Group C Experiment

Experiment Description

In the following questionnaire, you will be asked to read and analyze 2 short methods written in the Java programming language.
Regarding these methods, you will be presented with questions related to understanding the functionality expressed in the method and a maintenance process for it.

At the beginning of this questionnaire, you will be asked some personal questions that will be used solely for the purpose of conducting this study.
At the end of it, you will be asked a few more questions regarding your impressions and opinions about the experience of the just-conducted experiment.
The expected duration for this experiment is approximately 15 minutes. During this time period, you will be presented with various code comprehension and maintenance activities.
The experiment has been structured to make the experience interesting and engaging, while also being clear and understandable.

For the conduct of this questionnaire, it is planned to use materials shared in the drive folder available at the following address:
https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link

Participation in the survey is entirely voluntary.
You have the autonomy to stop answering and exit the survey at any time should you choose to do so.

About data:
Your participation in this study will primarily contribute to scientific research. The results of this research will be accessible through the dissertation or scientific publications resulting from this study.
The researchers gather social and demographic data in a manner that ensures anonymity and does not reveal personal identifying information.
The researchers assure that any personal information obtained through this research will be treated as confidential and will not be shared or disclosed to third parties. This questionnaire utilizes the Google Docs app package, meaning that any information collection and utilization through it are governed by the Google Privacy Policy (https://www.google.co.uk/policies/privacy/).

Thanks in advance for your effort and time!

* Indica una domanda obbligatoria

1. I consent to the use of this data for the purpose of conducting the experiment. *

Seleziona tutte le voci applicabili.

- [ ] I consent

Pre-Experiment Questionnaire

Thank you for choosing to participate in this experiment. Below are the initial demographic questions for the pre-experiment questionnaire.

2. Please provide your PROLIFIC ID *

_____

3. Create and provide a custom ID code for this experiment. *
The code should have the first two characters equal to the respective first letters of your first parent's name and your second parent's name, followed by a two-digit number of your birth month.
The code should appear as follows in the example: AF07
A = first parent's name is Andrea;
F = second parent's name is Fran;
07 = you were born in July.

_____

4. Please indicate your age (not mandatory)

_____

5. Please indicate your highest level of education. *

Contrassegna solo un ovale.

- ◯ High school diploma
- ◯ Bachelor's degree in Computer Science
- ◯ Bachelor's degree in other fields
- ◯ Master's degree in Computer Science
- ◯ Master's degree in other fields.
- ◯ Ph.D. degree in Computer Science
- ◯ Ph.D. degree in other fields
- ◯ Prefer not to say
- ◯ Altro: _____

6. Currently *

Contrassegna solo un ovale.

- I am a student
- I am a working student
- I am employed in a company
- I am self-employed
- I am unemployed
- Prefer not to say
- Altro: _____

7. How would you rate your knowledge of the Java programming language? *

Contrassegna solo un ovale.

	I am not familiar with this language					Excellent
	1	2	3	4	5	

8. How would you rate your skills in code maintenance on Java code? *

Contrassegna solo un ovale.

I have never performed code maintenance on Java code

1
2
3
4
5

Excellent

Comprehension of Method 1

This section of the questionnaire will be dedicated to the analysis and comprehension of the first proposed method. It is possible to view and download the method from the following shared drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link

9. Specify the **start** time in the format hh:mm. *

Esempio: 08:30

10. What is the main functionality expressed in this method? *
(It is possible to view and download the method from the following shared drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link).

```
//Code summary: returns a string representation of the object

public String toEncodedString() {
    //Priority threshold
    StringBuilder buf = new StringBuilder();
    buf.append(getMinPriority());
    //Encode enabled bug categories. Note that these aren't really used for much.
    // They only come in to play when parsed by a version of FindBugs older than 1.1.
    buf.append(FIELD_DELIMITER);
    for (Iterator<String> i = activeBugCategorySet.iterator(); i.hasNext(); ) {
        buf.append(i.next());
        if (i.hasNext()) {
            buf.append(LISTITEM_DELIMITER);
        }
    }
    // Whether to display false warnings
    buf.append(FIELD_DELIMITER);
    buf.append(displayFalseWarnings ? "true" : "false");
    return buf.toString(); //return an encoded string
}
```

Contrassegna solo un ovale.

- ◯ Calculates the sum of priorities, active bugs, and display options.
- ◯ Returns an encoded string containing the minimum priority and active bug categories.
- ◯ Generates a list of bug priorities, active categories, and display options.
- ◯ Adds delimiters to the priority values, bug categories, and display options to create a string.

11. Specify the **end** time in the format hh:mm. *

Esempio: 08:30

12. How confident are you in the answer given to the previous question? *

Contrassegna solo un ovale.

