two total examples, not two per project).

3. For each project, how many true positives the tool found which were not relevant to the project. Across all projects, give two concrete examples of the warning and relevant code, and an explanation of why it was a technically correct warning but was not of interest in the project.

4. For each project, how many false positives the tool reported. Across all projects, give two concrete examples of the warning and relevant code, and an explanation of why the warning was a false positive.

This year we have the unique opportunity to evaluate 5 high-quality—and pricey—commercial tools. These are **Agitar**, a test-generation tool, and **Coverity, Fortify, Grammatech**, and **Klockwork**, all static analysis tools. We ask you to bid for 3 of these among your bids, as we'd like to take advantage of the generosity of these companies and also provide them feedback based on your experience. The one caveat about these tools is that results from the commercial tool evaluations must be kept confidential to class members; please contact the instructor if this is a problem for you.

Ph.D. (754) student projects will have a relaxed expectation for the practical evaluation of the tool or technique, since there will be fewer people involved in each project. However, Ph.D. project reports are expected to place practical experience with the tool or analysis into a research context: describe the strengths and weaknesses of the analysis compared to other similar results in the research literature, and describe open research problems that are illustrated by your experience with the tool.

6 New Analysis Techniques

With prior permission from the instructor, apply your knowledge of analysis to develop a new analysis technique or an application of an existing technique to a new problem domain. Examples might include defining a new bug-finding analysis, implementing an existing analysis in a new platform, experimenting with a new testing technique, or exploring an idea you have that might lead to more precise alias analysis. You may build your analysis on top of Crystal or any other analysis toolkit, such as Soot from McGill University or SUIF from Stanford University.

You should find some way of evaluating your new technique, typically by applying it to some software artifact. However, the weight given to evaluation is considerably less than the tool evaluation projects because of the analysis development component of the project.

Your writeup should introduce the problem, explaining why your analysis technique is needed from a software engineer's perspective, and why existing analysis tools are inadequate for your purposes. Then describe your new analysis technique in sufficient technical detail that it could be independently reproduced from the description. If your analysis technique is automated, briefly describe the implementation. Describe your evaluation of the technique in the same terms described above, and finally describe what you learned from the experience.

As with tool evaluations, Ph.D. project reports are expected to place practical experience with the tool or analysis into a research context: describe the strengths and weaknesses of the analysis compared to other similar results in the research literature. Ph.D. project presentations will be given on April 24 or 29, and project reports will be due finals week (Ph.D. students do a bigger project in lieu of the quality assurance plan assignment).

While Ph.D. analysis projects are expected to use advanced techniques from the research literature, MSE projects are focused on practical utility. For example, a domain-specific tree-walker analysis that checks an important property for an MSE studio project would be appropriate for an MSE project. For example, in 2005 one team worked with the instructor to define the semantics of a domain-specific language that was relevant to their project. In 2006 a team built a Java Safety Analysis Tool based on Crystal flow analysis.

The instructor may allow Master's students to pursue a more in-depth analysis development project in lieu of the quality assurance plan project; contact the instructor if interested.

7 Literature Review (Ph.D. or MSE with instructor's permission; individual projects)

Choose a subtopic of the analysis literature and analyze the research in this area in depth. Your report should cover the most important recent results in the sub-area, and put them into a comparative framework that shows their similarities, differences, strengths, and weaknesses. Your report should also describe the major open research questions in the area.

Ph.D. students who choose the literature review option will give a presentation in class on April 22, 24 or 29. The length is to be determined, but initially plan on 60-80 minutes. In the presentation, briefly introduce the class to the surveyed area, the technical details, benefits and costs of various proposed analysis techniques, and highlight important open issues that you identified.