

(67125) Introduction to Object Oriented Programming

# Exercise Preparation and Submission Regulations

## 1. General

- All exercises must follow the guidelines in the **Coding Style** document (see in resources, in the OOP site).
- Some of the exercises will be solved in pairs. For more details, see the **Pair Programming** document.
- To prevent misunderstandings - unless stated otherwise, **exercises are done alone** (see section 10).

## 2. Using Java classes in the exercises

Some of the exercises will involve writing classes which exist in the standard java libraries. In each exercise it will be specifically detailed which classes you are allowed / not allowed to use. Using a class which you were not allowed to use will result in serious point reduction.

## 3. Submissions - contents and requirements

- **README file:** In addition to the program files (only .java files) that contain your solution to an exercise, your submission should include a README file. This file should describe all of the relevant information about the design and implementation decisions you have made. The "README" file has a fixed format, defined as follows:
  - Login Details: The first two lines in the README file should contain the login(s) of the submitter(s) **only, one per line**. If you are submitting an exercise alone, leave the second line blank. Any other characters contained in these two lines may fail the automatic testing.
  - Login Details of Other Students: the third line should contain a list of logins (separated by spaces) of students with whom you consulted and discussed the exercise. If you didn't consult anybody (other than your partner in case of pair exercises) leave the third line blank.
  - File Description: this section contains a list of the program files you submit. The description of each file starts on a new line, and contains its name and a short description (1-2 lines) of its content.
  - Design: in this section, you should document all design issues and decisions you have made (not including those that were provided by us of course). You should briefly describe the design (what are the classes, what are their relationships, etc.). When using inheritance, you should describe the inheritance hierarchy. Finally, but most importantly, you should briefly explain **why** you have made the various design choices.
  - Implementation Issues: this section should contain all important comments about implementation issues that you think would help understanding your code, or should be pointed out since they are important in some sense.

- Answers to Questions: in this section, you will answer the questions you were asked to answer in the README file.
- **Source files**: Each source file should be well documented according to the coding style guide.

## 4. Before submitting

- Your program should compile without errors or warnings on school computers (unless noted otherwise in the exercise). **A submission of a program that fails to compile will result in a 0 score for the specific exercise.**
- Note that typically a java program that compiles at home, on any system, should also compile in the environment of the labs. However, there might be some subtle differences between Windows and Linux systems. To be on the safe side, leave sufficient time to check and test your solution in the labs.
- **Make sure to check the files sent to your e-mail by the submission system.** The submission system (which also checks your code, see ahead) clips lines which are longer than 110 characters in length. This applies to README file as well. **Final submissions with lines cut due to this issue will not be forgiven and will be graded accordingly** (In other words, make sure your lines are no longer than 110 characters).

## 5. Javadoc

- Midway through the course you will learn *javadoc* - a java-specific documentation method which also generates easy to read HTML pages containing your description of your code.
- When introduced, your exercises will be checked for your javadoc implementation - meaning each source file (`Foo.java`) should contain a javadoc header.  
From that point on, the pre-submission tests will also check for successful generation of javadoc from your classes.

## 6. How to submit

- Submission of exercise is electronic, via the course web site. You must be enrolled to the course in order to submit. In case the exercise is submitted in pairs, **only one of the pair members should submit the solution** (let us know as soon as possible if you uploaded the exercise via two different logins). However, both students must be enrolled to the course (a grade will be attributed to the second member only if he or she is also enrolled).
- You should submit a `.jar` file which contains the `.java` files, with the package directories that contain them, and the README file (no `.class` files).
- **Failure to satisfy the requirements and follow the guidelines above will result in loss of points!**

## 7. Automatic Testing

- As introduced in the intro2cs\csp courses, automatic testing will check each file upon submission.
- Upon exercise submission, you will receive an email that details the results of the automatic testing. This allows you to resubmit as many times as necessary until you pass all automatic tests. Only the last submission counts.
- As noted before, the submission program cuts lines as mentioned above (see section 4 - before submitting). Make sure to check your code and README files for that.
- If you submit a file that fails some of the automatic tests, you will lose a number of points proportional to the number of failed tests and the type of tests you failed. For example, failing many tests due to the same reason (i.e., one bug) will result in losing less points than failing many tests due to different reasons (i.e., many bugs).
- The pre-submission tests **do not include the full test suite which will be used to test and grade your exercises**. Unless stated otherwise, the automatic system will contain **90%** of the full test suite. The remaining tests will be of course used by the course staff when grading your submissions. This is to encourage you to write your own tests in addition to the ones supplied by us.