	Not confident at all				Extremely confident
	1	2	3	4	5

13. Please indicate for each of the following items how important it was to understand the functionality expressed by the code. *

Contrassegna solo un ovale per riga.

	Not useful at all	Slightly useful	Neither useful nor useless	Useful	Very useful
Method name	◯	◯	◯	◯	◯
Code summary	◯	◯	◯	◯	◯
Variable names	◯	◯	◯	◯	◯
Usage of variables	◯	◯	◯	◯	◯
Comments within the method	◯	◯	◯	◯	◯
Methods called within the code	◯	◯	◯	◯	◯

14. Which of the following is a vegetable? *

Contrassegna solo un ovale.

- ◯ Eggs
- ◯ Pizza
- ◯ Broccoli
- ◯ Salmon
- ◯ Milk

15. Did you rely on other aspects to understand the code? If yes, please mention them here.

Maintenance of Method 1

This section of the questionnaire will be dedicated to the maintenance process of the previously analyzed method. You will be able to review the resulting codes for the different modifications we will propose. The codes are available and can be downloaded from the following shared drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link

16. Specify the **start** time in the format hh:mm. *

Esempio: 08:30

17. Which of the following modifications would you make to the code? Here follows the method. *
(You can view and download the codes related to the proposed modifications in the following Google Drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link).

```
//Code summary: returns a string representation of the object

public String toEncodedString() {
    //Priority threshold
    StringBuilder buf = new StringBuilder();
    buf.append(getMinPriority());
    //Encode enabled bug categories. Note that these aren't really used for much.
    // They only come in to play when parsed by a version of FindBugs older than 1.1.
    buf.append(FIELD_DELIMITER);
    for (Iterator<String> i = activeBugCategorySet.iterator(); i.hasNext(); ) {
        buf.append(i.next());
        if (i.hasNext()) {
            buf.append(LISTITEM_DELIMITER);
        }
    }
    // Whether to display false warnings
    buf.append(FIELD_DELIMITER);
    buf.append(displayFalseWarnings ? "true" : "false");
    return buf.toString(); //return an encoded string
}
```

Contrassegna solo un ovale.

- ◯ Use constants and descriptive names for the delimiters.
- ◯ Use of FIELD_DELIMITER in the statement buf.append(FIELD_DELIMITER) instead of buf.append(LISTITEM_DELIMITER).
- ◯ Change the instruction buf.append(displayFalseWarnings ? "true" : "false") to buf.append(displayFalseWarnings ? "false" : "true").
- ◯ Use a while loop instead of a for-each loop.

18. Specify the **end** time in the format hh:mm. *

Esempio: 08:30

19. How confident are you in the modification made to the code in the previous question? *

Contrassegna solo un ovale.

	Not confident at all
1	◯
2	◯
3	◯
4	◯
5	◯
	Extremely confident

20. What did you base your decisions on when choosing the changes to make during code maintenance? Rate the importance of each of the following items. *

Contrassegna solo un ovale per riga.

	Not useful at all	Slightly useful	Neither useful nor useless	Useful	Very useful
Method name	◯	◯	◯	◯	◯
Code summary	◯	◯	◯	◯	◯
Variable names	◯	◯	◯	◯	◯
Usage of variables	◯	◯	◯	◯	◯
Comments within the method	◯	◯	◯	◯	◯
Methods called within the code	◯	◯	◯	◯	◯

21. Did you consider any other aspects for code maintenance? If yes, please mention them here.

Comprehension of Method 2

This section of the questionnaire will be dedicated to the analysis and comprehension of the second proposed method. It is possible to view and download the function from the following shared drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link

22. Specify the **start** time in the format hh:mm. *

Esempio: 08:30

23. What is the main functionality expressed in this method? *
(It is possible to view and download the method from the following shared drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link).

```
private boolean heuristic_DeleteLatestTask() {
    // Choosing a task of the last interval
    DRCInterval interval = (DRCInterval)intervalList.getLast().getPrevious().getElement();
    DRCTask task = null;
    int chosenTask = gsc.random.nextInt(interval.tasks.size());
    Iterator it = interval.tasks.iterator();
    while (chosenTask-- ≥ 0) {
        task = (DRCTask) it.next();
    }
    // Deleting the task
    boolean success = deleteTask(task);
    if (Registry.LOG_CHANGES && success) {
        if (operationDescriptions ≠ null) {
            System.out.println("DRC DeleteLatestTask: " +
            operationDescriptions.get(new Long((long) task.operationVar.value())));
        }
        else {
            System.out.println("DRC DeleteLatestTask: " + task);
        }
    }
    return success;
}
```

Contrassegna solo un ovale.

