Real-time Operating Systems

Course Syllabus

This course contributes to the requirements for the Degree of MSc in Computer Science

Title of the Academic Program	Master's Degree Programs in English "Digital intelligent control systems"
Type of the course	Elective
Course period	3th semester from October, the 1st to February, the 1st (18 weeks)
Study credits	3 ECTS credits
Duration	108 hours
Language of instruction	English
Academic requirements	– BSc degree in Computer Science or equivalent (transcript of records), – good command of English (certificate or another official document) **Prerequisites:** – basic PC user skills

Course Description

" Real-time Operating Systems" is an elective course.

The Real-time Operating Systems program is an educational and methodological document included in the master's educational program, provides for the acquisition of skills and competencies related to the study of the features of embedded operating systems, as well as real-time systems.

Real-time Operating Systems (3 semester) is aimed at the formation of competencies aimed at obtaining theoretical knowledge about embedded operating

systems, and the acquisition of practical skills and competencies in installing, configuring and debugging operating systems.

The discipline is carried out in the computer classrooms of the university.

Special Features of the Course

The course provides an opportunity for students to personally install, configure and work with the embedded operating system. The student will be able to go all the way from choosing an operating system to deploying it using virtualization systems.

In the process of training, the undergraduate will get acquainted with:

• features of application

• architecture of embedded operating systems

Gains competencies and skills in:

• installation of embedded systems

• settings for embedded operating systems

• debugging embedded operating systems

The aim is to study the architecture, the main features of embedded operating systems and the application of this knowledge in the design of applications for various purposes.

Course Objectives

The objectives of the discipline are as follows: to form a solid foundation of knowledge, practical skills sufficient for successful production activities and allowing him to independently master new necessary knowledge and achievements in the field of designing applications running under the control of embedded operating rooms and solving engineering problems in this area.

Learning Outcomes of the Course

As a result of studying this discipline, students should:

• to understand the functionality and selection criteria of various operating systems when designing automation systems for technological complexes in real time;

• to know the structure, basic principles of construction and the scope of use of embedded operating systems.

• to be able to program applied tasks for embedded systems and be able to control the processes occurring in real-time systems

• to have practical skills for solving problems of designing control and monitoring systems for technological complexes in real time based on existing operating systems and programming languages.

Course (module) Structure

Learning Activities	**Hours**
Lectures	18
Practice sessions / Seminars,	18
Self-study Assignments	72
Final Exam (including preparation)	-
Total study hours	**108**

Detailed Shedule

Week	**Lectures**		**Seminars/ Assignments**	**Hours Lec/Lab/HA**
Semester 3				
1.	An introduction to Real-time Operating Systems	Installing and Configuring Linux Operating System on VirtualBox	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
2.	Real-time Operating Systemsbootlo	Studying and comparing the capabilities of operating system	Reading lecture notes. Preparation, implementation and	2/2/8

	ader	loaders	defense of lab work.	
3.	Linux operating system core	Updating and reinstalling the operating system on devices	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
4.	The root file system orgabisation	Creating and converting file system partitions	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
5.	Input-output drivers	Determining and Hardware setup on Linux Operating Systems	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
6.	Linux system initialization	Loading the Linux operating system in various modes	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
7.	Process and thread management in embedded systems	Configuring different policies when scheduling processes and data thread	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
8.	Linux memory management	Memory management in Linux operating systems	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8
9.	Real-time programming	Debugging processes and data thread on Linux operating systems	Reading lecture notes. Preparation, implementation and defense of lab work.	2/2/8

Course Instructor(s) and Tutor(s), Contact information

Dmitry A. Schwetz,
Ph.D. in Mathematical and software support for computers, complexes and computer networks, Lecturer, School of Space and Information Technologies, Siberian Federal University.
(room ULK 3-17) 26, Kirenskogo st, Krasnoyarsk, Russia
e-mail: DSchwetz@sfu-kras.ru
Google Scholar page:
https://scholar.google.com/citations?user=MWXrXhwAAAAJ
Additional information is available at:
http://structure.sfu-kras.ru/node/4244
Tel: +7 908 013 3587

Assessment

Assessment strategy	Points, max	Evaluation criteria
Tests	10	Test questions for lectures in the e-course
Lab works	90	Lab report

Grading policy for final assessment is:

- A (excellent work) 91–100 points
- B (above average) 81–90 points
- C (average) 71–80 points
- D (below average) 50–70 points
- F (failed) < 50 points

Attendance Policy

Students are expected to attend classes regularly. In case of missing an in-lab activity a student should perform additional work submitted to the instructor within a week after a class was missed.

Web page of the course

Course materials and required reading materials are available on the webpage of Embedded Operating Systems, SibFU E-learning portal, www.e.sfu-kras.ru. You must be logged in to access this course: https://e.sfu-kras.ru/course/view.php?id=32943

Core reading

1. Chris Simmonds. Mastering Embedded Linux Programming./ Chris Simmonds. – Packt Publishing Ltd, 2015. – p.384. ISBN-13: 978-1784392536, ISBN-10: 1784392537.
2. Wang, K.C. Embedded and Real-Time Operating Systems./ Wang, K.C – Springer International Publishing. AG, 2017. – p.481. ISBN-13: 978-3319515168, ISBN-10: 3319515160.
3. David E. Simon. An Embedded Software Primer/ David E. Simon. Eddison – Wesley, 2004. – p.359. ISBN-13: 978-0201615692, ISBN-10: 020161569X.

Facilities, Equipment and Software

Virtual Box 6.1