CS301 Assembly Language Programming - Fall 2018

Course	73348
Section	F01
Credits	3
Prerequisites	CS201

Instructor	Dr. Orion Lawlor
Phone	907-474-7678
Office	ELIF 134
Email	lawlor@alaska.edu

Office Hours	MTWRF 3:30pm - 5pm
Meeting Time	MWF 11:45am - 12:45pm
Course Room	ELIF 301

Course Website	https://www.cs.uaf.edu/2018/fall/cs301

Required Texts	None - readings will be provided online

Course Description

Describes how your code executes on real hardware, which is key to debugging, security, and performance. Includes implementation details from assembly language, such as registers and opcodes; plain C programming, such as pointer arithmetic and malloc; modern languages such as JavaScript; and security implications of these features.

Student Learning Outcomes

After taking this course, students will be able to:

- explain how their code executes on real hardware.
- explain how the bytes of a buffer overflow attack compromise a machine.
- measure actual performance on a given architecture.

Course Schedule (Tentative)

For the week starting:

1. Aug 27: Assembly language syntax and semantics
2. Sep 03: Function calls in assembly, preserved registers, and the stack
3. Sep 10: How pointers really work: memory, arrays, buffer overflows

4. Sep 17: Structs, classes, and linked data structures in assembly
5. Sep 24: Mixing assembly language, C, and C++
6. Oct 01: Stack smashing buffer overflow attacks
7. Oct 08: Performance Programming
8. Oct 15: Project 1 presentations, continue Performance Programming
9. Oct 17: Midterm exam in class
10. Oct 22: Floating-point numbers and float hardware
11. Oct 29: SIMD Parallel Programming
12. Nov 05: Caches and memory access performance
13. Nov 12: Kernel mode, virtual machines, pagetables and mmap
14. Nov 19: Writing malloc

    Nov 21 Wednesday: optional lecture on quantum computing

    Nov 23 Friday: no class due to Thanksgiving
15. Nov 26: ARM programming
16. Dec 3: Project 2 presentations
17. 10:15 a.m.-12:15 p.m., Monday, December 10: Final Exam

Grading Policies

Weight	Description
5%	Attendance and class participation, graded at random intervals throughout the semester
35%	Homework problems, to be assigned through the semester
15%	Project 1, an independent project, typically due near the time of the midterm exam
15%	Project 2, a second independent project, possibly an extension of Project 1, typically due near finals week.
15%	Midterm exam, held in class in mid-October
15%	Final exam, held on finals week in December

Grades will be assigned based on the following percentage intervals:

A+	*	**A**	[93%, 100%)	**A-**	[90%, 93%)
B+	[87%, 90%)	**B**	[83%, 87%)	**B-**	[80%, 83%)
C+	[77%, 80%)	**C**	[73%, 77%)	**C-**	[70%, 73%)
D+	[67%, 70%)	**D**	[63%, 67%)	**D-**	[60%, 63%)
F	[0%, 60%)				

* A+ indicates truly exceptional work, above and beyond the course requirements.

Late Work Policy

Late work will not be graded, unless it is due to circumstances beyond your control, or if you turn it in before I begin grading. I may begin grading at any time after the due date, even 12:01am the next day (grading is an effective treatment for insomnia!). You are encouraged to inquire if I have begun grading yet, since this acts as a reminder for me to do so.

Policies

Students are expected to be at every class meeting on time, and are responsible for all class content, whether present or not. If absence from class is necessary, in-class work (other than quizzes) and homework may be made up only if the instructor is notified as soon as possible; in particular, absences due to scheduled events must be arranged ahead of time. Academic dishonesty will not be tolerated, and will be dealt with according to UAF procedures. Students in this class must pay the CS lab fee. Payment allows access to the Duckering 536 lab.
UAF academic policies http://catalog.uaf.edu/academics-regulations/
CS Department policies http://www.cs.uaf.edu/departmental-policies/

How to Succeed in this (and every other) Course

The low-level programming we'll be doing in this course consists of dozens of features that are individually trivial, yet collectively are quite error prone.

Assembly language is famous for being hard. Careful attention to detail, and systematic scientific testing and debugging, are the keys to mastering it.

- **What** makes the code crash? I will often comment out key functionality in the program, with the explicit goal of measuring how the program's results change, even if the modified program is completely useless at the problem I'm trying to solve. Understanding the crash is what leads me to a working program later on.
- **When** did the code start crashing? I always work incrementally, starting with the simplest possible program (such as "return 5"), and adding small features one at a time. If you just added the print function, and the code starts crashing, it's much easier to debug than if you write everything first, then run the complete program. I try to structure homeworks so you are forced to build the code up incrementally.

- **How** does the code crash? I will often change code to force a crash earlier, to collect more data on the state of the machine leading up to the real crash.
- Which **assumption** is incorrect? Programs fail because you don't understand something. The goal is to narrow this down until you know what part you don't understand, so you can run more experiments or collect more data (or ask me!) until you do understand it.
- What have we **learned** so far? In a long debugging session, it is very easy to lose track of what you've determined from each experiment. I almost always have a text editor open where I paste in code snippets, experimental results, and free text with lessons learned, so I can see I'm making progress, and I don't forget things.

Your brain condition is also key:

- Are you correctly fed and hydrated? The brain is a liquid computer that runs on glucose, and automatically throttles down your higher brain functions when blood sugar is off nominal.
- Are you angry or upset? It is common to feel anger when your programs are not working, which does not help you understand why your programs are not working. The appropriate emotional response to the wrong output or a crash is "Ah ha! An interesting mystery. Let us begin the search for clues!" rather than "OH NO! WHY DID THIS HAVE TO HAPPEN!? WHY DOES C++ HATE ME?!". A perfect recipe for rage-soaked terror is waiting until the last minute.
- Is your study environment free of distracting inputs? The human brain is surprisingly poor at multitasking, which means minor distractions like incoming text messages can have a large impact on your ability to see what the program is telling you.
- Have you had enough sleep? Getting >8 hours of good sleep is like a super power. Many key brain maintenance functions are only run while offline, including learning.
- Are you sober? Many psychoactive substances degrade higher-level brain function for days, much longer than their acute effects. Even mild psychoactive substances such as sugar or caffeine can disrupt the high-intensity mental work of programming.

I personally find programming and even debugging to be a refreshingly zen experience, where I am so immersed in the task that the entire outside world fades away ("flow" or "being in the zone").

Inclusion Statement

UA is an AA/EO employer and educational institution and prohibits illegal discrimination against any individual: www.alaska.edu/nondiscrimination

The University of Alaska Board of Regents have clearly stated in BOR policy that discrimination, harassment and violence will not be tolerated on any campus of the University of Alaska. If you believe you are experiencing discrimination or any form of harassment, including sexual harassment/misconduct/assault, you are encouraged to report that behavior. If you disclose sexual harassment or sexual violence to faculty members or university employees, they must notify the UAF Title IX coordinator about the basic facts of the incident. Your choices for disclosure include:

1. You may confidentially disclose and access confidential counseling by contacting the UAF Health and Counseling Center at 474-7043.
2. You can get support and file a Title IX report by contacting the UAF Title IX coordinator at 474-7599.
3. You may file a criminal complaint by contacting the UAF Police Department at 474-7721.

From Disability Services: UAF is obligated to provide accommodation only to the known limitations of an otherwise qualified student who has a disability. Please identify yourself to UAF Disability Services by applying for accommodations. To be considered for UAF Disability Services accommodations, individuals must be enrolled for at least one credit as a UAF student. For more information contact Disability Services at uaf-disabilityservices@alaska.edu, 474-5655 or by TTY at 474-1827.