Syllabus

Advanced Computer Architecture

Course Name	Course type (credit/hours)	전선(3/3)	Course code	
	Target students Division/major/grade	컴퓨터공학과/	Opening semester	2019년 2학기
	Class time and classroom	()		
Reference to this course	Related basic courses	학부 컴퓨터구조, 학부 운영체제		
	Recommended concurrent courses	운영체제		
	Related advanced courses			

Instructor	Name (title/division)	안정섭 (조교수 / 소프트웨어학과)				
	Office Room Number	Paldal #1004-1	Office phone Number	3823	e-mail	jsahn@ajou.ac.kr
	Office hours	by appointment	Homepage address	http://jeongseob.github.io		
Teaching Assistant	Name (title/division)					
	Office Room Number		Office phone Number		e-mail	

1. Introduction

Computer architecture is a fast-evovling area with interesting new techniques added in every generation of processors. Recently, the area is a facing a new phase of evolution with billions of transistors on a chip and multicores techniques. The goal of this course is to learn important concept in computer architecture. This course will cover various aspects of high-performance microprocessors, which include pipelining, front-end desgin, out-of-order execution, and caches. As multicore technologies have been used in all levels of computing from laptops to supercomputers, the course will cover topics in traditional multiprocessors and recent developments of multicores technologies.

** Note: I will assume that students already took the undergraduate computer architecture and operating system course. **

2. Course Objectives

The goal of this course is to learn essential concepts in computer architecutre. Students can understand how their application codes are executed in a computer processor and how processors have been optimizing the applications.

After this course, students may apply the knolwedge to design or optimize their applications to improve the performance of their applications. In addition to that, students can introduce architectural techniques for future computer systems.

3. Class types and activities

4. Teaching Method

The major part of the class will be lectures, programming assignments, and discussions. For the assignments, I encourage two people to organize a group and do programming projects and discussions. The short quizzes may be given during course work.

5. Knowledge and ability required for taking this course

To sucessfully complete this course, students must have knowledge about the basic computer architecture(SCE212) and operating systems(SCE213). In addition to that, students need to be able to write C program and read assembly codes.

6. Method of Evaluation

Evaluation Item	The Number of Times	Evaluation Proportion	Remarks
Attendance			
midterm exam	1	25	The ratio can be changed according to the portion of programming assignments
final exam	1	25	The ratio can be changed according to the portion of programming assignments
quiz			
presentation	1	10	
discussion			
homework	3	40	
etc			

7. Textbooks

Main/Sub	Title	Writer	Publisher	Publication year
주교재	Computer Architecture : A Quantitative Approach	Hennessy and Patterson	Morgan Kaufmann	2012

8. Lecture Schedule

Week	Lecture contents	Lesson type	Remark
1	Course Overview	lecture	
2	Cache and Memory Hierachies	lecture, discussion	
3	ISA Design and Pipelining I	lecture	
4	Pipelining II	lecture	
5	Instruction-level Parallelism (ILP) I	lecture	
6	ILP II	lecture	
7	ILP III and Case Studies	lecture, discussion	
8	mid-term exam	mid-term exam	
9	Multiprocessors I: Overview and Consistency	lecture	
10	Multiprocessors II : Coherence	lecture	
11	Multiprocessors III: SMT and Multicores	lecture, discussion	
12	MP Case Studies	lecture	
13	Advanced Memory Hierarchies	lecture	
14	Transactional Memory	discussion	
15	Other issues and Course Review	discussion	
16	final term exam	final term exam	

9. Others

The scope of topics and schedule of lectures may be adjusted and/or changed after testing students'background knowledge and interests.