CSE 220 – C Programming

Introduction

Objectives

- Learn syntax and semantics of C
- Learn general programming concepts
- Learn problem solving
- Design, implement and test C programs

Previous programming experience is not required

Outline

- Instructors
- Course structure
- Course policies
- Unix Commands
- C basics
- First Lecture Announcements

http://questgarden.com/143/14/4/120417020451/images/teacher.jpg

Instructors

- Professor: Dr. Zhichao Cao
- TA: Manni Liu
- Office Hours
  - Zoom by appointment

Communication

- Communication should be directed through the class forum on Piazza.
- Advantages for Piazza over email:
  - Faster: Both instructors can respond to queries
  - Consistency: Both instructors can see each other's responses
  - Easier: Piazza makes it easier to format questions containing code
  - Useful: We can make good questions visible to the class if others may benefit

Outline

- Instructors
- **Course structure**
- Course policies
- Unix Commands
- C basics
- First Lecture Announcements

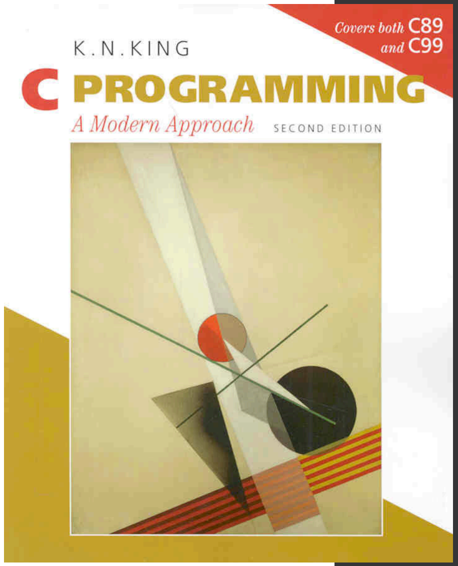
http://questgarden.com/143/14/4/120417020451/images/teacher.jpg

Course Structure

- 2 Lectures (50 minutes)
- 1 Lab (110 minutes)
- Attendance for labs is mandatory and important
- If you miss more than one labs, your final grade will be reduced (more info to follow).

