U N I V E R S I T Y

CS 240: Programming in C

Lecture 1: Course Introduction

Prof. Jeff Turkstra

© 2025 Dr. Jeffrey A. Turkstra

Lecture 01

- Objectives
- Course policies
- (*NIX commands)

Announcements

- Read Chapter 1 in K&R

Lab This Week

- We have lab this week!
- A hopefully somewhat gentle re-introduction to the UNIX command line
  - and gcc
- Only time the assignment will be submitted and due during lab!

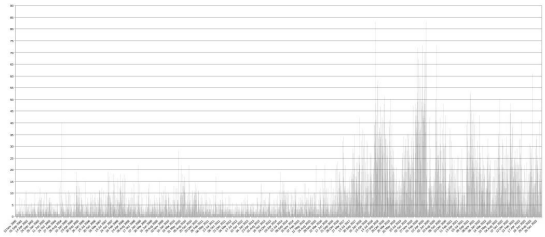
Instructor

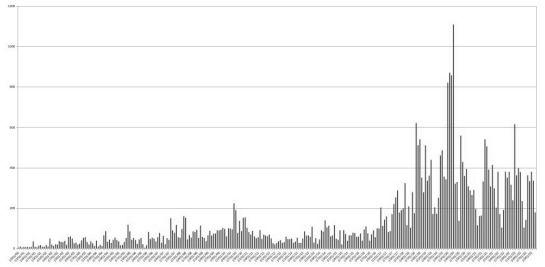
Dr. Jeff Turkstra
jeff@cs.purdue.edu
49-63088

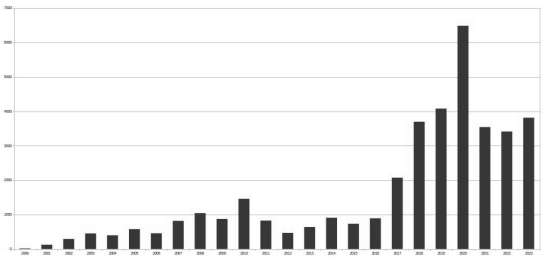
Office Hours:
TBD
(or by appointment)

Notes on Email

- Purdue IT has made email an unreliable form of communication at Purdue
- Email from me will originate from jeff@cs.purdue.edu
- Please send mail to jeff@cs.purdue.edu
- You must monitor your junk mail
  - Email from me may initially appear there. After marking it as not spam a few times, the system will learn it was wrong







About me

- BSCmpE, MSECE, and PhD from Purdue University
  - Focused on operating systems and distributed systems
  - Instructor, ECE 2005-2008
  - Software Engineer with HUBzero/RCAC
  - Microfluidic Innovations, LLC and other startups
  - CS starting January 2017
- Current teaching activity
  - CS 240, CS 307, CS 252, CS 180, CS 250, CS 50010, CS 50011
- Current research activity
  - Mostly instructional tools – PeerVal, C-Lab, Eastwood, and EnCourse
  - Metachory
- Enjoys
  - Linux, skiing, piano/saxophone, flying, HAM, etc

Course Staff

- Head TA:
  - Rebekah Sowards (rsowards@purdue.edu)
    - Contact for most course logistics-related issues
    - Most homework questions of interest should be posted on Ed Discussion!
- Homework Grading
  - Me! (jeff@cs.purdue.edu)
- Absences, Extensions, Accommodations, and DRC-Related:
  - Megan Bechtlofft (mbechtlo@purdue.edu)
- Instructional Specialist:
  - Justin Gillingham (jdgillin@purdue.edu)
    - Contact for Brightspace, Ed Discussion, etc technical difficulties

Teaching assistants

- Found on the website
- Office hours too (soon)
- Do not expect the TAs to do your homework
- Ask TAs <u>about your code</u> – don't ask the TAs to come up with code for you.

Course Website

- https://mandalore.cs.purdue.edu/~cs240/
- Contains things like:
  - Syllabus
  - Lecture videos and slides
  - Schedule
    - Lectures may be different, roughly same schedule

Syllabus

- You are responsible for reading and understanding the entire syllabus
- “I didn’t know that was the policy” is not an excuse
- Do not hesitate to ask questions!

