SYLLABUS

<table>
<thead>
<tr>
<th>COURSE</th>
<th>CSCI 274 (Computer Architecture) Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>INSTRUCTOR</td>
<td>Prof. Aftab Ahmad</td>
</tr>
<tr>
<td>E-MAIL</td>
<td><a href="mailto:Aahmad.alt@gmail.com">Aahmad.alt@gmail.com</a> and aahmad@jjay</td>
</tr>
<tr>
<td>OFFICE</td>
<td>6.65.12NB</td>
</tr>
<tr>
<td>TELEPHONE #</td>
<td>Office (212) 393-6314 Text (757)550-9700</td>
</tr>
<tr>
<td>OFFICE HOURS</td>
<td>Tue / Thu Appt. Meeting</td>
</tr>
<tr>
<td></td>
<td>Tue 3:05-4:20 PM NB 6.64.02. Thu Online</td>
</tr>
<tr>
<td>FACEBOOK:</td>
<td><a href="mailto:aahmad.alt@gmail.com">aahmad.alt@gmail.com</a></td>
</tr>
</tbody>
</table>

OBJECTIVES:
This course introduces several concepts and their practice at the interface between computer software and hardware. This includes (i) representing information in computers, (ii) the CPU registers and (iii) assembly language programming.

DESCRIPTION:
The course will introduce several new concepts at the interface between computer hardware and software. Students will learn the significance of base number systems and do problems in decimal, binary, octal and hexadecimal systems and conversions back and forth. They will learn how unsigned and signed integers are stored in computer memory and the IEEE Standard 754 for storing floating point numbers. With intel x64 processor as an example, students will learn the functions of various registers. Then, using a contemporary assembler, they will do programming solutions that demonstrate the efficient usage of CPU registers. Upon successful completion of this course, you will understand the capabilities and limitations of a CPU in manipulating instructions and data. Instruction set architecture (ISA) for the intel x86-64 processors, and the role of CPU register selection for efficient lower level programming in the 64-bit x86-64 processors.

Course Learning Outcomes:
1. Be able to understand that numbers can be represented in multiple forms. Be able to convert signed and unsigned integers from one system to another.
2. Understand and be able to use the functions of various hardware components of a computer, such as processor registers and memory by being able to recognize simple code in lower level language for a contemporary processor, such as x86-64.

How the course works?
This is a hybrid class with face-to-face (f2f) meetings on the first of the two days of the week, and online posting on the second of the two days of the week. In one meeting online a Blackboard posting with an announcement will be made followed by a lecture on the same topic or vice versa. The posting of f2f meeting will include either homework assignment, or a quiz, the quiz being either in class or on Blackboard, or in class but on Blackboard. All homework will be submitted via the Blackboard. Please have a good understanding of using the Blackboard. Consult various tutorials provided on the Blackboard site for this purpose. Keep track of announcements on Blackboard every week. The first-week-day office hour is in my office and the second-week-day office hour is online. Please make an appointment for each office hour meeting. For online meeting, I am available via the Skype, phone, text or even instant email exchanges.

ATTENDANCE
Each student is expected to be in class during face-to-face (f2f) meeting. Class attendance will be taken regularly. Attendance on online day is via participation, asynchronously, unless otherwise announced for events such as Discussion Forums, in which case participation will also act as class attendance.
**METHOD OF ASSESSMENT:**

1. Handwritten notes by students are allowed to be used during tests and final exams.
2. There will be more than seven (7) weekly tests but only 7 counted. There is no remake regardless of the reason for missing a test.
3. All tests will be during the first 15 minutes of class.
4. The Final Exam will be as announced in the Final Exam Schedule.
5. The weekly discussions will be on the following topic every week. First, between Tue and Thu, every student comments on the Discussion Board what they understand in the topic of this week in at least 5 sentences. Then, between Thu and Mon, every student discusses the weekly test topics in his/her own way. There are 50% points for each posting. The Thu-Mon posting can be a response to another student's posting but must be 5 or more sentences to get credit in either case.

