Jordan University of Science and Technology
Faculty of Computer and Information Technology
Department of Computer Engineering
1st Semester, 2007 -2008

Course Information

Course Title	Systems Software Design, Development, and Documentation
Course Number	CPE 471
Prerequisites	CPE 311 (Object-Oriented Software Development) and CPE 351 (Microprocessor Systems and Assembly Language)
Course Website	http://www.just.edu.jo/munzer
Instructor	Monther Aldwairi
Office Location	C5 L1
Office Phone	7201010 (22280)
Office Hours	11 – 12 Sunday, Thursday or by appointment
E-mail	munzer@just.edu.jo
Teaching Assistant	TBA

Course Description

Design and implementation of various system software components including assembler, macro processor, compiler, and loader; analysis of modern software engineering practice for multi-person projects; methods for requirements specification, design, implementation, verification, and maintenance of large software systems; advanced software development techniques and large project management approaches; project planning, scheduling, resource management, accounting, configuration control, and documentation; special emphasis on technical writing. A set of laboratory experiments will provide hands-on experience in related topics.

Text Book

Title	System Software, An Introduction to systems programming
Author(s)	Lelan L. Beck
Publisher	Addison-Wesely
Year	1997
Edition	3rd Edition
Book Website	NA
References	Systems Programming, J.J. Donovan, McGraw-Hill.

Assessment Policy

Assessment Type	Expected Due Date	Weight
First Exam	TBA	10%
Second Exam	TBA	10%
Final Exam	TBA	40%
Projects	Editor (30/09), Simulator (04/11), Assembler (02/12), Linker/Loader(23/12)	30%
Laboratory	Assignments + Quizzes	10%

Course Objectives	Weights
1. To develop a basic knowledge in systems software.	20%
2. To explain the design and implementation of various systems software components.	25%
3. To explain the principles and roles of software engineering, software documentation, and software testing.	15%
4. To apply the above principles in the design and implementation of systems software.	20%
5. To develop an understanding and appreciation of team work.	10%
6. To improve communication skills.	10%

Teaching & Learning Methods

- Class lectures, lecture notes, homework assignments, and project are designed to achieve the course objectives.

- There will be two lectures (1 hour) and one lab session (3 hours) per week.

- The lecture style of the course will be more like project status meetings rather than traditional lectures. I will pace you, lecture on selected topics, lead discussions, etc. The workload for this course is 10 hours per week including classes.

- The course includes three major projects which should be carefully planned and managed.

- You should read the assigned handouts before class, complete assignments and projects on time, participate in class and do whatever it takes for you to grasp this material.

- Ask question and participate in classes: any question you is an important question regardless how you feel about it. Do not be afraid to repeat the question, think that your question is stupid, or think that you are wasting the class time.

- You are responsible for all material covered in the class.

- Please communicate any concerns or issues as soon as practical either in class, by phone or by Email.

- The web page is a primary communication vehicle. Lecture notes will be available before each class. It will contain homework assignments, study guides, and important instructions.

Learning Outcomes: Upon successful completion of this course, students will be able to		
Related Objective(s)		Reference(s)
1	Describe the main features and instruction sets of a test machine	Handouts
1,2	Perform hand assembly for the machine	Handouts
1-6	Design, implement, document, and test a simulator for the theoretical machine	Handouts
1,2	Describe the main features, functions, and types of assemblers	Chapter 2 and Handouts
1-6	Design, implement, document, and test an assembler for the theoretical machine	Chapter 2 and Handouts

1,2	Describe the main features and functions of macro processors	Chapter 4 and Handouts
1,2	Design a macro processor for the machine	Chapter 4 and Handouts
1,2	Describe the main features, functions, and types of loaders/linkers	Chapter 3 and Handouts
1-6	Design, implement, document, and test a loader/linker for the theoretical machine	Chapter 3 and Handouts
3,4	Use software engineering techniques and tools in developing systems software.	Chapter 8 and Handouts
3,4	Develop professional software guide	Handouts
5,6	Function on a multi-disciplinary teams	Handouts
6	Manage meetings and take meeting minutes	Handouts
5	Develop an understanding of professional and ethical responsibility	Handouts

Useful Resources
• Ohio State University (http://www.cis.ohio-state.edu/~stutz-a/)   • http://www.cis.ohio-state.edu/~paolo/teaching/560/