Things you should already know how to do…

- How to write and compile a Java program in the UNIX environment
- How to declare variables
- How to manipulate arrays
  - Declaration/definition is a little different
- How to write a for loop!!!!
- How to write a while loop
- How to write a do-while loop
- How to use if statements
- How to use switch statements

- Logical operators (|| && !)
- How to write data with printf()
- How to write functions

- How to copy, move, execute, and delete files
- How to navigate a file system (pwd, cd, etc)
- How to edit files in a UNIX environment
  - Vi(m), Emacs, pico, nano, etc

IDE

- We will not use an IDE for this class
- You are expected to develop on data.cs.purdue.edu, or another CS Linux system
  - E.g., borgNN.cs.purdue.edu
  - Can do so remotely with ssh

Questions and contact

- Ed Discussion
  - https://edstem.org/us/join/nRphkc
- Lab
- Office Hours
- Email
  - https://mandalore.cs.purdue.edu/~cs240/contact.php

Lab

- Environment to work on your homework with help available
- Again:
  - Do not expect the TAs to do your homework
  - Ask TAs about your code – don't ask the TAs to come up with code for you.
- The first hour of each lab is reserved for students in that section only
  - Students of that section always have priority
  - TAs reserve the right to ask people to leave

Homework Assignments

- Usually due on Wednesdays at 9PM
- Usually assigned Monday of previous week
  - The overlapping 3 days are there as a buffer to help you manage your time
  - The expectation is that you are done with assignments no later than the weekend
- Approximately 12 assignments
  - 100 points each
- Style grade worth additional 20 points

Extensions/Excused Absences

- ...must have documentation
- Contact Megan, except for exams
- Quizzes are excused when approved (not included in grade computation)
- Homework extensions are typically not granted unless the incapacitation is > 3 days
  - Remember, we expect you to finish homeworks by Sunday
  - 3 extra days are there specifically for cases where you become ill, have other obligations, etc
- Exams are only rescheduled in the most extreme circumstances
  - Contact me directly

Code Standard

- Intended to serve as basic recommendations and requirements for how to write your functions
- Like a *dialect* or *accent* of a language
- All good software development environments use a code standard
- You may not agree with the format

https://mandalore.cs.purdue.edu/~cs240/code.php

Test Modules

- You will be provided with a test module for most assignments
  - Can run it as many times as you want
- If your program crashes at any point, you will receive a score of 0
- Test module will not progress to the next function until all tests up to that point pass
- Run your test modules often
  - Many are randomized!
  - We will run it many times for grading, and take the lowest score

Submission

- Programs that do not compile generally will not receive credit
- We will always grade the latest submission
- Only run "make submit" if you want us to grade the current version using the current date and time

Regrades

- Aside from exceptional circumstances, regrades are typically honored only when a bug in the test module is discovered
  - The entire course is then regraded
- Always compile and test your programs before submitting
  - Even for one line changes! We all make mistakes!

Help

- Be prepared to talk about steps you have already taken in debugging
- Avoid questions like “this doesn’t work. Why?”
- Instead, “I am having trouble with part X. I have printed out the values of variables Y and Z on lines K and N. I set a breakpoint for function F and stepped through to line W. I think that the issue involves variable Z ….”

Ed Discussion

- Great resource for posting questions about homework and other course content
- Any post involving code must be private
- We expect people to treat each other with respect
- Work to keep a positive atmosphere

!Ed Discussion

- Ed Discussion is NOT a place for:
  - Regrade requests – submit through gradescope for quizzes and code standard; or, email jeff@cs.purdue.edu for homeworks
  - Complaining
- Posts that are not productive or overly negative will be removed

Quizzes

- 25 points – one quarter of a homework
  - Submit no later than 24 hours after lecture
- 10 points – one tenth of a homework
  - Submit in lecture
- No submission? Score of 0 unless excused.
- See syllabus for more information

Lecture Quizzes

- You must have precise location turned on
- You cannot have privacy settings/VPN/proxying turned on
- It is your job to properly configure your device
- Apple devices in particular require extra effort to ensure the correct location data is conveyed

Exams

- Two midterms worth 14% each
  - Written!
- One cumulative final worth 22%
  - Also written

Grades

Homework and Quizzes	50%
Midterms	14% ea
Final	22%

Grading issues will be addressed as they occur. Not at the end of the semester.

Grade Determination

Hwk/Quiz Avg		Test Avg		Course Avg	Grade
>= 85%	and	>= 85%	and	>= 90%	A
>= 75%	and	>= 75%	and	>= 80%	B
>= 65%	and	>= 65%	and	>= 70%	C*
>= 55%	and	>= 55%	and	>= 60%	D
< 55%	or	< 55%	or	< 60%	F

* C threshold is often lowered

Academic Integrity

"If you claim someone else's work as your own, we call that 'cheating'".

