CSC236 - Basic Computer Organization And Assembly Language Programming  
Syllabus For On-Campus And Distance Education Sections - Fall 2016

1.0 Actions To Be Taken Immediately After The First Lecture

- Read this Syllabus and the CSC236 Calendar which contains due dates for assignments.
- Get the CSC236 Class Notes and Overhead Projections course packages, available at the NCSU bookstore and online.
- Complete and submit HW0.
- Install DOSBox.
- Retrieve the automated homework generator program and generate the homeworks HW1 - HW4.

2.0 Prerequisites

The prerequisite for this class is a C in CSC216.

3.0 Instructor

Mr. Dana Lasher     Phone: 515-7890     Email: lasher@ncsu.edu     Office: 2296-EBII
Office hours are posted on the WEB.

4.0 Course Topics

This course explains what happens beneath High Level Languages such as C++ and Java. It covers the history of computing, number systems, von Neumann architecture, instruction sets, machine code, assembly language programming, program testing, compilers, logical operations, microprogramming and interrupts. It includes a detailed study of a contemporary processor, the Intel x86 family; as well as introducing other processors such as the Java Virtual Machine, Floating Point Numeric Data Processor, ARM.

5.0 Course Objectives

At the end of the course, students will be able to:
- Add and subtract and convert, signed and unsigned integers, using bases 2, 10 and 16.
- Enumerate the functional components of a computer; explain trade-offs in computer design as they relate to cost and function and performance; outline computer architectural enhancements beyond the von Neumann model.
- Explain the basic operation of interrupts and microcode.
- Program in x86 assembly language and ARM assembly language
- Link assembler subroutines with a High Level Language.
- Convert symbolic assembler code into machine code and convert machine code into symbolic assembler code.
- Explain Floating-Point architecture and program the Floating-Point co-processor
- Explain the basic operation of the Java Virtual Machine and Java Bytecode.

6.0 Information On The Web And Information Sent To Your NCSU Email Account

- Class messages are sent to your NCSU email account. You must check that account regularly.
- The WEB site is: http://wolfware.ncsu.edu

7.0 Text And Lectures

- The CSC236 Class Notes course package is the text. It is at the NCSU bookstore and on the WEB site. It contains a detailed set of lecture notes and reference information for programming the Intel 8086 processor.
- For Distance Students the course content is delivered via video lectures which are viewed on the WEB.
8.0 Tests

Our open book tests measure your ability to use the information taught. They may contain new types of questions that require you to apply your knowledge. To prepare, you must take as many old tests as you can. Studying old tests is considered an integral part of the learning process. Consider the old tests as a set of examples and an extension of the notes.

In the test you may use anything written down (typed, printed, and handwritten). Thus these are all okay: old tests, homeworks, other notes, any text book, class notes, overhead projections. You may not use anything electronic (no calculator, no computer).

If you must miss a test, contact the instructor before the test to get agreement. For example you are going on a trip. In emergency situations contact the instructor as soon as you return to school. Illness must be certified by a physician in writing. Make-up tests are only given for university excused absences. The options for taking tests for Distance Students are described on the web.

9.0 General Grading Policies

Our grading policies have these goals.

- Emphasize that in the business world products need to be delivered on time.
- While in the learning mode, allow students to recover from the pressures of life that create difficult situations.
- There should be no surprises in the grade a student receives for an assignment.

When are assignments due.
- The due dates for all assignments are given in the CSC236 Calendar and all assignments are due at noon.

Self grading assignments.
- Most assignments are self-grading. You will know your grade when you submit the assignment.
- Grade recording is automated. Incorrect or forgotten submissions cannot be processed and the assignment is considered late.
- You must correctly electronically submit the required file on time. An assignment is considered complete at the time that the correct file, specified in the assignment handout, is submitted to the assignment's submit locker. The submit time stamp is the time used to determine any bonus or penalty.
- You may re-submit any assignment as long as the submit locker is open. We will record the last grade submitted.

Special problems.
- If you have extenuating circumstances that may warrant an extension, contact the instructor before the assignment is due to request alternative arrangements. Do not wait until after an assignment is due to bring up special situations.

This Grade Table lists the graded components of the class and their value toward a course grade

<table>
<thead>
<tr>
<th>%</th>
<th>Description</th>
<th>Late Policy</th>
<th>Bonus Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>68</td>
<td>Two open book tests and a final. The lowest test counts 18% and the other two count 25% each. You may use notes and books. You may <strong>not</strong> use any form of calculator or computer.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>HW0 - HW8 homeworks</td>
<td>not accepted late</td>
<td>no bonus for early submission</td>
</tr>
<tr>
<td>1</td>
<td>TOOLS homework</td>
<td>not accepted late</td>
<td>no bonus for early submission</td>
</tr>
<tr>
<td>2</td>
<td>KEY homework</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
<tr>
<td>2</td>
<td>FLOAT homework</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
<tr>
<td>2</td>
<td>LINKHILL homework</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
<tr>
<td>4</td>
<td>P1 - Program 1</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
<tr>
<td>4</td>
<td>P2 - Program 2</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
<tr>
<td>4</td>
<td>P3 - Program 3</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
<tr>
<td>4</td>
<td>P4 - Program 4</td>
<td>accepted late with penalty</td>
<td>bonus for early submission</td>
</tr>
</tbody>
</table>

