Software 1 – Submission Guidelines

General

The following page describes the submission guidelines for the exercises you submit for both automatic check and the hardcopy in the course 'Software 1' at the 2005C (summer) semester. Please read this document through fully - Failing to follow the exact names, files, times, etc. will probably lead to the failure of the automatic checking mechanism, which in turn, will cause a significant reduction in the exercise grade.

The content of the various exercises will be separately detailed in the exercise sheet itself, containing all the items in the submission. In case of collision on any guideline given in this document and a guideline given in the exercise sheet itself the one given in the exercise should be regarded.

This semester the submission will be performed both as a hardcopy and a file sent by an email to the exercise checker (tochna1@post.tau.ac.il). While not sending the sources will NOT lead for failure in the exercise (but to a 0 grade on the wet part), not submitting the hardcopy on time, even if you did send the files will fail the ENTIRE exercise.

Unless stated else in the exercise sheet itself the wet part is 25% of every submission

Procedural Issues

Each student must submit all the exercises by himself/herself. You may share ideas, but you may not share code (or documentation) lines! Severe actions will be taken against students whose work will be suspected as copied!

Homework should be submitted on time. Students, who serve in the army during the semester, are asked to present an official confirmation as soon as he is called to the duty (before the deadline) and will receive additional time for submission. Failing to submit the homework in time due to any other reason is unacceptable and will be regarded as if not submitted.

For other (humanitarian) reasons a delay can be requested from the exercise checker (only! – The exercise checker will not bother the lecturer or the TA trying to find out whether they concurred with some request. The latest time for such requests is 24h before the deadline (to allow the exercise checker some response time), later requests will probably not treated on time.

The exercises will be programs in Java 1.5 and should generate no warnings whatsoever.

Upon reading this document (for the first time) please send an email with the subject 'REGISTER' (uppercase) to the exercise checker (tochna1@post.tau.ac.il), containing (in the following order) your UNIX login, your full name, and your ID number, to register yourself as a student at the current semester. To delete yourself from the submission system send an email with subject 'UNREGISTER' to the same email address, containing the same data. This must be done before submitting the first exercise!

Hardcopy

The hardcopy should be placed, on time, in box #378 (Alperovich Alexander) in the Shreiber building.

Unless stated otherwise in the exercise sheet, the submission hour at the submission date is 17:00 strict.

The printouts you submit should be composed of nicely stapled papers. No plastics whatsoever (please spare me).

The first page of the printout should include the following (no separate page is required, but make it as clean as possible):

- Your full name.
- Your ID number.
- Your user name (login).
- Exercise sheet number.

Softcopy

Please read following carefully – not following any of the following guidelines will lead to 0 grade on the wet part.

The softcopy should be sent, on time, to the exercise checker (tochna1@post.tau.ac.il).

The submission must include a SINGLE ZIP of all the sources required for the exercise. Just to make things clear – sources only. No '.class', no '.jar', no additional documents.

The name of the zip must be USERNAME.zip , where USERNAME is your username on the UNIX machine (lowercase!!!). For example, the file sent by the tochna1 user for any exercise would be 'tochna1.zip'.

The mail sent should have the following line as it's subject and should not contain any additional text in the message body: [Submission #N] (where N is the exercise number, and without the "[ ]"). Writing anything in the message body will not fail the submission, but as the exercise is checked by an automatic tool it will not be read. Any massages you want the exercise checker to read should be with different subject.

In case you want to re-submit an exercise already sent (before the submission deadline) just send it again with the same terms. The last one counts.

File, class and method naming

The names of submitted files must fully correspond (including case) to the names defined in the exercise sheet.

The names of submitted classes/interfaces must fully correspond (including case) to the names (if specified) defined in the exercise sheet.

The names of any public method in a submitted class/interface must fully correspond (including case) to the names (if specified) defined in the exercise.

Package

Please read the following carefully – not following all the guidelines will lead to 0 grade on the wet part.

Unless stated elsewhere in the exercise document itself, the name of your package should be 'il.ac.tau.cs.USERNAME.exN' where USERNAME is your username on the UNIX machine (lowercase!!!) and N is the exercise number. For example, the package name of the tochna1 user for exercise 1 would be 'il.ac.tau.cs.tochna1.ex1'.

Once again, since UNIX is case sensitive, please make sure you keep everything lowercase.

Questions and Support

Unclear issues or problems regarding the Linux environment, the nova machine, or any other system relating issues should be directed to the system consultants. They can be accessed via email (system@cs.tau.ac.il) or in person in the advisors room on the first floor in Schreiber building, room 019.

Other questions, regarding procedural or programming issues should be coped with in the following order:

1. Read the relevant exercise directions.
2. Take a moment and think again whether the question makes sense and cannot be solved alone
3. Check the course's forum for previous answers or ask your question there. Note that the forum is for student use only, the staff may respond to a question, but it is not a formal communication method.
4. Questions about the checking procedures, grades and other should be sent to tochna1@post.tau.ac.il (your exercise checker).
5. If all else fails, send an email to the course mailing list stating the question shortly and clearly and one of the course's staff will answer it. Such an email should be in plain English only and its subject field should be [Software 1]. Students are also welcomed to answer questions sent to the mailing list.

Exercises should be read upon receiving them. Questions regarding the exercise should be submitted up to 5 days before the submission deadline, to allow sufficient time to answer.

Grading Criteria

The grades will be composed of the following parts:

1. **Submission on time**. Handing in printouts or sending the mail after the due are counted as not submitting at all. As a rule of thumb, there will always be an 'objective' problem in the day preceding the submission; a power failure, a printer malfunction, printer running out of paper, etc. Therefore, prepare your homework way in advance. Leave plenty of time for debugging, testing, and dealing with unexpected problems. Unexpected problems are the most expected things in programming projects!
2. **Documentation**. The inline documentation is composed mainly of a reasonable amount of comments (when needed). Make sure that each file and function includes a short paragraph explaining its purpose, IO and method of work. Your program should be understood by reading the documentation only, without viewing a single line of code.
3. **Correctness of the program**. The correctness will be checked by an automatic mechanism, and therefore a special care should be taken meeting the exact syntactic requirements.
4. **Design**. The programs should be well designed using the concepts studied in class (e.g., object-oriented, design by contract)
5. **Implementation efficiency**. Your programs are expected to be reasonably efficient.
6. **Readability and Clarity**. Naming - Use intelligent naming for variables, methods and classes. Modularity - divide code into methods and classes (a by product of good Object Oriented design). Comments.
7. **Indentation**. Sometimes a Tab is better than thousands words. Note that the Eclipse editor can do it automatically.
8. **Honesty**. No extra points will be given for honesty. Any suspicion of deception, however, will cause immediate and severe steps. Please spare me any dealing with this, in such a small group everything is crystal clear, and the staff puts a lot of effort making and checking these exercises and this is eventually only for your benefit.

We wish you all a pleasant and successful semester.