- Do not do quizzes or exams for other students
- Asking questions is probably OK, but do not look at other students' homework files and do not work with other students. At all.
- Protect your work
  - If your work is turned in by another student, you are both guilty

AI/ML/LLMs/etc

- Using ChatGPT and similar software to generate partial or complete solutions to assignments, quizzes, exams, etc is also cheating
- Think of it like using a calculator instead of learning basic arithmetic
  - You're only hurting yourself and future career prospects

Academic Integrity

- When in doubt, talk to a TA or instructor instead of a fellow student
- All work is subject to computer-based comparison and analysis...

Academic Integrity

- The minimum penalty for any incident of academic dishonesty is a score of zero for the item in question
- More serious or repeated infractions will result in a grade of 'F' for the course
- All incidents will be referred to the Dean of Students and the Department of Computer Science

Academic Integrity

- Ask questions about the policy ***without risk***

Supplemental Instruction

- Monday/Wednesday 5:30pm – 6:20pm, WALC 3138
  - Keshav Shylesh
    - kshylesh@purdue.edu
- Monday/Wednesday 6:30pm – 5:20pm, WALC 2121
  - Manya Gupta
    - gupta969@purdue.edu
- Data show students that regularly attend SI sessions obtain higher grades in the course, on average

Tips for Success

- Write your programs out on paper first
- Start on homework projects as early as possible
- Practice
- Experiment
- Create a cool personal project
- Look at source code available on the Internet
- If you can learn to enjoy programming you are guaranteed to do well

How to Fail This Course*

- Assume that since your prerequisites were easy, this class will be too!
- Try to do the homework at the last minute
- Don't do the homework at all
- Don't come to lecture
- Get "help" from somebody else in the class
- Don't practice for the exams

* most students do quite well

Why C?

- It is still widely used and growing
  - TIOBE January 2024 index lists it as #2
  - Python is #1, C++ is #3
- Programming Language of the Year 2019
- Used in a huge number of embedded and IoT devices
- Ubiquitous for systems programming
- Small
- Fast

C vs. Java

- You already know how to write some C code
  - Java was designed using C/C++-style syntax
- But, there are many differences

Slides

- Some slides based on Prof. Rodgriguez-Rivera's, Prof. Fred Mowle's, and Dr. Richard Kennell's material

Java vs. C

Attribute	C	Java
language type	function oriented	object oriented
basic programming unit	function	class (ADT)
source portability	yes, when done right	yes
compiled binary portability	recompile for each architecture	"write once, run anywhere" (bytecode)
security	yes, when done right	more built-in safety
compiling	gcc hello.c	javac Hello.java
linking a library (e.g., math)	gcc -lm mycalc.c	javac MyCalc.java (no flags needed)
execution	a.out (or name of binary)	java MainClass
(runtime) array declaration	int *a = malloc(N * sizeof(*a))	int[] a = new int[N];
array size	often unknown	a.length
strings	'\0'-terminated char array	built-in immutable String type
using a library	#include <stdio.h>	import java.io.File;

© 2023 Dr. Jeffrey A. Turkstra

Java vs. C

Attribute	C	Java

accessing a library function	#include <math.h>
	x = sqrt(2.2);
		import java.util.Math;
		x = Math.sqrt(2.2);

standard output	printf("sum = %d\n", x);	System.out.printf("sum = %d\n", x);
reading from stdin	scanf("%d", &x);	int x = StdIn.readInt()
memory address	pointer	reference
manipulating pointers	*, &, +	not permitted
functions	int max(int a, int b)	public int max(int a, int b)
data structures	struct	class
methods	function pointers	yes
pass-by-value	yes	yes
allocating memory	malloc()	new
de-allocating memory	free()	garbage collection
constants	const and #define	final

© 2023 Dr. Jeffrey A. Turkstra

Java vs. C

Attribute	C	Java
variable auto-initialization	not guaranteed	yes, compile-time checking
data type for generic item	void *	Object
variable naming convention	sum_of_squares	sumOfSquares
file naming convention	stack.c, stack.h	Stack.java
assertions	assert	assert

* Not a complete list :-)

© 2023 Dr. Jeffrey A. Turkstra

For Next Lecture

- Familiarize yourself with C programming
- Ensure your computer account is setup
- Choose an editor. Practice using it
- Practice writing a few programs
  - "Hello, world"
  - Prompt for a string, print it back
  - Write something to do your calculus homework
- Read and review Chapter 1 of your text

Boiler Up!