Computer Science 2A

Practical Assignment 01

Assignment date:	2024-02-20
Deadline	2024-02-27 12h00
Marks: 80	

This practical assignment must be uploaded to eve.uj.ac.za **before** 2024-02-27 12h00. Late[1] or incorrect submissions **will not be accepted**, and will therefore not be marked. You are **not allowed to collaborate** with any other student.

Good coding practices include a proper coding convention and a good use of documentation. Marks will be deducted if these are not present. Every submission **must** include a batch file unless stated otherwise.

The **reminder page** includes details for submission. Please ensure that **ALL** submissions follow the guidelines. The reminder page can be found on the last page of this practical.

This practical aims to familiarise you with the differences and similarity of **C++** and **Java**.

Under additional files is a **C++** program (***reverse-cpp.zip***). Convert the **TextStore** and **TestReverse** classes, and create a **Main** class. The ***main*** method, and only the **main** method, must be placed in the **Main** class. Test your Java application against the **C++** version to see if performs the same.

Place the relevant classes into the **acsse.csc2a** package[2].

Hints

The **String** class has the following useful methods:

- ***toCharArray()*** : creates a **char[]** from a **String** instance.
- ***charAt(int index)*** : get a character at a specific index in the **String**.

Create a UML class diagram that contains the relevant classes

Bonus

Submit the bonus project as a separate zipfile with the practical number as P01_B. Failure to do so will result in issues with marking the normal practical and lead to a loss of marks!

Create a batch file for compiling the **C++** (***build-cpp.bat***) in the source folder, making sure the ***EXE*** is in the bin folder and then subsequently run the resulting ***EXE***.

[1] Alternate arrangements for exceptional circumstances will been posted on eve.

[2] Hint: The **Main** class does not need to be in a package

Mark Sheet

1. UML class diagram **[05]**
2. **`TextStore`**
   (a) Constructors (2 marks each) **[06]**
   (b) Instance variables (1 mark each) **[03]**
   (c) Methods (1 mark each) **[04]**
3. **`TextReverse`** - ***`transform`*** method **[05]**
4. **`Main`**
   (a) Input from user. **[02]**
   (b) Reverse text and display. **[02]**
5. Coding convention (structure, layout, OO design). **[05]**
6. Commenting (normal and JavaDoc commenting). **[05]**
7. Correct execution. **[43]**
8. **C++** batch file **[10 (bonus)]**

Total **[80]**

NB

Submissions which **do not compile** will be capped at 40%!

Practical marks are awarded subject to the student's ability to explain the concepts and decisions made in preparing the practical assignment solution. (Inability to explain code = inability to be given marks.)

Execution marks are awarded for a correctly functioning application and not for having related code.

Reminder

Your submission must follow the naming convention below.

`SURNAME_INITIALS_STUDENTNUMBER_SUBJECTCODE_YEAR_PRACTICALNUMBER`

Example

Surname	Berners-Lee	**Module Code**	CSC02A2
Initials	TJ	**Current Year**	2024
Student number	209912345	**Practical number**	P01

`Berners-Lee_TJ_209912345_ CSC02A2_2024_P01`

Your submission must include the following folders:

Folder	State	Purpose
`bin`	*Required*	Should be empty at submission but will contain runnable binaries when your submission is compiled.
`docs`	*Required*	Contains the batch file to compile your solution, UML diagrams, and any additional documentation files. All files must be in **PDF** format. Your details must be included at the top of any **PDF** files submitted. **Do not include generated JavaDoc.**
`src`	*Required*	Contains all relevant source code. Source code must be places in relevant sub-packages! Your details must be included at the top of the source code.
`data`	*Optional*	Contains all data files needed to run your solution.
`lib`	*Optional*	Contains all libraries needed to compile and run your solution.

NB

Every submission **must** include a batch file that contains commands which will:

- Compile your Java application source code.
- Compile the associated application JavaDoc.
- Run the application.

Do not include generated JavaDoc in your submission. All of the classes/methods which were created/updated need to have JavaDoc comments.

Multiple uploads

Note that only **one** submission is marked. If you already have submitted once and want to upload a newer version then submit a newer file with the same name as the uploaded file in order to overwrite it.