Computer Science 2506
Computer Organization II

MIPS32 Pipeline
### Instructors

<table>
<thead>
<tr>
<th>William D McQuain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email:</td>
</tr>
<tr>
<td>Office:</td>
</tr>
<tr>
<td>Office Hours:</td>
</tr>
</tbody>
</table>

©2005-2013 McQuain
Instruction formats and construction, addressing modes, instruction execution, memory hierarchy operation and performance, pipelining, input/output, and the relationships between high level programming languages and machine language.

The course will focus on the MIPS32 architecture.
Course Objectives

Having successfully completed this course, the student will be able to:

- Design and analyze instruction sets and their impact on processor design,
- Identify and analyze the design and function of the basic instruction execution elements of a modern processor,
- Apply finite state automata to computer design,
- Describe the basic elements of computer architecture and their impact on the performance of a modern processor,
- Explain the design and function of each element in a memory hierarchy,
- Identify and explain the different methods of I/O in a computer system,
- Explain the relationship between the computer hardware, the operating system that runs on it, and the applications that are compiled to it,
- Compare computer architectures and organizations based on quantifiable performance metrics.
## Course TAs

<table>
<thead>
<tr>
<th>Name</th>
<th>Email</th>
<th>Office Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mariam Umar</td>
<td>mariam.umar AT vt DOT edu</td>
<td>Office Hours on course website</td>
</tr>
<tr>
<td>Akshay Ravichandran</td>
<td>akshay87 AT vt DOT edu</td>
<td>Office Hours on course website</td>
</tr>
<tr>
<td>William Preston</td>
<td>will96 AT vt DOT edu</td>
<td>Office Hours on course website</td>
</tr>
</tbody>
</table>
### Pre- and Co-requisites

<table>
<thead>
<tr>
<th>Course</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 2114</td>
<td>Software Design and Data Structures</td>
</tr>
<tr>
<td>CS 2505</td>
<td>Computer Organization I</td>
</tr>
<tr>
<td>or</td>
<td></td>
</tr>
<tr>
<td>ECE 2504</td>
<td></td>
</tr>
</tbody>
</table>

These prerequisites must have been completed with a grade of C or higher (C- is not acceptable) or AP credit.

<table>
<thead>
<tr>
<th>Course</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math 2534</td>
<td>Discrete Mathematics</td>
</tr>
<tr>
<td>or</td>
<td></td>
</tr>
<tr>
<td>Math 3034</td>
<td>Intro to Proofs</td>
</tr>
</tbody>
</table>

I will not grant any exceptions to the stated prerequisites, including the minimum grade requirements.
Textbooks

Required:

David A Patterson and John L Hennessy
Morgan-Kaufmann ©2012
ISBN 978-0-12-374750-1

Recommended:

C Programming a Modern Approach, 2nd Edition
K. N. King
WW Norton ©2008
978-0-393-97950-3
Recommended References

(available ONLY at the course website)

Available via the Safari Database in the VT Library:

The Art of Debugging with GDB, DDD, and Eclipse

Write Great Code, Volume 1

Write Great Code, Volume 2

C Programming for Scientists and Engineers with Applications
Final grades will be based on the average achieved over the following:

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
<th>Dates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>60%</td>
<td>See course website</td>
</tr>
<tr>
<td>Tests</td>
<td>10% each</td>
<td>Tentatively, Feb 28 and April 18</td>
</tr>
<tr>
<td>Final Exam*</td>
<td>20%</td>
<td>7:45 – 9:45 May 14</td>
</tr>
</tbody>
</table>

**Grade Scale**

The usual 10-point scale will apply (subject to any curve). A final average of 90% will guarantee an A-, 80% will guarantee a B-, and so forth.

**Curve**

A grade curve may or may not be employed in this course. The application of a curve is dependent upon class performance on tests, projects and homework. The decision to utilize a curve rests entirely with the course instructors.

