LTAT.06.007 Distributed Systems

Course overview

Huber Flores, PhD
ASSOCIATE PROFESSOR

Tartu, Estonia, Delta building (auditorium 1021) 07/02/2022
Course overview

Basic (Lectures 1-5)
  - Computer networking
  - RPC and threads
  - System architectures

Intermediate (Lectures 5-10)
  - Coordination of parallel processing, and mutual exclusion
  - Data consistency models and management
  - Scalability and performance

Advanced (Lectures 10-15) – Invited instructor
  - Fault tolerance
  - Replica management
  - Recovery
Course overview

Fundamental principles
   Concepts and definitions (in lectures)

Practical work
   Exercises (in the seminar) – Zhigang Yin and Mohan Liyanage

Discussion on emerging topics
   Distributed ledger
   Opportunistic networks
Course requirements

Important to know before taking this course!

• Do you need to have basic programming skills?
  Answer: Yes

  LTAT.03.001 Computer programming

  LTAT.06.004 Network Technology I

• Is the course focus on a particular technology?
  Answer: The course will rely on Java and Python to teach fundamental concepts.
Course requirements

Important to know before taking this course!

```java
import java.util.ArrayList;

public class ListComprehension {
    public static void main (String[] args) {
        ArrayList<Integer> list = new ArrayList<Integer>();
        //For (i in Range.between(0, 10)){
            for(int x = 0; x < 10; x++){
                int num = x + 10;
                list.add(num);
            }
        System.out.println(list.toString());
    }
}
```

"/

```java
public class Parser {
    private File file;
    public synchronized void setFile(File f) {
        file = f;
    }
    public synchronized File getFile() {
        return file;
    }
    public String getContent() throws IOException {
        InputStream i = new FileInputStream(file);
        String output = "";
        var data;
        while ((data = i.read()) > 0) {
            output += (char) data;
        }
    }
}
```

This is a data structure (not a network)  This is an input parser
Course schedule

• Lecture meetings
  - Monday 10:15-12:00 in Delta, Narva mnt 18 - 1021

• Weekly exercise sessions
  - Seminar (Group 1): Narva mnt 18 – 2048; Wednesday 10:15-12:00
  - Seminar (Group 2): Narva mnt 18 – 2048; Friday 12:15-14:00

• Exam (tentative) – Two options (you can just take one)
  - Beginning of June (TBD)
  - Moodle version
  - There will be time limit
  - Exams are completed individually
    - No digital talks/chats/etc.

• Resit is just available for those that obtained a negative result (Not for grade improvement)
Course grading (Terminology)

- **Lectures** cover concepts and fundamental aspects of Distributed Systems
  - **Mini-projects** are assigned during lecture times, and are submitted in a specific deadline (a mini-project is completed by a group or individually)

- **Practical seminars** cover programming and written exercises about a topic, e.g., using JMeter.
  - **Exercises** are performed during the seminar. Exercises have to be completed and submitted (at home if seminar time is not enough)
    - Exercises must be completed before the next seminar (upload through the course webpage)
  - **Quizzes** are also part of the practical seminar. Moodle will be used to conduct the quizzes. **You will have 2 attempts to complete a quiz.**
Course grading (Terminology)

• Students (assigned with pseudonymous) collect points during the course. *Ongoing results are updated weekly (see course website)*
  ▪ Mini-projects
  ▪ Exercises + Quizzes

• Bonus points will be available in each mini-projects. These points are *optional*, and will not influence your overall grade. However, you can improve your overall grade with bonus points.
  ▪ Bonus points are given when the result of that extra work is correct. No bonus points are given for attempts.

• Be careful with plagiarism (presenting similar mini-projects/exercises, uncited work)
Course grading (Summary)

• Exam **40pts**
  - You need to score half points to pass (MANDATORY)

• Practical work **60pts**
  - 2 mini-projects (30pts) – Bonus points (if any) are announced within the project description
    - Within each mini-project description are described grading instructions based on implementations of functions and standard answers
  - Quizzes (15pts) / exercises (15pts)

Total = Mini-projects (30) + Practical (Exercises + Quizzes) (30) + Exam (40)
Total = 100 + Bonus (if you have any)
Course grading

Points collected are then converted into a grade

<table>
<thead>
<tr>
<th>Grade</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>91-100</td>
</tr>
<tr>
<td>B</td>
<td>81-90</td>
</tr>
<tr>
<td>C</td>
<td>71-80</td>
</tr>
<tr>
<td>D</td>
<td>61-70</td>
</tr>
<tr>
<td>E</td>
<td>51-60</td>
</tr>
<tr>
<td>F</td>
<td>50 and below</td>
</tr>
</tbody>
</table>
Course book

Free download from https://www.distributed-systems.net/

Other usefull course books:


Discussions

The lecture sessions will be based on discussions. Everybody participates.

• Step 1: Pair discussion for two minutes,
• Step 2: Collect the points together

Tools for online communication:
• Slack (Message board)
• Moodle
• E-mail
Questions?

E-mail: huber.flores@ut.ee

Let’s get into it!