100% Total points for all work

The course grade is composed of 100 potential points, and is mapped into a letter grade.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>97.0 - 100</td>
</tr>
<tr>
<td>A</td>
<td>93.0 - 96.9</td>
</tr>
<tr>
<td>A-</td>
<td>90.0 - 92.9</td>
</tr>
<tr>
<td>B+</td>
<td>87.0 - 89.9</td>
</tr>
<tr>
<td>B</td>
<td>83.0 - 86.9</td>
</tr>
<tr>
<td>B-</td>
<td>80.0 - 82.9</td>
</tr>
<tr>
<td>C+</td>
<td>77.0 - 79.9</td>
</tr>
<tr>
<td>C</td>
<td>73.0 - 76.9</td>
</tr>
<tr>
<td>C-</td>
<td>70.0 - 72.9</td>
</tr>
<tr>
<td>D+</td>
<td>67.0 - 69.9</td>
</tr>
<tr>
<td>D</td>
<td>63.0 - 66.9</td>
</tr>
<tr>
<td>D-</td>
<td>60.0 - 62.9</td>
</tr>
</tbody>
</table>

C- is required for S
D- is required for AU

Syllabus Page 2   Updated: 07/04/2016
10.0 Late Work Policy

- All work is due at noon on its due date
- Late assignments can only be submitted if their late submission locker is still open.
- These assignments are not accepted late: HW0-HW8, TOOLS. Their lockers close at noon on the assignment's due date.
- These assignments are accepted up to noon 7 days late with a penalty: KEY, P1, P2, P3, P4, FLOAT, LINKHLL.
- Late submissions are made to the late submission locker associated with the assignment
- The late penalty is 15 points.
- All lockers close at noon on the day of the last lecture ... At that time work is no longer accepted.

11.0 Early Submission Bonus

- All work is due at noon on its due date
- Early assignments can only be submitted if their early submission locker is still open.
- These assignments are eligible for the early submission bonus: KEY, P1, P2, P3, P4, FLOAT, LINKHLL.
- Early submissions are made to the early submission locker associated with the assignment.
- Early submissions must be before noon at least 3 days before the assignment's due date.
- The early submission bonus is 15 points.

12.0 Homeworks HW0-HW8

For HW0 create a file named **hwanswer.txt** and submit it electronically using Moodle.

Running the HW1-HW8 homework program requires a 16 bit DOS box. The DOSBox environment is recommended.

- They are turned in electronically using Moodle.
- The only item submitted is the **hwsubmit.txt** file created by the grading program.

If you caused a problem in submitting HW0 or HW1-HW8 (e.g. forgot to submit, had the wrong file name, submitted the wrong file, submitted to the wrong locker), but we judge that you had completed and made a reasonable effort to submit the homework on schedule, then we will enter the homework grade manually with a 15-point penalty deduction.

13.0 Programs, KEY, FLOAT and LINKHLL Assignments

13.1 The development software requires a 16 bit DOS box. The DOSBox environment is recommended.

13.2 Treat each program as if you worked for a company and this was your project.

13.3 Do not circumvent the automated grading process. Examples include: fake documentation; tuning your program for the grading system's specific test cases; any manual intervention during the running of the grading system batch files.

13.4 Do not use any form of automated code generation such as .if statements, macros, high-level languages, etc.

13.5 Programs must only use 8086 instructions and the small memory model (directives .8086 and .model small).

13.6 Documentation and efficiency are only graded and given points after the program passes the functional tests.

13.7 If a grading system defect assigns your program the wrong grade then the program will be re-graded manually. You have one week from the time you are notified to resubmit your program. During that week, no additional late penalty will be applied. This late waiver does not apply to intentional actions to circumvent the grading system or documentation.

13.8 Any design information, algorithms, model high-level language code, model assembler code provided is only meant to show functionality and not efficiency. Assembler code efficiency is the responsibility of the coder.

13.9 The submit time stamp is the time used to determine any bonus or penalty.

**Do not edit or modify or erase any of the grading system files.**

The file you submit for KEY, FLOAT, LINKHLL and the Programs is named **xxx.ans** where **xxx** is the assignment name. That file is created by the grading system. No other file is acceptable.

Incorrect submissions will result in your program not being graded and considered late.
14.0 Rules Relating To Academic Integrity

All work that you turn in for grading must be your own. This means that all work must be an independent and individual creation by you. Any attempt to gain an unfair advantage in grading, whether for yourself or another, is a violation of academic integrity. The only exception is when the instructor states that you may work as a team in which case, the work is an independent creation of the team.

The only people that you may receive help from on an assignment are your instructor and the official TAs for the class.

The minimum penalty for any incident is -100 on the assignment. All incidents are reported to the Office of Student Conduct.

