

## SYLLABUS

**1. Professor :** Haldun Hadimioglu

Office : 10.009 2MTC

Tel : (718) 260-3101  
Fax : (718) 260-3609haldun at photon dot poly dot edu  
<http://cis.poly.edu/haldun>**2. Course format :**

Lectures : 2.5 hours/week

Lecture Section : 1277

**3. Prerequisite :** CS 6133 Computer Architecture I.

⇒ Students who took CS 2214 at NYU-Poly can take the course

**4. Course web page :** <http://cis.poly.edu/cs614>

⇒ Course handout files are at the course web site

**5. CS6143 is a course on advanced pipelining and parallel processing !****6. Textbooks :****a)** *Computer Architecture : A Quantitative Approach*, 4<sup>th</sup> edition, J. L. Hennessy and D. A. Patterson, Morgan Kaufmann Publishers Inc., 2007.⇒ Publisher's web site, <http://books.elsevier.com> and the CD that comes with the book have a large amount of material. Students are strongly suggested that they study the web site material and the CD and print the appendices that will be used especially during the exams.

➤ Students are reminded about printing the appendices !

**b)** *Fundamentals of Parallel Processing*, Harry F. Jordan & Gita Alaghband, Prentice-Hall, 2003.**In addition :** Students will also read other books and papers, especially on multi-core, parallel random access machine (PRAM) and single-instruction stream, multiple-data stream (SIMD) machines.**7. Professor's message :**→ **Target learning and thinking, not the grade nor pattern matching (problem solutions) !**⇒ Do **not** go for memorizing patterns (problem solutions)→ Knowledge is **not finite nor independent pieces of patterns**→ Knowledge is a **hierarchy** of unmeasurable width and depth of interconnected entities→ Knowledge generates knowledge : Knowledge helps us **think** which leads us to discover new knowledge→ Knowledge also helps us solve **problems** by means of **thinking**⇒ Learning means one acquires knowledge which helps for both **thinking and solving problems**→ Lectures are **interdependent** and courses are **interdependent**

## 8. Benefitting from the course for the years before and after graduation :

→ Do not go for *passing the course*, but *learning*

- Industry and academia look for graduates who can **solve problems**, are **systems oriented** and **creative**
  - ⇒ In order to accomplish those you need to have
    - Analytical skills, synthesis skills and team work skills
  - ⇒ For these three skills, you will need to develop **critical thinking** and **personal skills** for which you need to do the following
    - **Learning is your target !**
    - If you study **past exams**, without studying the book and others, you do **pattern matching**, not thinking !
      - ⇒ You need to study **books, notes, handouts !**
    - If you **miss classes** you are not learning enough !
  - ⇒ **You interact with people to work on a problem, to exchange ideas, to help, etc. !**
- All of these are possible by doing the following
  - ⇒ You **care** for the course and show with your own actions.
  - ⇒ You take the course to **learn**, not to pass !
  - ⇒ You take the course for its **content**, not for the grade !
  - ⇒ You **attend** classes and **do** the homework !
- Then, you will enjoy the course and appreciate it which will make you more interested in the course !
- Students understand and satisfy the **goal** of the course :
  - ⇒ Acquiring skills to be **systems oriented** and a **problem solver** as well as acquiring the necessary course content which is **parallelism** :
    - CS6143 introduces **parallelism** techniques to improve computer performance and capacity :
      - ⇒ **Advanced pipelining** techniques to improve the uniprocessor system performance by exploiting instruction-level parallelism (ILP) and loop-level parallelism of application programs.
      - ⇒ **Parallel processing** techniques to improve the performance and capacity, by utilizing multiple processors or processing elements to exploit loop-level and higher-levels of parallelism : thread-level, task-level and process-level parallelisms. The focus will be on PRAM, SIMD (single-instruction stream, multiple-data stream) and MIMD (multiple-instruction stream, multiple-data stream, multi-core) systems.

## 9. Homework :

There will be six homework assignments. **An assignment submitted late will not be accepted.** Students are reminded about studying the solutions provided by the textbook.

