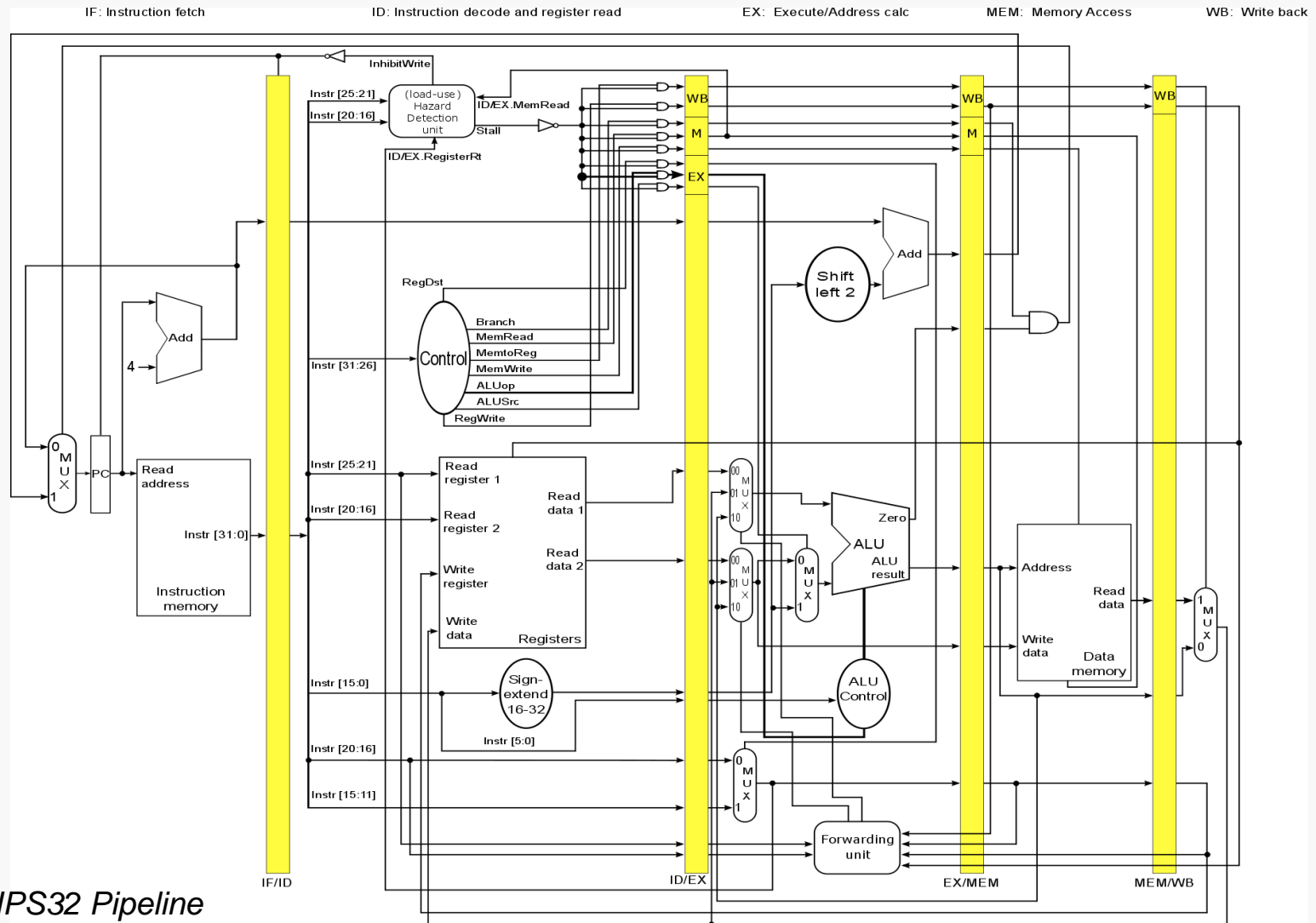


Computer Science 2506: Computer Organization II



MIPS32 Pipeline

Dimitrios Nikolopoulos

Email: `dsn@vt.edu`
Office: 122 McBryde
Office Hours: see course website

William D McQuain

Email: `wmcquain@cs.vt.edu`
Office: 634 McBryde
Office Hours: see course website

<http://courses.cs.vt.edu/cs2506/Spring2020/>

Detailed coverage of a RISC architecture, including instruction formats and construction, addressing modes, instruction execution, digital logic, arithmetic/logic units, control unit design, memory hierarchy operation and performance, pipelining, input/output, and the relationships between high level programming languages and machine language, with practice in the development of medium-sized system software solutions in C.