- **One Final examination** ... 200 point
- **Seven Weekly Tests** ... 700 points
- **Weekly Discussions** ... 100 points

**THE WEEKLY TESTS EXPECTATIONS:**

During the semester, weekly Tests will be given. The questions in a Test will be from the topics covered during the previous lectures. This is done to encourage students to cultivate the habit of preparing for the classes on a weekly basis, and hence be ready to learn new concepts the following weeks. This also helps them to prepare for the final examination. Generally a Test will be given at the beginning of a class or during a specified time online and students who are absent or late for the class will not be allowed to take that Test at a later time. Test can be in the form of a Home assignment, given in order to assess student's ability to solve problems independently without supervision. Typically, a homework assignment will be due in a week time and corrected assignments and individual feedback will be given within a week from submission deadline. A general feedback for the whole class will be included in the class announcements.

**LIST OF WEEKLY TEST TOPICS**

1. Number conversion between decimal, binary, hexadecimal and Octal.
2. Signed and unsigned integers – general
3. Signed and unsigned integers, 2’s and 1’s complements
4. Floating point number conversion and IEEE 754 representation
5. Midterm Examination
6. CPU and its components
7. Storage media and their characteristics
8. Machine language, assembly language and higher level language, x64 assembly Programming
9. x64 Assembly Programming

**GRADING POLICIES/FINAL PROJECT:**

A student who shows significant comfort level in following the concepts in class can optionally replace the final exam with a project involving lower level programming through a mutual consent with the instructor. The purpose of this project is allow a student to get more than what is taught in the classroom.

**Plagiarism:** If two or more submitted projects are identical, except for some minor variations, then a score of zero will be assigned to all identical projects.

**PROPOSED SCHEDULE OF LECTURES:**
<table>
<thead>
<tr>
<th>LECTURE</th>
<th>TOPICS</th>
<th>Tue f2f Thu online</th>
</tr>
</thead>
</table>
| Week 1  | • Base-q number systems  
          • Binary number system  
          • Counting in binary  
          • Conversion between decimal, binary octal and Hex numbers | |
| Week 2  | • Decimal value of a binary number  
          • Decimal to binary conversion  
          • Signed and Unsigned integers in general (Assignment 2) | |
| Week 3  | • Discussion on weaknesses  
          • Signed and unsigned integers (1's and 2's complement) | |
| Week 4  | • Conversion between various number systems  
          • Hexadecimal and Binary  
          • Octal and Binary | |
| Week 5  | • Signed and Unsigned Integers  
          • Sign bit  
          • Bias  
          • One's complement system | |
| Week 6  | • Floating point numbers  
          • Scientific notation  
          • Normalized scientific notation | |
| Week 7  | • Storing floating point numbers  
          • Binary normalized scientific notation  
          • IEEE 754 Standard for floating point representation  
          • Practicing IEEE 754 representation | |
| Week 8  | • CPU components and their interconnection | |
| Week 9  | • Computer storage types  
          o Speed, size and proximity to the processor  
          • CPU registers  
          • X64 Registers, types, function, addressing  
          • | |
| Week 10 | • Machine language, assembly language | |
| Week 11 | • Higher level language versus lower level language  
          • X64 Instruction format  
          • X64 program structure | |
| Week 12 | • Assembly programming editors  
          • Input / Output registers  
          • Input / Output Instructions  
          • Syscall  
          • Assessment 10 (Programming) | |
| Week 13 | • X64 data structures  
          o Word, half, byte, label  
          o X64 programming examples | |
| Week 14 | **FINAL EXAM PREP** | |
RESOURCES
TEXTS:
• Notes distributed by instructor via Blackboard, on number systems and assembly programming
• Lecture notes taken by students

REFERENCES:
• Author: Deconinck, Stéphane, Computer systems: a programmer's perspective, (2016).
  ISBN: 9781292025841’