- ⇒ Students will form teams by the *third* week of the semester. The homework will be submitted by teams.
- ⇒ *Students who do homework are faster at solving problems.* **Showing work** (intermediate steps) is **required** to get full/partial credits on a question. Although, the homework will **not** affect the term grade, it can help raise grades as explained below.
- ⇒ Homework assignments have **modified** past exam questions and answers to help learn chapters and solve homework problems. Students need to study them **before they solve homework problems**, not before exams. Note that, these past exam questions are samples and do **not** give hints about exams this semester.

## 10. Exams :

There will be a 150-minute midterm exam and a 150-minute final exam.

- ⇒ **Showing work** (showing intermediate steps) is required to get full credits on a question. That is, both the final answer and the steps to get it, the **approach**, are important.
  - These steps are given in class and past exam solutions. Therefore, students are expected to solve exam questions as such. Showing the approach also helps students acquire and improve their **documentation** skills, critical for the technical world.

⇒ In order to facilitate this, the exams are **open book** exams : students can use their own material, i.e. their books, notebooks and handouts during the exams. Note that once the exam starts there is **no** sharing.

⇒ **Students must prepare for the exams as if they are closed book exams !**

## 11. Term Grade :

The term grade is based on the weights of the exams :

**40% Midterm Exam**

**60% Final Exam**

⇒ The homework does **not** affect the term grade directly but it is taken into account when a student's term grade is near a grade "border." Also, taken into account is student's attendance record. If they are good, the grade is raised. Finally, the professor may change the term grade computation. Thus, students are strongly suggested that they fulfill the requirements of the course, i.e. lectures and homework assignments.

## 12. Office Hours :

The professor has an **open-door** policy that if he is not busy, students can ask questions in his office. If the door is closed, he might be teaching or at a meeting. If a student wants to see the professor at a certain time, he/she makes an appointment with the professor.

⇒ Students can use email. But, they are strongly suggested that they **see** the professor to ask questions, instead of sending email. If email is sent, a Polytechnic email address must be used and student's name and section must be included. Broadcast messages will be sent to class to make announcements. Note that grades are **not** given out to students via email or telephone. Students need to see the professor to learn their grades.

## 13. The Theme of the Course :

i) CS6133 introduces a uniprocessor computer with an integer pipelined CPU and a hierarchical memory of caches, physical and virtual memories.

ii) CS6143 explores designs for higher computer performance and higher capacity through a more rigorous exploitation of parallelism. In CS6143, parallelism on several layers of computers is targeted. The layers we will target are the computational method, algorithm, high-level language, architecture and microarchitecture.

iii) The first half of the semester is on low-level parallelism, including the instruction-level parallelism, **ILP**, and loop-level parallelism, that are on the architecture and microarchitecture layers. These two forms of parallelisms have been extensively exploited in the form of dynamic pipelining, superscalar execution, vector processing and VLIW (EPIC used by the Intel Itanium is a version of VLIW).

iv) The second half of the semester is on parallel processing, especially massively parallel processing, i.e. SIMD and MIMD computers. We will start with interconnection networks used in SIMD and MIMD computers. Then, we will cover PRAMs which are theoretical systems to help understand real parallel systems better. We will then discuss SIMD machines. We will continue with shared-memory MIMD systems and the related topic of cache coherency (current multi-core systems are shared-memory MIMD systems). This is followed by distributed memory MIMD systems. We will discuss their properties and why today's fastest supercomputers are distributed memory MIMD systems. Finally, we will sum up the semester by studying current conference and workshop papers that outline issues for now and future for parallel processing systems.