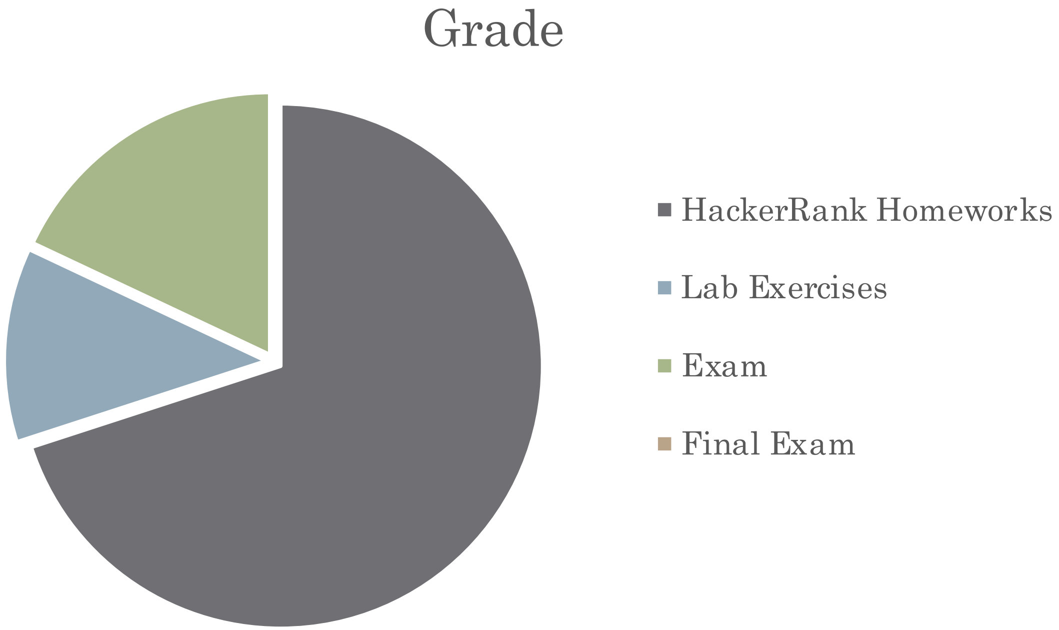
Suggested Course Material

- Textbook: *C Programming a modern approach* 2nd Edition
  - by K. N. King
  - ISBN: 978-0393979503
- Note: This book is not required, all material for this class will be distributed via the class website. However, the book is a good resource regardless.
- All material is on the course website:
  - www.cse.msu.edu/~cse220
  - Class slides will posted on the day of the lecture (but hopefully before).



Grading

- Mimir Homeworks: 70%
- Lab Exercises: 12%
- Exam: 18%
- Final Exam: 0% (No final exam!)



Grade Scale

• 4.0	90% of points available
• 3.5	85% of points available
• 3.0	80% of points available
• 2.5	75% of points available
• 2.0	70% of points available
• 1.5	65% of points available
• 1.0	60% of points available

- *The instructor reserves the right to adjust the scale for course grades, if necessary*

Mimir Homeworks

- There will be at least 12 homeworks assigned.
- Only the top ten scores will be counted in your grade.
- Each assignment will be worth 7 points.
- Homework assignments will be distributed (via emailed links) on Thursday nights.
- Homework is always due on the following Thursday at 10pm.
  - There are no late turn-in. If you haven't submitted the homework by 10pm you will receive a zero for the assignment.

Lab Exercises

- Distributed in lab every week.
- Mimir project.
- Lab exercises are expected to be completed during the lab session.
- There is no turn-in apart from receiving participation credit each session from the TA.
- If you are unable to attend your designated lab session for a particular week, you may attend the other section with permission from the TA.
- There are thirteen lab sessions, but only 12 points will be graded, so you may miss one lab section without repercussion.

Exam

- There is one exam (no final) tentatively scheduled to occur during the lecture of November 9th.
- The exam is worth 18 points (18% of your grade).
- The lecture prior to the exam will be a review session.
- A sample exam containing questions similar in form to the ones on the actual exam will be given out at least one week prior to the exam.
- The details of online exam will be announced later.

Outline

- Instructors
- Course structure
- **Course policies**
- Unix Commands
- C basics
- First Lecture Announcements

http://questgarden.com/143/14/4/120417020451/images/teacher.jpg

Make up policy

- Make-ups for the exam may be arranged:
    - If your absence is caused by documented illness or personal emergency,
    - and written explanation provided to the instructor
- No make-ups for labs (one automatically excused absence)
- No make-ups for homeworks (two automatically dropped assignments)
- If the automatic dropped labs/homeworks are not sufficient, then you should contact me with explanation for the extraordinary circumstances.

Collaboration

- We encourage collaboration on assignments and labs (obviously not during the exam).
- You can work together (in person) on work, however every line of code you submit must be written and understood by you. If you submit work that is not your own and (especially) that you do not understand, you will be reported for academic dishonesty.
- You are allowed to turn in the exact same code as someone else, if and only if, you wrote the code together and you could complete a similar assignment in the future on your own.

Outline

- Instructors
- Course structure
- Course policies
- **Unix Commands**
- C basics
- First Lecture Announcements

http://questgarden.com/143/14/4/120417020451/images/teacher.jpg

Unix Terminal

- Edit, compile and run programs
- Mimir IDE
  - Unix environment is automatically set.
- CSE Computing System
  - Need CSE account
    - Students enrolled in CSE course get a CSE account
  - CSE Facilities: http://www.cse.msu.edu/Facility/
  - CSE hostnames:
    - arctic.cse.msu.edu
    - adriatic.cse.msu.edu