* Exam score will replace the lower test score, if it is higher.
In order to receive a grade of C or higher for the course, you are required to meet some minimum requirements:

**Buffer bomb**  You must trigger at least the first three phases (0-2)

Understand something clearly: if you fail to meet the stated requirements, you will receive a grade of C- or lower for the course, regardless of your overall average.
Test Environments

- When relevant, a test environment, such as MARS or Logisim will be specified for homework assignments.
- The C-language assignments will be compiled on Linux with gcc 4.5.x or later.
- Solutions will only be tested under the specified environment.
- It is the YOUR responsibility to ensure that YOUR solutions execute correctly in the appropriate environment; solutions that do not will receive substantial deductions.
Late Work

Due dates

Each programming project and homework assignment will have a due date and time and will include instructions for submission.

Homework

Usually, no late submissions will be allowed for homework assignments.

Programming Assignments

Except in the very rare case that an extension is granted, late submissions will incur a penalty per diem late penalty that will be included in the project specification. This is typically 10%.

Extensions

Any request for an extension must be made, preferably by email, at least 24 hours prior to the due date.

Late submissions will not be given any credit if submitted after graded assignments or solutions have been released.

Statute of Limitations

Any questions about the grading of an assignment must be raised with your instructor within one week after the graded assignment has been made available to you.
Sources of Help for This Course

General Issues
- CS 2506 classmates
- CS 2506 Forum board (forum.cs.vt.edu)
- CS 2506 TAs
- CS 2506 Instructor

Programming Language Help
- P&H text and other resources from the course website
- C language references
- CS 2506 Forum board

Lecture Instruction
Lectures will consist of presentations, applications, problems and solutions interspersed with classroom discussion.
Backups

Students are responsible for making backup copies of all their work in this (and all) courses.

Loss of work due to hard drive failure is NOT an acceptable excuse. Backup copies of files on the same hard drive are not backup copies. Backup copies of files on second hard drives are also risky. Backup copies should be maintained on two separate distinct storage mediums, (e.g., hard drives and Zip disks).

Backup copies should be maintained until after the end of the term and students have received their course grade. (The Army lives by triplicate for a reason.)

Remember: Computer systems are mechanical devices.

   Network connections break.
   Plan for it. It is inevitable!
Honor Code

An exhaustive list of Honor Code violations would be impossible to present here, but among other things, each of the following is a flagrant violation of the Virginia Tech Honor Code, and violations will be dealt with severely (Honor Court):

• Working with another student to derive a common program or solution to a problem. Unless explicitly stated otherwise, there are no group projects in this course.

• Discussing the details required to solve an assignment. You may not share solutions, or collaborate in the creation of a solution.

• Copying source code (programs) in whole or in part from someone else.

• Copying files from another student's disk or lab account even though they might be unprotected.

• Editing (computer generated) output to achieve apparently correct results.

It is acceptable to discuss an assignment with classmates in a general way, i.e., to discuss the nature of the assignment. In other words, you may discuss with your classmates what your solution is required to accomplish but not how to achieve that goal using C, MIPS32 assembly, or other relevant tools. In no way should the individual statements of a program or the steps leading to the solution of the problem be discussed with or shown to anyone except those people cited in the following statement.
Honor Code

Feel free to discuss the homework assignments and your program source code with the teaching assistants assigned to CS 2506, the instructor, or the free tutors provided by UPE. The discussion of your program source code must be limited to these people. Note that this specifically excludes discussions of your program source code with other students (even if they are not enrolled in CS 2506), or with tutors except for those named above. Privately hired tutors are not an exception to this requirement, nor are athletic or other tutors provided by the University.

Copies of all submitted work are retained indefinitely by the Department. Submitted programs are subjected to automated analysis for detection of cheating.

If you have any question as to how the Honor Code applies to this class, remember that:

- Any work done in this class must be done on an individual basis.
- Credit will be given only for work done entirely on an individual basis.
- Do not make any assumptions as to who can provide help on a programming assignment.
- All submitted work is archived. All submitted programs will be subjected to automated cheat analysis.

Evidence indicating the violation of the policies stated above will be submitted to the Honor Court. It is much easier to explain a poor grade to parents or a potential employer than to explain an Honor Court conviction.