**14. Material Coverage :** Chapters from Hennessy and Jordan books will be covered. Also, students will study other books and papers. They will be also given additional material in class. The **tentative** schedule is as follows :

Day(s)	Subject
Jan 26	Introduction ; Computer layers ; Computational methods ; Popular scientific applications ; MIPS FP instructions ; CS613 Integer pipeline ; (Chp 1 ; Appendix A, B) ; (Chp 1)
Feb 2, 9, 16, 23	MIPS FP pipeline ; Advanced pipelining : Speculative execution, superscalar execution, VLIW computation ; (Chp 2, 3, Appendix A, B, G) ; (Chp 7)
Mar 1	Vector processors ; (Appendix F)
Mar 29	Midterm Exam : Lectures on Jan 26 - Mar 1 ; HW : 1 - 3
Mar 1, 8, 22	Flynn's classification ; Interconnection networks ; (Chp 2, 3, 4 ; Appendix E, H) ; (Chp 1, 4, 6)
Mar 22, Apr 5	Parallel processing overview ; Dependency analysis ; Levels of parallelism ; Parallelism issues ; Algorithm complexity ; PRAMs ; (Chp 2, 3, 4 ; Appendix H) ; (Chp 2, 4, 7)
Apr 12, 19	MIMD machine overview ; Shared-memory MIMD systems ; Cache coherency ; Shared Memory MIMD systems ; Synchronization ; Programming ; (Chp 2, 4, Appendix H) ; (Chp 1, 3, 4, 5, 7, 8, 9, 10)
Apr 26	Distributed Memory MIMD systems ; SIMD machines ;
May 3	Current parallel processing system issues ; (Chp 2, 4, Appendix H) ; (Chp 2, 7, 9, 10)
May 10	Final exam : All lectures : Jan 26 - May 3 ; HW : 1 - 6

### 15. References :

Students are suggested that they study **recent** computer architecture and parallel processing books since the field advances fast. **Using the web to gather information is strongly discouraged !**

The following references are recommended with respect to their relevance to the course and the textbooks :

**a)** *MIPS RISC Architecture*, G. Kane and J. Heinrich, Prentice-Hall, 1992.

**b)** *Advanced Computer Architectures*, Sajjan G. Shiva, CRC Press, 2006.

**c)** *Modern Processor Design : Fundamentals of Superscalar Processors*, John P. Shen and Mikko H. Lipasti, McGraw-Hill, 2005.

**d)** *Speculative Execution in High Performance Computer Architectures*, D. A. Kaeli and Pen-Chung Yew, Editors, CRC Press, 2005.

**e)** *CPU Design : Answers to Frequently Asked Questions*, Chandra M.R. Thimmannagari, Springer Verlag, 2005.

**f)** *Programming Many-Core Chips*, Andras Vajda, Springer Verlag, 2011.

- g) *Computer Architecture : From Microprocessors to Supercomputers*, Behrooz Parhami, Oxford University Press, 2005.
- h) *Advanced Computer Architecture and Parallel Processing*, H. El-Rewini and M. A. El-Barr, John Wiley, 2005.
- i) *Interconnection Networks : An Engineering Approach*, J. Duato, S. Yalamanchili and L. L., Morgan Kaufman, 2003.
- j) *The Art of Parallel Programming*, Bruce Lester, 2/e, 1<sup>st</sup> World Publishing, 2006.
- k) *Sourcebook of Parallel Computing*, Jack Dongarra, et.al. Editors, Morgan Kaufman, 2003.
- l) *Parallel Programming : Techniques and Applications Using Networked Workstations and Parallel Computers*, B. Wilkinson and M. Allen, 2/e, Prentice Hall, 2005.
- m) *Highly Parallel Computing*, G. S. Almasi and A. Gottlieb, 2/e, Benjamin/Cummings, 1994.
- n) *Programming Massively Parallel Processors : A Hands-on Approach*, David B. Kirk and Wen-mei W. Hwu, Morgan Kauffman, 2010.
- o) *The Art of Multiprocessor Programming*, M. Herlihy and N. Shavit, Morgan Kauffman, 2008.
- 16) *Computer Architecture : Pipelined and Parallel Processor Design*, M. J. Flynn, Jones and Bartlett, p.
- q) *Parallel Computing : Theory and Practice*, Michael J. Quinn, McGraw-Hill, 1994.
- r) *Principles of Parallel Programming*, C. Lin and L. Snyder, Addison-Wesley, 2009.
- s) Numerous research papers from journals and conference and workshop proceedings on computer architecture, parallel processing, supercomputing, operating systems, algorithms, programming languages and compilers.