Course Content		
Week	**Topics**	**Chapter in Text (handouts)**
1	• Introduction & Policies *What this course is all about, how we will run this course, how to excel, and how to fail.* • Team work How to be a good team player, guidelines for teamwork, forming the course teams. • Project 0: Editor	Handouts
2-4	• Principles of Software Engineering • Software Documentation • CPE471 machine • Simulator High-level simulation, implementation details and issues • Project 1: Simulator	Chapter 8 and Handouts
Course Content		
5-8	• Assemblers • One-Pass, Two-Pass, and Multi-Pass	Chapter 2 and Handouts

	• Assemblers	
	• One-Pass Assembler: Limitations and Remedy	
	• Two-Pass Assembler	
	Functions of each pass, required data structures, and implementation details & issues	
	• Project 2: Assembler	
8-10	• Macro Processors	
	• Macros and Subroutines	
	• Simple Macro's facility, Macro with arguments, Macro call within Macro, Macro defining a macro	Chapter 4 and Handouts
	• Implementations: Two-Pass, One-Pass, Integrated with pass one of the assembler	
	• Project 2 Bonus (20%): Implementing a simple macro facility	
11-13	• Loaders and Linkers	
	• Functions of Loaders and Linkers	
	• Schemes for Loaders and Linkers: Compile & Go, Absolute Loader, Subroutine Linkage, Relocating Loader, Direct Linking loader, Binding, Dynamic Loader, Dynamic Linker	Chapter 3 and Handouts
	• Implementations: Absolute Loader and Direct Linking Loader	
	• Project 3: Loader and Linker	
14-16	• Compilers Basic Compiler functions	Chapter 5 and Handouts

Additional Notes	
Assignments	• Assignments are due at the beginning of class. • Late assignments are subjected to late penalties. • All assignments are to be done in teams.
Exams	• The format for the exams is generally (but NOT always) as follows: general definitions, Multiple-choice (True/False), Hand-assembly, Writing algorithms, and short essay questions. • To make sure you pass the exams, you should be a good team player. • Grades will not be given out via e-mail
Makeup Exams	• Makeup exam should not be given unless there is a valid excuse. • Arrangements to take an exam at a time different than the one scheduled MUST be made prior to the scheduled exam time.
Drop Date	Last day to drop the course is before the twelfth week of the current semester.
Cheating	• Cheating, copying programs from other teams, writing code with another student, copying code from another student, giving code to another student, posting code in a publicly accessible location, or destroying files or sessions of others will not be tolerated. • You may discuss the assignment with anyone and use any reference material, provided you do not copy any other person's work. Appropriate reference or credit must be acknowledged if you do not solve the homework problems on your own. • JUST Academic dishonesty policy will be applied.

Attendance	- Excellent attendance is expected. - JUST policy requires the faculty member to assign ZERO grade (35) if a student misses 10% of the classes that are not excused. - If you miss class, it is your responsibility to find out about any announcements or assignments you may have missed.
Workload	You are expected to spend 10 hours/week including class time. Please plan your time carefully.
Graded Exams	Graded exam papers will be returned to students one week after the exam date.
Participation	- Participation in, and contribution to class discussions will affect your final grade positively. Raise your hand if you have any question. - Making any kind of disruption or side talks in the class will affect you negatively.
Laboratory	- There will be a 3-hour lab per week. - Missing two labs without excuse or four labs with excuse will render your grade 35. - TA is going to give lab policy and assignments.
Projects	- All projects will be done in teams. A team consists of three to four students. - Students are expected to have a good background in programming and how to use assemblers, macro processors, compilers, and loaders. This course is devoted for studying how these components are implemented. - You are expected to design, test, and document your project. - All of your work will be a part of teamwork. - You have to use an OOP language. - Only one lab report per group needs to be turned in for grading. - If your lab has known errors, you are responsible for reporting them when the lab is turned on. - Your program must be generating results in order to be graded. Labs that fail to generate output will be assigned a grade of zero. - More details about phases and topics of the projects will be given later on.
Late Penalties	Plan our schedule carefully. It is definitely worth you while to avoid turning in any of your lab assignment late. The penalties are extremely severe.

1 to n full days early	5% bonus up to 15% of lab total
1 day late	10% off the score received on the lab
2 days late	20% off the score received on the lab
3 days late	40% off the score received on the lab
4 or more	FORGET IT !!!!!!!!!!!!!!!!!!!!!!!!!!!

Finally	**Make backups of all of your work** including any assignment and project materials you and your group produce. - Share and distribute your work to your teammates.