Object Oriented Programming (TC2016). Final project definition

1 PREREQUISITES

Research the basic principles related to the **Container Loading Problem**, until you can answer the following questions:

1. What is the Container Loading Problem?
2. Is this problem worth researching? Report yes or no, and why (try your best to explain this).
3. Does this problem have a definitive (optimal) solution? If so, what is it? If it doesn't, why?
4. Explain the concept of heuristic.
5. Explain the heuristics you can find, which interest you the most, to approach the Container Loading Problem. Explain why you found them interesting (report at least two).

Suggested sources for your research:

1. The Campus Digital Library.
2. Lim, A., & Zhang, X. (2005, March). The container loading problem. In *Proceedings of the 2005 ACM symposium on Applied computing* (pp. 913-917). ACM. https://www.researchgate.net/publication/221002227_The_container_loading_problem
3. Bortfeldt, Andreas, and Gerhard Wäscher. "Constraints in container loading–A state-of-the-art review." *European Journal of Operational Research* 229.1 (2013): 1-20. http://www.fww.ovgu.de/fww_media/femm/femm_2012/2012_07-EGOTEC-503ec3895182dc0d922a6bd7feebb3a5.pdf

2 CONTAINER LOADER

Your team will design and develop an **Object-Oriented Java** container loader simulation, using the concepts discussed in class to program the behavior of a loader for virtual containers which need to be stored inside a cargo ship.

There will be four kinds of containers (which already have some product inside):

- A normal basic container that is used for most shipping.
- A heavy container is a kind of basic container.
- A refrigerated container which is a kind of heavy container.
- Any container may have special properties for containing explosives.
- Any container may have special properties for containing toxics.
- Any container has a serial number, weight, width, height and depth. Tip: research the actual dimensions and weight capacity for standard containers.
- After the initial project is done, you will add a new kind of heavy container that can store liquids.

Each ship has a unique storage capacity. Storage capacity involves the following elements:

- Maximum number of toxic and explosive containers allowed.
- Maximum number of refrigerated containers allowed.
- Maximum number of heavy containers allowed.
- Maximum number of all containers allowed, including basic containers.
- Maximum total weight allowed.

3 REQUIREMENTS

- You will use **Object Oriented** programming in **Java** to develop your solution.
- You will design, develop and submit the full **Class Diagram** for your project.
- Your project **must include at least one** functional and pertinent example of the following to obtain the full score of the coding section:
  - ✓ Classes with **inheritance**. **(5 points)**
  - ✓ **Interfaces** and/or **abstract** classes. **(4 points)**
  - ✓ Class **composition**. **(4 points)**
  - ✓ Class **aggregation**. **(4 points)**
  - ✓ **Arrays of objects**. **(4 points)**
  - ✓ Objects that **contain arrays**. **(4 points)**
  - ✓ **Static** variables and methods. **(5 points)**

Use the following information to infer the proper design of your simulation:

- The application will read an input text file, which stores the name and capacity information of the cargo ship to be loaded. This input text file also stores a list of containers including the serial number of each, and the type and properties of each.
- When run, the program will read the input text file, instance the ship and containers, load the ship and print a manifest into another text file.
- The manifest will include: the ship's name, capacity features, and the list of serial numbers and all information about each container put on the ship. The best answer (***up to 5 extra points for this**: consider that the more cargo a ship can carry will be translated in money for the transport company*) would be the one to maximize the number of loaded heavy, refrigerated, toxic and explosive containers. Finally, the file will list the containers that could not be loaded into the ship.
- Note that we're not assigning a cost for each kind of container. We are only considering the capacity problem. We're also ignoring the legal (local and international) problems limiting the loading. This is a very complex and interesting problem indeed.

4 Allowed Tools

- **Research** from the Digital Library.
- Java command line compiler.
- Plain text editor.
- You are **not** allowed to use any other libraries, IDEs or software.

5 Format and Dates

- You will work alone or in teams of at most three members. Report the team members to your instructor before you begin to work.
- For the **first partial term** (one class before the exam), deliver a **PDF** document in **Blackboard** (only one member of the team will do this) including:
  - Cover with:
    - Institutional logo.
    - Course name and group.
    - Project name.
    - Team members. Full names and student IDs.
    - Submission date.
  - A report of your research about the Container Loading Problem. We will evaluate:
    - What **your team** understood about the Container Loading Problem. Use your own drawings and your own words. Include appropriate research references in MLA or APA format.
    - Be sure to **answer all the questions** presented in the first section.
  - Development plan.
    - How you will schedule the work to be done until the program is finished.

        - **Weekly** responsibilities **for each member** of the team.
        - Include a textual definition, and be **very clear** about your solution strategy (how you intend to solve the problem).
        - Report your expected advance for the second term and commit to achieve it.
    - Use MLA or APA style references.
- For the **second partial term** (one class before the exam) submit to **Blackboard** your Java source code (only `.java` and `.txt` files properly commented. Only one member of the team will do this) in **ONE ZIP** file including the advance estimated in your development plan.
- For the **final submission,** the last day of classes, one member of the team will upload **ONE ZIP** file to **Blackboard** with:
    - Complete source code inside a **code** folder: everything needed to run the program on any java-enabled PC.
    - Complete **Class Diagrams** inside a **diagrams** folder.
    - A video showing:
        - Only the program running.
        - Your team using the software (all members).
        - Total video length: about 2 minutes.
    - The research document for the first submission, now complete with your development experience and **conclusions for each team member**. *A proper conclusion is: reflecting about the project's objectives, and stating if they were achieved or not. A conclusion is also a reflection about the work that could not be completed or possible extensions to the work already made.* Include and explain your diagrams.
    - Be prepared to show your work to the class: include a **presentation** in your ZIP file inside a **presentation** folder.
    - Your presentation **may not begin** if your upload is not ready. Be sure to submit it on time. If your team fails to submit the complete ZIP file with the requested documents on time, your grade will be **discounted 10 points per hour (and/or the corresponding fraction)** until you do, and your team will lose the presentation grade points.

6 Code of Ethics

By being enrolled in a course, all students must recognize that there is an obligation to adhere to the code of ethics presented in this section. A student's actions shall be governed by the following values: respect, responsibility, academic honesty, and a commitment to the community. Also, student's activities within the course must be carried out with honesty and in an individual manner, reflecting his or her knowledge exclusively. Finally, a student must accept the grade assigned by the professor.

7 EVALUATION

Concept	Points
Complete and working source code	**30**
Class diagrams	**20**
Video (screen capture, team using software)	**20**
Presentation (for students attending the session) **	**10**
Research document	**20**
Total	**100***

*According to the Syllabus, the final project represents **20%** of the final course grade.

**Presentation order will be assigned randomly. For large groups (+20 students) we may need to present the project during the last 2 classes. If this is the case, your instructor will notify you with enough anticipation (at least 3 weeks).