## 8. Extensions & Late Submissions

- The time of submission of each exercise will be **23:55**. Submitting an exercise later during the night will be treated as a **late submission** and will result in losing points.
- Extension requests should be sent to the course mail. Extensions will be allowed only in the **special cases** of reserve-duty, hospitalization & sickness – all cases will of course **require** the necessary forms (doctor's approval or an official army note describing days of service). Other extreme cases will be up for TAs approval.
- **All** extension requests must be presented **at least 48 hours** prior to the submission time. After that time, all requests will be denied automatically.
- Students that know they will be going on reserve-service should consult the TA (via the course mail) as early as possible about the future exercises. Students coming back from reserve-service should write course mail immediately, to discuss the time of submission for the missed exercises. The same goes for other reasons.
- **Unapproved late submissions will result in losing points in the following manner.**  
**Note that the delay time is rounded UP to a whole number of days.**

**1-day delay – 1 point,**

**2-days delay – 3 points (1+2),**

**3-days delay – 6 points (1+2+3),**

**4-days delay – 9 points (1+2+3+3),**

**5-days delay – 12 points (1+2+3+3+3),**

**6-days delay – 15 points (1+2+3+3+3+3),**

**7-days delay – 18 points (1+2+3+3+3+3+3),**

**AFTER 7 DAYS FROM THE SUBMISSION DEADLINE NO EXERCISE WILL BE ACCEPTED NO MATTER WHAT!**

## 9. Appeals (Ir'urim)

- In order to save both students and staff time on trivial appeals, we automatically assume an appeal on 3 points and accept it - this results in a bonus initially given to every exercise (this bonus will be added last, and won't result in a grade higher than 100). If a more thorough appeal is requested by the student, the appeal is manually reevaluated and **the bonus is lost**.
- Appeals on exercise grades may be submitted **up to one week** after the grades are published. Later appeals will not be handled.
- Appeals will be submitted through a personal forum in Moodle. With each exercise we will open a new appeals forum. Messages on this forum will be visible only by the person who posted the message and the course staff.
- **The title of the appeals submitted in the forum should follow these rules:**
  - (1) **If the reason for the appeal is automatic tests, the title should be the word "AUTO" followed by your user name.**
  - (2) **If the reason for the appeal is manual grading, the title should be the word "MANUAL" followed by the name of your grader, followed by your user name.**
  - (3) **If the reason is both automatic and manual grading, the title should be the words "AUTO MANUAL" followed by the name of your grader, followed by your user name.**

**Approaches in other forms will be erased and ignored.**

- In case of appeals, the submitter(s) may be called to explain their solution.
- The result of the appeal is final.
- Don't send any messages to the graders regarding your appeal.
- **Submitting an appeal means the exercise will be re-evaluated - your grade may also go down.**

## 10. Solve the exercise on your own

- Unless stated otherwise, each student should design and code the exercise by him or herself. Solutions may be discussed with others **verbally**. However, you **may not** under any circumstance exchange program code segments. You **may not** show the code, or part of the code of your solution to other students. You **may not** write it down for them on paper, or give it to them via any media. Copying part or a whole exercise from another student, or obtaining it from any other source, is regarded a **serious** offense, and will result in a severe penalty.
- Furthermore, if we judge that two submitted exercises contain copied code, we cannot know, and will not distinguish, between them, and will regard both as cheating. Note: It is your responsibility to protect your code and files. Claims such as *"my code was stolen"* will not be accepted.
- **We maintain the right to run a cheating detection program at any time, even after grades have been published. This means that even if an exercise of yours was graded, you can be informed at a later time that you are suspected of cheating in this exercise.**