## 16. Reminders about the course:

Students are required to read the web pages whose links are provided with at the course web site :

1) **NYU-Poly Code of Conduct web page : <http://www.poly.edu/academics/code-of-conduct>**

2) **NYU-Poly Life page web page with links to Health+Wellness, Campus Safety, Students Resources and other.** In addition, keep the following in mind :

a) **A successful course experience** : To enjoy the course as much as possible and **be ready for the follow up courses**, students need to be **committed** to the course

- ⇒ **Attending classes** and **doing the work** are needed.
  - Lectures are **dependent** on each other.

- ⇒ **Study** the notes, books, handouts. But, just reading does not mean studying ! Also, do the homework !

- **Do not study past exams, without studying books, notes and handouts.**

b) Students must realize that every action they take has consequences. They also should **not** make assumptions and decisions on the course (the exams, lectures, the homework and attendance) without asking the professor. An assumption that is often made by students is that every course is the same. This is not the case !

c) Students are asked that the professor does his assigned job : **Teaching** ! Any time the professor is not doing it

means students are not benefitting from the course. Examples of when the professor is **not** doing teaching include discussions involving submitting late homework, late attendance, missing exams, etc.

**d)** Students are asked that the professor does his assigned job : **Teaching !** Any time the professor is not doing it means students are not benefitting from the course. Examples of when the professor is **not** doing teaching include discussions involving submitting late homework, late attendance, missing exams, etc.

**e)** A reason for a low grade is **missing classes**. Even if one gets the notes, it does **not** help. This is because, first, the notes taken from the board may not be correct. Second, someone taking the notes may not write down all the verbal comments and suggestions made by the professor. Third, attending classes forms better memory because of visual (seeing the writing on the board), audio (listening to the professor) and tactile (writing down the notes) inputs. In addition, during lectures the professor refers to earlier lectures (past topics, comments, suggestions, etc.) which refreshes students' memory and further reinforces their knowledge. Overall, students learn and remember more. Finally, since their memory is fresh, students save time when they study for exams.

**f)** Missing an exam is **not** a minor case. A careful assessment is made to excuse a student or to grant an incomplete to a student. The professor makes the decision. The decision is made also based on the information by the student's academic department and the Student Affairs Office. One of the requirements to excuse a student is that at the time the student is not able to take the exam, he/she **be in good standing in class**, i.e. has good attendance, a good homework performance and a good exam performance : The professor wants to see that the student has been committed to the course and learning the material has been his/her main objective.

A student who is excused from a midterm exam is **not** given a make-up exam. The weight of the midterm exam is distributed to the other exams at the discretion of the professor. The make-up exam for the final exam will be harder than the one given to the whole class.

If a student experiences **any** problem, including health/personal problems, he/she must immediately contact Judith Simonsen who is the director of the Student Development Office : jsimonse@poly.edu. Her number is (718) 260-3197.

**g)** For a course, the semester is over when the final exam is over. Students are **not** given extra work, a project, a make-up exam or any other kind of special treatment to raise their grade during or after the semester.

**h)** It has been observed that a student pays unnecessary penalty, because he/she does not know/follow NYU-Poly and CS2204 rules and regulations. They also do not seek advice from Polytechnic staff. Therefore, students, especially, **transfer students**, are strongly suggested that **they speak with the professor**, the TAs, the major advisor, the personnel of the Student Affairs Office, and the Counseling Center.

**h)** Students are advised that they be cautious about web sites used when they gather information related to the course. Many web sites contain erroneous material and are not reliable. Students can contact the professor for web site links. They can also access web sites of academic people who specialize in the area of CS6143.

**i) Always keep contact with the professor. Discuss personal matters in professor's office.**