The course will focus on the MIPS32 architecture.

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,
- Write moderately complex system programs in C,
- Compare computer architectures and organizations based on quantifiable performance metrics.

Rohit Kumar Chandaluri rohitchandaluri AT vt DOT edu

Xiang Cheng xiangcheng AT vt DOT edu

Nick Gill nsgill1 AT vt DOT edu

Redwan Ibne Seraj Khan redwan AT vt DOT edu

Josiah Denton jfdenton AT vt DOT edu

Chapman Pendery cpendery AT vt DOT edu

Lin Zhang lz2304 AT vt DOT edu

Ishaan Singh ishaan15 AT vt DOT edu

Office hours will be posted on the course website

CS 2114 Software Design and Data Structures

CS 2505 Computer Organization I

or

ECE 2504 Intro Computer Engineering

ECE 2574 Data Structure and Algorithms

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

Math 2534 Discrete Mathematics

or

Math 3034 Introduction to Proofs

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

Required:

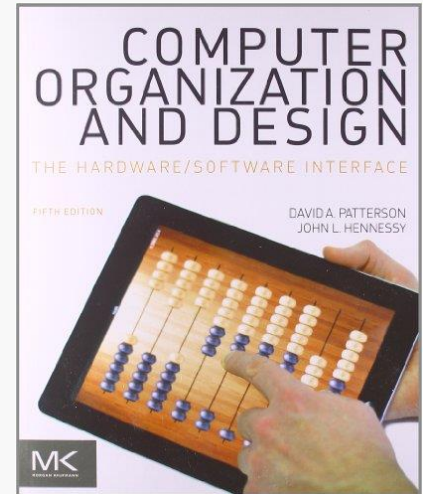
Computer Organization and Design: the Hardware/Software Interface, 5th Edition

David A Patterson and John L Hennessy

Morgan-Kaufmann ©2013

ISBN 978-0-12-407726-3

The 4th Edition will also be fine, but some of the reading assignments will not correspond to the organization of topics in that version.



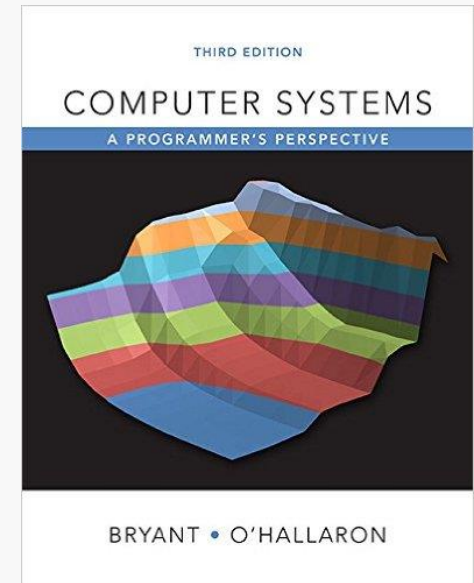
Computer Systems: a Programmer's Perspective, 3rd Edition

Randal E Bryant & David R O'Hallaron

Addison Wesley ©2016

ISBN 9780134092669

This contains considerable information related to the
Attack Lab assignment.

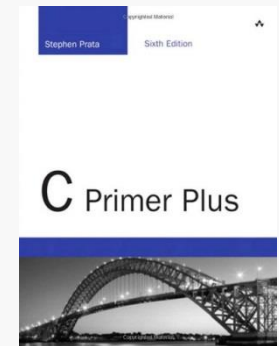


C Primer Plus, 6th Edition

Stephen Prata

Developer's Library ©2014

978-0-321-92842-9

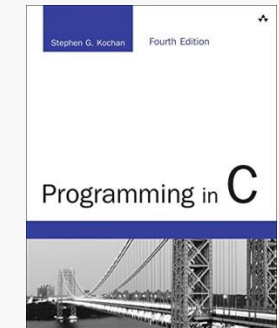


Programming in C, 4th Edition

Stephen G Kochan

Developer's Library ©2014

978-0-321-77641-9



CS 2506 Course Notes, Spring 2020 Edition, W D McQuain, ©2005-20
(available at the course website)



Available via the Safari Database in the VT Library:

The Art of Debugging with GDB, DDD, and Eclipse

N Matloff & P J Salzman, No Starch Press ©2008 ISBN 978-1-593-27174-9

Write Great Code, Volume 1

R Hyde, No Starch Press ©2004 ISBN 978-1-593-27003-2

Write Great Code, Volume 2

R Hyde, No Starch Press ©2006 ISBN 978-1-593-27065-0

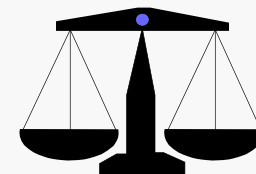
Final grades will be based on the average achieved over the following :



Item	Weight	Dates
Assignments		See course website
Digital Logic	6%	
MIPS	28%	
C/x86-64	30%	
Midterm Exam	12%	In class on Thursday March 5
Final Exam*	24%	16:25 – 18:25 Friday May 8

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 your midterm 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:

- Attack Lab** You must trigger at least:
- I CTARGET 1-3
 - II RTARGET 4
 - III 70% success rate

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.

Each assignment will specify certain requirements:

- exactly what is to be turned in (for example, which C files)
- the format of the files to be turned in (for example, plain text, tar file, etc.)
- the time by which solutions must be turned in for full credit
- whether solutions will be accepted after that time

It's your responsibility to make sure that you turn in the correct thing(s), in the specified format, and by the specified deadline.

It's our responsibility to accurately evaluate what you turn in.

There are many students, and a small course staff.

Therefore, we will not provide any special treatment in cases where, due to student inattention, the wrong thing is submitted, or the submission is in the wrong format, or the submission is late.

You will probably be allowed to work together on some assignments. If you are allowed to work in pairs, or groups, it is important you understand what we expect.

Acceptable pairs work requires:

- Each partner contributes to the analysis of the assignment, and to the derivation of a solution. This does not mean the contributions will always be equal, but both partners must be actively involved. When the solution is complete, each partner should understand the entire solution.
- The partners do not "divide" different parts of a multi-part assignment, with each working independently on his/her parts and having little or no involvement in the other parts.
- No partner "runs away" with the problem and solves it independently, not giving the other partner(s) the opportunity to contribute.

If you are allowed to work in pairs, you will choose your partner. Choose wisely.

We will not make any accommodations for situations in which one partner makes an error in a submission; it is the joint responsibility of partners to make sure that everything is correct.

Test Environments

- When relevant, a test environment, will be specified for homework assignments.
- The C-language assignments will be compiled with gcc 4.8 (as installed on the rlogin cluster) and tested on CentOS 7.
- All assignments will be compiled as 64-bit code when graded.
- 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.



For each of the programming assignments used in the course, we will provide you with a testing/grading environment that exactly duplicates most, if not all, of the steps we will use in grading your submissions.

We will not make any adjustments to our grading procedures, so pay full attention to the information that is provided by these testing/grading environments.

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 two weeks after the graded assignment has been made available to you.

This policy is motivated by several considerations:

If we have graded something incorrectly, that needs to be fixed for all the students who might have been involved, and that should be done as soon as possible.

If we have deducted points for a valid reason, you need to understand why your answer was inadequate, and you need to understand that before your confusion leads to errors on later assignments.

This policy is NOT intended to provide you with an opportunity to fix your errors after an assignment has been graded.

Our obligation is to correctly evaluate what you submit for grading.

Your obligation is to submit correct work, in the correct format, by the specified deadline.

File Formats

Each assignment in this course will include precise instructions for the type of file that is to be submitted (e.g., zip, tar, text, jar, etc).

Such requirements may stem from a variety of concerns, none of which are negotiable.

If you do not understand how to create a file of the specified type, discuss that with a member of the course staff.

Submission Contents

Each assignment in this course will include precise instructions for exactly what must be submitted to us.

If you make a submission that is incomplete, or otherwise incorrect, we will grade that.

It is your responsibility to make sure that you submit the correct information.

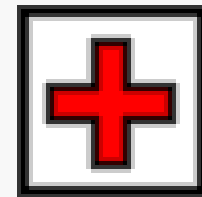
Special Treatment

There will be none.

There are many of you, and few of us, and I will not permit the TAs to spend extra time to compensate for your failure to follow the instructions that are given in the assignments.

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.

Systems fail. Drives die. Bad sectors appear.

Network connections break.

Plan for it. It is inevitable!



There are certainly legitimate uses of laptops and tablets during lectures.

But, most students who use laptops or tablets during class are essentially unaware of the details of the material that is being presented, and are therefore effectively absent.

The use of most other devices, such as cell phones and media players, is almost certainly a pure distraction to the student.

Therefore:

- You may not access your cell phones or media players during lectures.
- You may not use Google Glass during lectures.

You may use an audio recorder during class.