All of these are academic integrity violations.
- Plagiarizing another person's work. This includes all forms of using code written by another person in whole or part.
- Using code you find online (code from a textbook is okay with proper citation).
- Having another person help you design or code your assignment.
- Requesting help on a public WEB site or bulletin board.
- Giving your work to another person, even after the course is over. We periodically reuse programs and you are responsible if someone submits your program even in a later term.
- Posting your work on a public WEB site or bulletin board.
- Any intentional attempt to circumvent the automated grading process.

Examples of NOT Cheating:
- Researching algorithms (from the web or textbooks with citation in the header)
- Using code from the class website (with citation in the comments).
- Using code from other programs YOU wrote.
- Help from the instructor or TAs.
- Using code from a published textbook (with citation in the comments).

This course has two different environments for assignments. The calendar lists the designation for each assignment.

Individual assignments.
- You must work completely alone on these assignments, except for help from the TA or instructor.
- Do not discuss the assignment with anyone. Do not work with anyone else on design or coding. Do not give or take ideas on how to solve the problem; clever ideas and a good design lead to higher grades on the programs and those higher grades are only due the person who thought up the design ideas. Do not jointly write any assembler instructions. Do not give or receive any actual assembler code. Do not give or receive program listings. All instructions you submit, must have been designed, developed and written yourself.

Team assignments. (Team participation is optional.)
- You may select one student who is enrolled in this class to form a team. A two-person team is treated as a single unit.
- You may communicate within the team and share ideas and code.
- The names of both team members must appear in the owner field of the program header.
- Create a single xxx.ans file on one computer that is submitted by both members of the team to their respective submit lockers.
- Communication outside the team is not permitted and is subject to the same restrictions as individual assignments.

15.0 Notes On Submitting Assignments

- If you have any problem with the submit process, including submitting an incorrect file, then before you request that the late penalty be waived, answer these two questions.
  1. Were the instructions that specified what to submit and when to submit clear?
  2. Was the error caused by your actions?
- If the answer to both questions is "yes" then please do not request the penalty be waived.
- If the answer to either is "no" then send an email with an explanation.
16.0 Help From The Staff

- Send all technical questions to the instructor and TAs
- Response time targets: 3 hours during the business week 8-5; the next business day for questions sent at night; we try to answer questions on holidays and on the weekend but that is not guaranteed.

16.1 help with old test questions and homeworks

Many issues with old test questions can be answered directly by the student by entering the question into a small program and assembling it and looking at the listing file or running the program through the debugger. This process gives the student more experience with the tools and increases their self-sufficiency. That is the expected process to follow in all cases where it is practical. If after following that process, there are still unresolved issues then send the staff an email.

- Only refer to a single test question per email. Specify the term and test number. Copy the whole test question into the email.
- You explain how you answered the question. We will indicate which step in your process was incorrect.

Based on the above and that the response time target is the next business day for questions sent at night, trying to get multiple questions answered the night before an exam will not be practical. Study for exams earlier than the night before.

16.2 help with programs

- Have a design that can be sent electronically. Before looking at your assembler code, we look at your design to determine if you have a reasonable solution.
- Have line comments and block headers in the code that tell what the code is intended to do.
- If the problem is a bug, have one simple test case that fails; the actual output from your code and the expected output.
- Submit all items to the ask4help assignment submit locker, or send as an email attachment.
- Send an email to the support group: state you have submitted code to ASK4HELP; provide a concise explanation of the issue to be resolved.

17. Documentation and Communications

17.1 Programs must adhere to the documentation requirements that are specified on the WEB. Your documentation must be meaningful. Whether or not you consider documenting a program as a useful function, you must act professionally and provide useful documentation. Do not just write words to fill space. There are some documentation aberrations that are not detected by the automated grading system. They can however be detected by the staff after an assignment is submitted. If any of these are detected, the program will lose up to all the credit awarded for documentation; the exact amount to be set by the staff. Examples include:

- Using a header block that does not match the program e.g. from a previous assignment or from sample code provided.
- Specious documentation where the comment just echoes the assembler instruction but provides no useful information. For example `add ax,bx ;ax=ax+bx`
- Meaningless documentation that appears to just fill space in order to get credit.

17.2 A high degree of professionalism is required with email communications. You must adhere to these email rules:

- Identify the course, section, and your name in the subject line along with the subject of the message. For example: "CSC236-001 Walter Denton - Question about program 1".
- Include a salutation to identify the recipient. For example, to a specific staff member, you can use Mr. Lasher or to the support group you can use Greetings support staff. You now have the attention of the email recipient.
- Write full and proper English sentences with correct punctuation. Use a spelling checker.
- If you have several items, number them for ease of reading. The response will also be easier to understand.
- Be both concise (do not ramble) and clear (assure what you say is understandable). To quote Albert Einstein ... “Make everything as simple as possible, but not simpler.”

18. Disability Accommodations

Reasonable accommodations will be made for students with verifiable disabilities. In order to take advantage of available accommodations, students must register with Disability Services for Students at 1900 Student Health Center, Campus Box 7509, 515-7653. Students registered with Disability Services should present their letters of accommodations to the instructor prior to the end of the first week of classes.