Unix commands

- Filesystem Management

`ls`	List “normal” files.
`ls -a`	List all files.
`rm file`	Remove file.
`rm -r dir`	Remove dir and all subdirs.
`mv file1 file2`	Rename file1 to file2.
`mv dir1 dir2`	Rename dir1 to dir2.
`cp file1 file2`	Copy file1 to file2.
`cd dir`	Change directory to dir.
`cd ..`	Goes to the parent directory
`mkdir dir`	Make a directory dir.
`rm -r dir`	Remove the directory dir
`pwd`	Print working directory

Unix commands

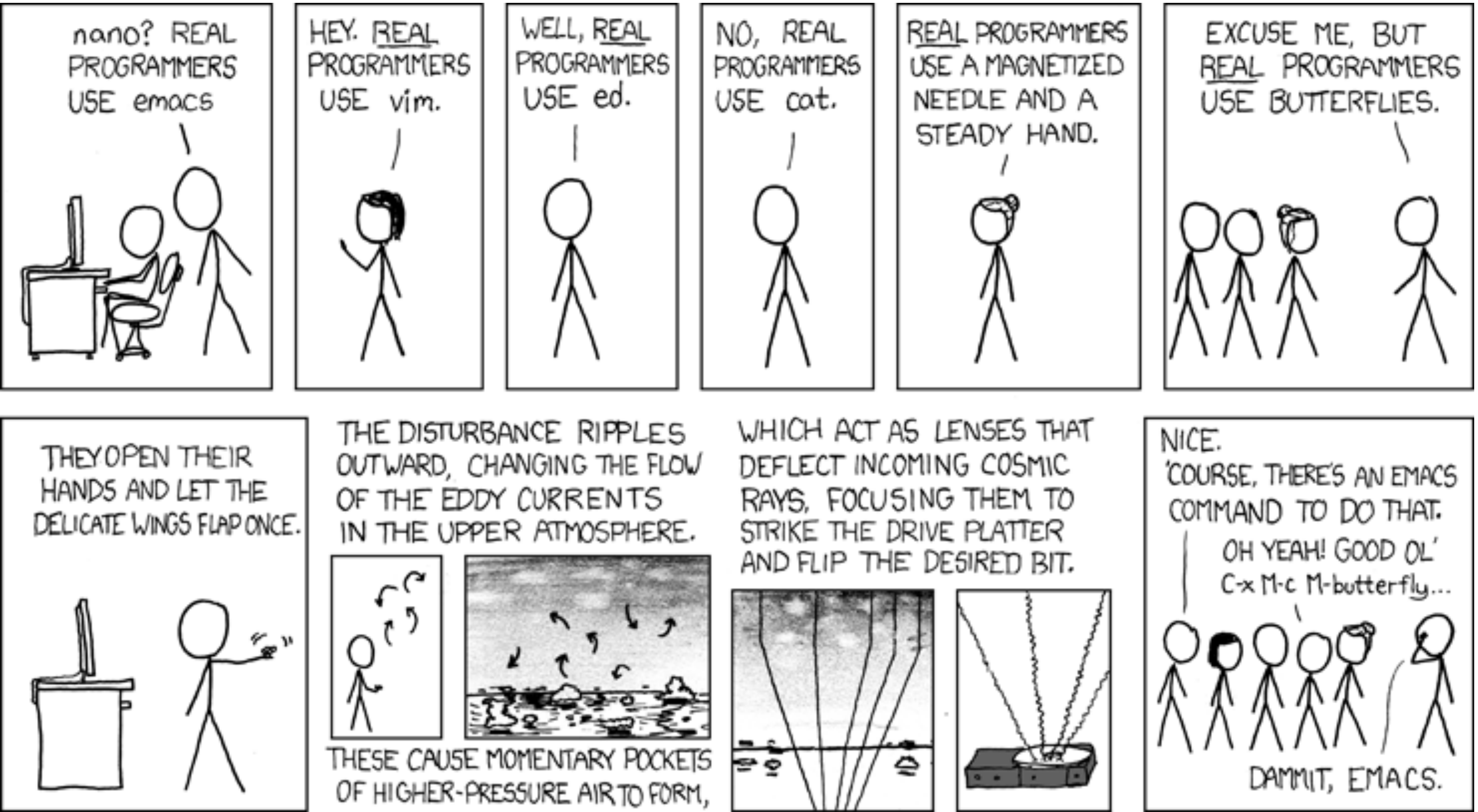
- File Management

`more file`	Show contents of file.
`head -n file`	Show first n lines of file.
`tail -n file`	Show last n lines of file.
`cat file1 file2 > file3`	Append file2 to file1 and save as file3

- Process Management

`top`	Show the top cpu processes.
`ps -elf`	Show info about all processes.
`kill [pid]`	Kill process with the (ex. kill 372).

- A number of editors are available:
  - gedit (this is the one we will use during lab)
  - vi
  - emacs
  - pico
  - kate


nano? REAL PROGRAMMERS USE emacs
HEY. REAL PROGRAMMERS USE vim.
WELL, REAL PROGRAMMERS USE ed.
NO, REAL PROGRAMMERS USE cat.
REAL PROGRAMMERS USE A MAGNETIZED NEEDLE AND A STEADY HAND.
EXCUSE ME, BUT REAL PROGRAMMERS USE BUTTERFLIES.
THEY OPEN THEIR HANDS AND LET THE DELICATE WINGS FLAP ONCE.
THE DISTURBANCE RIPPLES OUTWARD, CHANGING THE FLOW OF THE EDDY CURRENTS IN THE UPPER ATMOSPHERE.
THESE CAUSE MOMENTARY POCKETS OF HIGHER-PRESSURE AIR TO FORM,
WHICH ACT AS LENSES THAT DEFLECT INCOMING COSMIC RAYS, FOCUSING THEM TO STRIKE THE DRIVE PLATTER AND FLIP THE DESIRED BIT.
NICE.
'COURSE, THERE'S AN EMACS COMMAND TO DO THAT.
OH YEAH! GOOD OL' C-x M-c M-butterfly...
DAMMIT, EMACS.

https://imgs.xkcd.com/comics
/real_programmers.png

Outline

- Instructors
- Course structure
