Planning

Hanna Melnyk – s3257819

|  |  |
| --- | --- |
| Goals: |  |
| To learn all of the topics listed in Table 01 " List of Topics to learn & respective materials" table | |
| To be able to apply learned material in coding on intermediate level or higher | |
|  |  |
| Materials: |  |
| For details of learning materials please refer to Table 01 " List of Topics to learn & respective materials" below | |
| The learning materials that are planned to be used are the following: | |
| Main materials: |  |
|  | Lectures |
|  | Book (Introduction to Programming Using Java, David J. Eck) |
|  |  |
| Secondary materials: |  |
|  | YouTube videos |
|  | <https://www.w3schools.com/> |
|  | <https://www.tutorialspoint.com/> |
|  |  |
| Mechanisms of self-regulation: |  |
| Time tracking. Please refer to tables "Activity log" and "Activity plan" below | |

**Table 01: List of Topics to learn & respective materials**

|  |  |
| --- | --- |
| **Topic** | **Material** |
| P.02.01 - Computer Programming Paradigms | Lecture |
| P.02.01 - Computer Programming Paradigms | Book (Introduction to Programming Using Java, David J. Eck) |
| P.02.02 - Object-Oriented Programming (Classes & Objects) | Lecture |
| P.02.02 - Object-Oriented Programming (Classes & Objects) | Book (Introduction to Programming Using Java, David J. Eck) |
| P.02.03 - OOP: Encapsulation and Access Modifiers | Lecture |
| P.02.03 - OOP: Encapsulation and Access Modifiers | Book (Introduction to Programming Using Java, David J. Eck) |
| P.02.04 - Programming by contract, Preconditions, Postconditions and Invariants | Lecture |
| P.02.04 - Programming by contract, Preconditions, Postconditions and Invariants | Book (Introduction to Programming Using Java, David J. Eck) |
| P.02.05 - Testing | Lecture |
| P.02.05 - Testing | Book (Introduction to Programming Using Java, David J. Eck) |
| P.02.06 - Testing with Junit | Lecture |
| P.02.06 - Testing with Junit | Book (Introduction to Programming Using Java, David J. Eck) |
| P.03.01 - OOP: Inheritance | Lecture |
| P.03.01 - OOP: Inheritance | Book (Introduction to Programming Using Java, David J. Eck) |
| P.03.02 - OOP: Polymorphism | Lecture |
| P.03.02 - OOP: Polymorphism | Book (Introduction to Programming Using Java, David J. Eck) |
| P.03.03 - OOP: Abstract class/method, Interface | Lecture |
| P.03.03 - OOP: Abstract class/method, Interface | Book (Introduction to Programming Using Java, David J. Eck) |
| P.03.04 - OOP: Subtyping and dynamic typecasting | Lecture |
| P.03.04 - OOP: Subtyping and dynamic typecasting | Book (Introduction to Programming Using Java, David J. Eck) |
| P.03.05 - Security Engineering Fundamentals (theoretical LO) | Lecture |
| P.03.05 - Security Engineering Fundamentals (theoretical LO) | Book (Introduction to Programming Using Java, David J. Eck) |
| P.03.06 - Security Design (theoretical LO) | Lecture |
| P.04.01 - Arrays & Lists | <https://www.youtube.com/watch?v=h2SWED58mmE> |
| P.04.02 - Collections: Sets | <https://youtu.be/rruCajMgvGA> |
| P.04.03 - Collections: Maps | <https://youtu.be/H62Jfv1DJlU> |

**Table 02: Activity plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **week** | **activity** | **mandatory?** | **time planned, minutes** | **actual time** | **diff** |
| week 02 | lecture | - | 210 | 210 | - |
| week 02 | practical (assignment) | yes | 435 | 435 | - |
| week 02 | self study | - | 120 | 120 | - |
| week 03 | lecture | - | 315 | 315 | - |
| week 03 | practical (assignment) | yes | 555 | 555 | - |
| week 03 | self study | - | 120 | 360 | 240 |
| week 04 | lecture | - | 585 | 585 | - |
| week 04 | practical (assignment) | yes | 1215 | 1,215 | - |
| week 04 | self study | - | 180 | 300 | 120 |
| week 04 | diagnostic test | yes | 120 | 100 | (10) |

**Table 03: Activity log example**

|  |  |  |  |
| --- | --- | --- | --- |
| **date** | **week** | **activity** | **actual time spent, minutes** |
| 11/17/2023 | week 01 | practical | 90 |
| 11/17/2023 | week 01 | self study | 120 |

Conclusions after 4 weeks of programming:

After 4 weeks I have made the following observations:

* The time I assigned for programming self study is not enough. I planned to spend 120 extra minutes apart from self-study sessions scheduled in the timetable. It turned out that I need more time than that.
* Practicals were not as effective for me as expected. Because of the noise in the classroom it’s difficult for me to concentrate. I’ve done less exercises than I planned on practicals
* Learning materials chosen are not enough for understanding of the subject. Apart from going to the lectures and reading a book, I also watched youtube tutorials and read articles related to topics of the week.
* I often forget to add how much time I’ve spent studying in my weekly log.
* I progress faster if I read the related materials before starting to do the weekly exercises.

Overall, I kept up with my planning, and all of the things planned were done in time. For the next weeks I will try to read the materials beforehand in order to have more understanding of weekly topics before doing the exercises. Since the conditions of practicals are not suitable for me, I will try to do the exercises beforehand, and list all of the questions I have to TA’s. I will use practicals mostly to ask questions, not to do the exercises. Time scheduled for practicals could be used to read the weekly materials in a more quiet space. I will try to pay more attention to logging my study time – because I often forget doing it.