Universiti Malaysia PAHANG

Engineering • Technology • Creativity

FACULTY OF COMPUTER SYSTEMS & SOFTWARE ENGINEERING

GROUP PROJECT DEVELOPMENT

COURSE	**:**	**Formal Methods**
COURSE CODE	**:**	**BCS2213**
LECTURER	**:**	**Prof. Dr. Vitaliy Mezhuyev**
SUBMISSION DATE	**:**	**15, 16, 17 December 2015**
SESSION/SEMESTER	**:**	**Semester I 2015/2016**
PROGRAMME CODE	**:**	**BCS**

INSTRUCTIONS

1. The **maximum** amount of marks for successful development of the project is **10%**.

2. This mark also includes evaluation of your **group work (5%)**.

3. This document specifies **14 topics** for research, so **14 groups to be formed** (containing 4-5 students). Every group should elaborate own and unique topic (so you need coordinate selection of topics with other students of our class. Hint – you may create FB group).

4. Forming groups is your responsibility (and a part of the project assignment). Please form the groups in order they have equal chances, e.g. containing both strong and not very strong students.

5. The given problems are described the PhD thesis "Combining Lock-Free Programming with Cooperative Multitasking for a Portable Multiprocessor Runtime System" (attached). Please refer to this thesis for explanations and pseudo-code of algorithms, you need develop.

6. The result of the group project are: 1) TLA specification and 2) UPPAAL time automata for the chosen problem (algorithm).

7. TLA and UPPAAL models should contain clear, but not superfluous comments.

8. Project report (one per group) to be given as printed Word doc (containing 7-10 pages). Here you need clear formulate what problem you have modelled and what properties you have checked.

   The Report should include next elements:
   - Title (containing names and IDs of all members of your group).
   - Table of content.
   - Introduction (problem statement, importance of the task, your methodology to solve it).
   - TLA Model and its discussion.
   - UPPAAL model and its discussion.
   - Conclusion (what do you learn from this project, how can you improve the models).
   - References.

   Please also prepare one per group PowerPoint presentation (every person in the group should take a part in the presentation and need emphasize what is his/her role in the project).

9. You will have the Group project defence on our final classes on 15, 16, 17 December 2015. If a task looks complex for you, just develop its simplified model. Please note, your mark will depend not only on the level of elaboration of the models, but on your understanding and answers during QA session.

10. You may contact me with questions by e-mail: mejuev@ukr.net, Facebook or personally in my office in FSKKP building (B-1), 3rd floor. Preferably, if one person from the group will be responsible for discussions.

LIST OF THE TOPICS

Note 1. Description of problems refers to chapters and listings of code in the book "Combining Lock-Free Programming with Cooperative Multitasking for a Portable Multiprocessor Runtime System" (attached). Use it for the case study.

Note 2. Some models, specified as separate topic, may be interlinked with other models, so reading other chapters of the book may be also needed.

1. Modelling Lock-Free Queues (chapter 3.3, listings 3.4 and 3.5).
2. Modelling and solving the ABA Problem (chapter 3.4, listing 3.6).
3. Modelling safe Memory Deallocation (chapter 3.5, listings 3.7, 3.8).
4. Lock-free queue operations with safe node reuse (chapter 3.5, listings 3.9, 3.10).
5. Modelling Lock-Free Scheduling - Task Switches (chapter 4, listings 4.1, 4.2, 4.3).
6. Modelling Lock-Free Scheduling - Basic Scheduling (chapter 4, Listing 4.4, 4.5, 4.6).
7. Modelling Event synchronisation primitive (chapter 4.3.1, listing 4.8).
8. Modelling Mutex synchronisation primitive (chapter 4.3.2, listings 4.9, 4.10).
9. Modelling Monitor synchronisation primitive (chapter 4.3.3, listings 4.11, 4.12, 4.13, 4.14).
10. Modelling Interrupt Handling (chapter 4.5, listing 4.15 and 4.16).
11. Modelling Lock-free memory management (chapter 5.1, listing 5.1 and 5.2).
12. Modelling hazard pointers (chapter 5.1, listing 5.3 and 5.4).
13. Modelling Lock-free heap allocation and deallocation (chapter 5.1, listings 5.5, 5.6, 5.7).
14. Modelling garbage collection (chapter 5.3, listings 5.9, 5.10, 5.11).