- Course policies
- Unix Commands
- **C basics**
- First Lecture Announcements

http://questgarden.com/143/14/4/120417020451/images/teacher.jpg

Why C?

- Base to other languages:
  - C++, Java, C#, Perl, Objective-C
- Low level language:
  - Suitable for system programming
- Small language:
  - Easy to learn
- Powerful language
  - Core of many software/hardware
- Because you are forced to by your degree requirements

https://upload.wikimedia.org/wikipedia/commons/thumb/3/35/The_C_Programming_Language_logo.svg/2000px-The_C_Programming_Language_logo.svg.png

Strengths of C

- Efficiency:
  - C programs run quickly and with limited memory
- Portability:
  - Due to ANSI/ISO standards
- Power:
  - Collection of data types and operators accomplish a lot with few lines of code
  - Bullshit! Most other languages can accomplish more with less
- Standard Library:
  - input/output, string handling, storage allocation, and others

Weaknesses

- Error prone:
  - Flexibility allows programmers to make mistakes without warnings
  - Hard to debug and fix
  - Hard to maintain
- Lacks new and useful features:
  - Object-oriented design
  - Exceptions
  - Testing Harnesses
  - Library Management

Compiling

- Compilation:
  - Checks if you followed the rules of C
  - If no errors => creates an executable
  - If errors found => returns list of errors
- Errors can be cryptic
- Executable
  - The file you run
  - Generated only when all compilation errors are fixed

https://imgs.xkcd.com/comics/compiling.png

Gnu compiler

- A number of compilers available: cc, gcc
- `gcc myProgram.c`
  - Generates an executable called a.out in the same directory
  - Run a.out by typing `./a.out`
- `gcc -o myProgram myProgram.c`
  - Generates executable myProgram