- ◯ The function randomly selects a task from the last interval and replaces it with a new one.
- ◯ The function selects the most difficult task from the last interval and modifies it to make it easier.
- ◯ The function randomly selects a task from the last interval and deletes it.
- ◯ The function selects the easiest task from the last interval and moves it to the end of the task list.

24. Specify the **end** time in the format hh:mm. *

Esempio: 08:30

25. How confident are you in the answer given to the previous question? *

Contrassegna solo un ovale.

	Not confident at all				Extremely confident
	1	2	3	4	5

26. Please indicate for each of the following items how important it was to understand the functionality expressed by the code. *

Contrassegna solo un ovale per riga.

	Not useful at all	Slightly useful	Neither useful nor useless	Useful	Very useful
Method name	◯	◯	◯	◯	◯
Variable names	◯	◯	◯	◯	◯
Usage of variables	◯	◯	◯	◯	◯
Comments within the method	◯	◯	◯	◯	◯
Methods called within the code	◯	◯	◯	◯	◯

27. Did you rely on other aspects to understand the code? If yes, please mention them here.

Maintenance of Method 2

This section of the questionnaire will be dedicated to the maintenance process of the previously analyzed method. You will be able to review the resulting codes for the different modifications we will propose. The codes are available and can be downloaded from the following shared drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link

28. Specify the **start** time in the format hh:mm. *

Esempio: 08:30

29. Which of the following modifications would you make to the code? Here follows the method. *
(You can view and download the codes related to the proposed modifications in the following Google Drive folder: https://drive.google.com/drive/folders/10y1GM-KOOBFj41JvHwUO8k8vmXob1d8w?usp=drive_link).

```
private boolean heuristic_DeleteLatestTask() {
    // Choosing a task of the last interval
    DRCInterval interval = (DRCInterval)intervalList.getLast().getPrevious().getElement();
    DRCTask task = null;
    int chosenTask = gsc.random.nextInt(interval.tasks.size());
    Iterator it = interval.tasks.iterator();
    while (chosenTask-- ⩾ 0) {
        task = (DRCTask) it.next();
    }
    // Deleting the task
    boolean success = deleteTask(task);
    if (Registry.LOG_CHANGES && success) {
        if (operationDescriptions ≠ null) {
            System.out.println("DRC DeleteLatestTask: " +
            operationDescriptions.get(new Long((long) task.operationVar.value())));
        }
        else {
            System.out.println("DRC DeleteLatestTask: " + task);
        }
    }
    return success;
}
```

Contrassegna solo un ovale.

- ◯ To replace the "--" operator with "++" in the while loop.
- ◯ Moving the declaration of the 'task' variable.
- ◯ To use the 'remove()' method instead of 'deleteTask()'.
- ◯ To randomly select a task, you can directly use the `get` method on the `interval.tasks` list.

30. Specify the **end** time in the format hh:mm. *

Esempio: 08:30

31. How confident are you in the modification made to the code in the previous question? *

Contrassegna solo un ovale.

	Not confident at all				Extremely confident
	1	2	3	4	5

32. What did you base your decisions on when choosing the changes to make during code maintenance? Rate the importance of each of the following items. *

Contrassegna solo un ovale per riga.

	Not useful at all	Slightly useful	Neither useful nor useless	Useful	Very useful
Method name	◯	◯	◯	◯	◯
Variable names	◯	◯	◯	◯	◯
Usage of variables	◯	◯	◯	◯	◯
Comments within the method	◯	◯	◯	◯	◯
Methods called within the code	◯	◯	◯	◯	◯

33. Which of the following is not a month? *

Contrassegna solo un ovale.

- ◯ January
- ◯ February
- ◯ Wednesday
- ◯ April
- ◯ May
- ◯ June

34. Did you consider any other aspects for code maintenance? If yes, please mention them here.

Post-Experiment Questionnaire

Thank you again for participating in this experiment. Below are the concluding questions for the post-experiment questionnaire.

35. How would you evaluate the level of complexity encountered during the code comprehension activity? *

Contrassegna solo un ovale.

Very easy	
1	◯
2	◯
3	◯
4	◯
5	◯
Very complex	

36. Please indicate the level of fatigue you experienced during the code comprehension phase. *

Contrassegna solo un ovale.

	Not tiring at all				Very tiring
	1	2	3	4	5

37. How would you evaluate the level of complexity encountered during the code maintenance process? *

Contrassegna solo un ovale.

Very easy

1 ◯

2 ◯

3 ◯

4 ◯

5 ◯

Very complex

38. Please indicate the level of fatigue you experienced during the code maintenance phase. *

Contrassegna solo un ovale.

Not tiring at all

1
2
3
4
5

Very tiring

39. Feel free to share any thoughts or observations you have regarding the code comprehension phase or about the code maintenance process, if you wish.