The Undergraduate Honor Code pledge that each member of the university community agrees to abide by states:

“As a Hokie, I will conduct myself with honor and integrity at all times. I will not lie, cheat, or steal, nor will I accept the actions of those who do.”

The assignments in this course may include a more specific statement, which is binding on your conduct on those assignments.

Students enrolled in this course are responsible for abiding by the Honor Code.

A student who has doubts about how the Honor Code applies to any assignment is responsible for obtaining specific guidance from the course instructor before submitting the assignment for evaluation.

Ignorance of the rules does not exclude any member of the University community from the requirements and expectations of the Honor Code.

For additional information about the Honor Code, please visit:

<https://www.honorsystem.vt.edu/>

Commission of any of the following acts shall constitute academic misconduct. This listing is not, however, exclusive of other acts that may reasonably be said to constitute academic misconduct. Clarification is provided for each definition with some examples of prohibited behaviors in the Undergraduate Honor Code Manual located at <https://www.honorsystem.vt.edu/>

CHEATING

Cheating includes the intentional use of unauthorized materials, information, notes, study aids or other devices or materials in any academic exercise, or attempts thereof.

PLAGIARISM

Plagiarism includes the copying of the language, structure, programming, computer code, ideas, and/or thoughts of another and passing off the same as one's own original work, or attempts thereof.

FALSIFICATION

Falsification includes the statement of any untruth, either verbally or in writing, with respect to any element of one's academic work, or attempts thereof.

FABRICATION

Fabrication includes making up data and results, and recording or reporting them, or submitting fabricated documents, or attempts thereof.

MULTIPLE SUBMISSION

Multiple submission involves the submission for credit—without authorization of the instructor receiving the work—of substantial portions of any work (including oral reports) previously submitted for credit at any academic institution, or attempts thereof.

COMPLICITY

Complicity includes intentionally helping another to engage in an act of academic misconduct, or attempts thereof.

VIOLATION OF UNIVERSITY, COLLEGE, DEPARTMENTAL, PROGRAM, COURSE, OR FACULTY RULES

The violation of any University, College, Departmental, Program, Course, or Faculty Rules relating to academic matters that may lead to an unfair academic advantage by the student violating the rule(s).

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 assignments 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.

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 cheating analysis via the MOSS system.

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.

If you have questions or are unclear about what constitutes academic misconduct on an assignment, please speak with me.

I take the Honor Code very seriously in this course. The normal sanction I will recommend for a violation of the Honor Code is an **F*** sanction as your final course grade. The F represents failure in the course. The “*” is intended to identify a student who has failed to uphold the values of academic integrity at Virginia Tech.

A student who receives a sanction of **F*** as their final course grade shall have it documented on their transcript with the notation “FAILURE DUE TO ACADEMIC HONOR CODE VIOLATION.”

You would be required to complete an education program administered by the Honor System in order to have the “*” and notation “FAILURE DUE TO ACADEMIC HONOR CODE VIOLATION” removed from your transcript.

The “F” however would be permanently on your transcript.

In recent terms we have observed a new behavior regarding cheating.

Some of the projects we use each semester may have been used in previous offerings of the course.

A small number of students have submitted solutions that were based (in whole or in part) on solutions submitted by other students in previous offerings. Be advised:

- Using another student's solution and supplying a solution to another student are both cheating.
- When we do the cheating analysis for a project, we include submissions from previous offerings as well as the current term. (We save everything.)
- If we detect this form of cheating, the students from the previous terms will also be charged with violating the Honor Code.
- A charged student will not receive a degree until the charges are resolved.
- The University does make provisions for cases in which a charged student has already graduated. You do not want to discover how this works, so safeguard your code for the long term!

At some point you will very likely want to make samples of the code you write available to potential employers. There is absolutely nothing wrong with that!

But if you do that, you need to be sure you're not making it easy for other students to access your code.

GitHub is an interesting service, but you need to use the protection options that the free student accounts offer.

And there are alternatives...

Students are encouraged to address any special needs or special accommodations with me during the first two weeks of the semester, or as soon as you become aware of your needs.

Those seeking accommodations based on disabilities should obtain a Faculty Letter from the Services for Students with Disabilities office (540-231-0858) located in Lavery Hall, Suite 310) <http://www.ssd.vt.edu/>.

If you need adaptations or accommodations because of a disability (learning disability, attention deficit disorder, psychological, or physical), if you have emergency medical information to share with the instructor, or if you need special arrangements in case the building must be evacuated, please meet with the instructor as soon as possible.