Errors

- Compilation Errors:
  - Program violates C rules

    Statements must end with a semicolon

    x = y + 5

    Compiler will complain about missing ;

- Runtime Errors:
  - Errors that occur during the execution of the program

    Division by zero

    Program will crash during execution

- Logical Errors:
  - Program does not run as desired

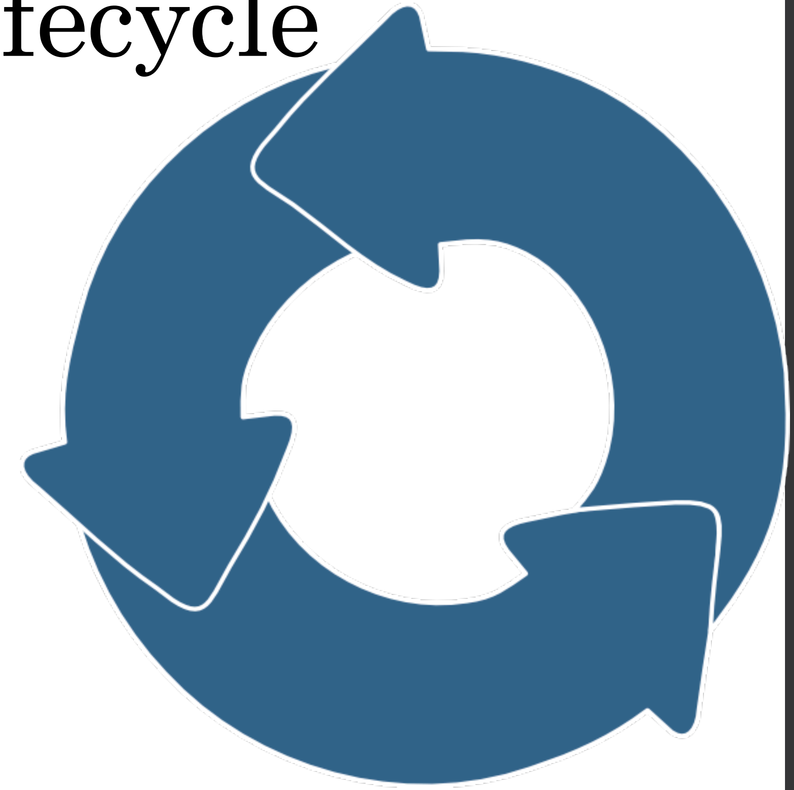
    Compute the area of circle as

    area = 2*3.14*radius

    Program will output incorrect value

Development Lifecycle

- Step 1: Write/Edit a program
- Step 2: Compile
- Step 4: Test
- Step 5: Fix and goto Step 2
- Step 6: Profit???



http://images.clipartpanda.com/cycle-clipart-cycle-hi.png

Tips

- Think and design before you start coding
- Experiment
- Start early
- Ask questions

http://f.tqn.com/y/inventors/1/W/q/9/1/QTips.jpg

Outline

- Instructors
- Course structure
- Course policies
- Unix Commands
- C basics
- **First Lecture Announcements**

http://questgarden.com/143/14/4/120417020451/images/teacher.jpg

Demo Mimir Homework

- Tonight (Sep 2nd, 2020), I'll instruct Mimir to send an email to your MSU email (username@msu.edu).
- This email will contain a link to the assignment.
- Complete the assignment (it isn't worth any points) before the next Thursday 10pm.
  - All assignments are due on Thursday at 10pm
- On the next Friday, I'll test the grading software (you should receive an email with the correct answers).
- As always, I'll keep the class up-to-date via Piazza, so you should register ASAP.

First Week Special

- No labs
  - So don’t connect to lab Zoom Meeting (Friday 04/09)!!!
- Don't forget about Labor Day (Monday). No classes.