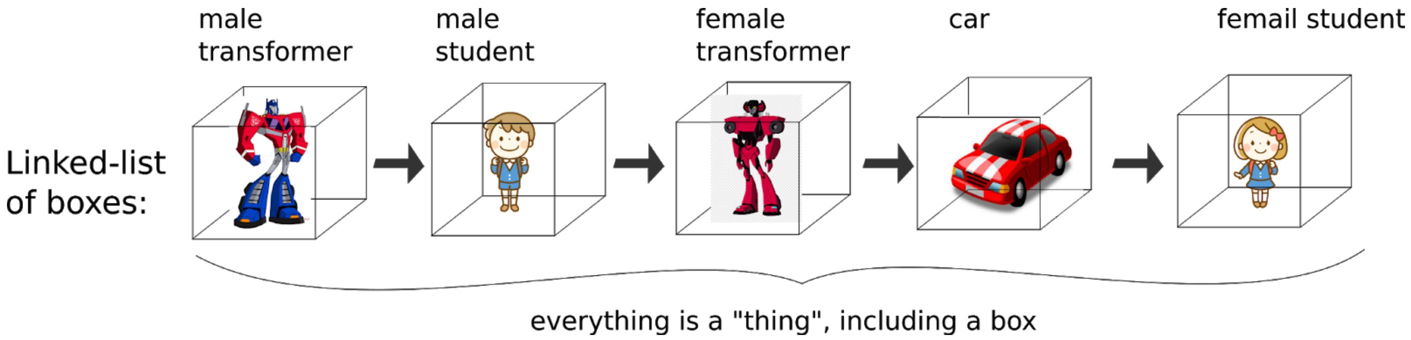
Introduction to Programming II 2021 - Midterm

Deadline: 23:59 on 14/03/2020

For this project, you should implement a linked-list of boxes, which will be sorted, scrambled and then sorted again. Each box contains a “thing” (see the figure below) and a pointer to the next box.



Many things can be a “thing” - a car, a student, a transformer. Each thing has a unique ID. You should write a C++ program that does the following:

1. Generate a random linked-list with at least 5 items of each type shown in the figure above. Remember that:
   - 1.1. A student is a person - a very annoying person sometimes ♥
   - 1.2. A transformer is also a person
   - 1.3. A transformer is also a car
   - 1.4. Transformers and students can be either male or female
2. Sort the boxes according to the “things” within the boxes. You should use Merge-Sort. REMEMBER: this is an **in-place** sorting algorithm.
   - 2.1. Students should be placed before transformers and cars
   - 2.2. Transformers should be placed before cars
   - 2.3. Females should be placed before males
3. Print the linked-list (print the IDs and types of the things)
4. Scramble the linked-list.
   - 4.1. Don’t create a new linked-list for this step.
   - 4.2. Don’t use any external library or anything that requires the installation of third-party software or library
5. Similar to steps (2) and (3) but using Quick-Sort

1. Additional Requirements

a. Your code should contain only UTF-8 characters

b. Your project should be compatible with C++ 20

2. Assignment Requirements

a. Your entire code (to be submitted) should consist of **a single CPP file**

   i. While we recognize this is not a good practice, it will facilitate a more accurate and easier grading of tens of students

   ii. The name of the CPP file should have the form: **yourname_yoursurname.cpp**

b. There should be a comment before each method explaining what it does

3. Grading schema (total of 20 points)

You have to demonstrate knowledge on each concept given to you through the course till this moment. In other words, your code should contain at least one **justifiable** use of each of the following concepts:

a. References and constant types (lecture 1 and 2)
b. Use of the keyword "**delete**" to prevent implicit type conversion (lecture 2)
c. Use of **constexpr** (lecture 2)
d. At least one class with a destructor and several constructors (default, with initializer-list, copy, move) (lecture 3)
e. At least one class with private and public members (lecture 3)
f. Implement operator =, +, and > (lecture 3)
g. User-defined conversion (lecture 3)
h. Multiple Inheritance (lecture 4)
    i. For this, you can ignore the UML Class Diagram
i. Static and dynamic typing (lecture 4)
j. Virtual and pure virtual functions (lecture 4)
k. Accessing the subobject of an objective OR calling the base constructor from a derived constructor (lecture 5)
l. A call to a constant member function (lecture 5)
m. A non-copyable object (lecture 5)
n. A line of code with the form: call to a function (e.g. foo()) = some rvalue or temporary object (lecture 6)
o. At least one abstract class and at least one interface (lecture 5)

In the end, 5 points are also given for the quality and readability of your code as perceived by the instructors. For example, naming conventions, indentation, consistence, comments before functions stating arguments, return and what the function does, quality of the visualization.

IMPORTANT: Add a one-line comment right before the part of the code in which you implemented each of the requested items. This will allow us to quickly search for your answers. Format of the comment:

```
// ITEM X.y: My justification of why I implemented the item here
// some code...
```

Example:

```
// ITEM 3.a: Multiple Inheritance is useful here because this class can... bla bla bla...
```

4. Notes

a. Plagiarism will incur zero as the grade of your midterm, independently of if you are the person who implemented the original code or made the copy.
b. This project (practical midterm) is individual.

5. FAQ

a. What happens to the students who worked on the previous version of the midterm until the deadline?
    i. They should present (in-person or Zoom) what they manage to finish and explain their code
    ii. Such a student will get **up to +10% in the final grade**
    iii. Only the first 20 students will get this bonus