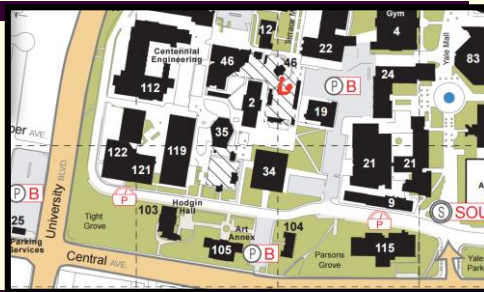


Welcome to CS 241 Data Organization using C



Instructor: **Joel Castellanos**
e-mail: joel@unm.edu
Web: <http://cs.unm.edu/~joel/>
Office: Farris Engineering Center
(FEC bldg. #119)
Room 2110



8/21/2019

1

CS-241 Course Description & Goals

Introduction to the C Programming language

*****NOT an introduction to programming*****

Students taking this course should already be familiar with **variables, conditional control flow, loops** and have a solid ability to apply **Algebra I**.

Intro to Linux operating system, command-line interface and command-line programming.

Intro to machine level data organization and memory allocation.

Intro to (short and small) to GPU (parallel) programming.

Intermediate use of loops and control flow.

2 Intermediate logic and algorithms.

2

Computer Science Course Sequence

- 1 CS-105: Intro programming JavaScript and HTML5
 - 2 CS-152: Intro programming Java
 - 3 CS-251: Intermediate programming Java
 - 3 CS-261: Mathematical Foundations of Computer Science
 - 4 CS-241: Data Organization using C
- CS-341: Intro. to Computer Architecture and Organization
CS 357: Declarative Programming
CS-351: Design of Large Programs.
CS-361 & CS-362: Algorithms I & II

3

3

Quiz: Write a function in Java, C, C++, JavaScript, MATLAB or Python that...

- 1) Takes no arguments.
- 2) Does not return a value.
- 3) Uses **a nested loop** to display a 15 x 15 multiplication table.
- 4) The multiples of 1 (1 through 15) must print on the first line with at least one space between each number.
- 5) The multiples of 2 (2, 4, 6, ... 30) must print on the second line with at least one space between each number.
- 6) The same must be true for each of the other multiples.

4

4

Display 15 x 15 multiplication table

- No need for "if" statements.
- No need for modulus (%).
- No need for more than a 10 lines of code.
- No need for arrays.

5

5

Output of C program on cs.unm.edu

```
moons.cs.unm.edu - PuTTY
joe1@phobos:~/cs241$
joe1@phobos:~/cs241$
joe1@phobos:~/cs241$
joe1@phobos:~/cs241$
joe1@phobos:~/cs241$
joe1@phobos:~/cs241$
joe1@phobos:~/cs241$ gcc foo.c
joe1@phobos:~/cs241$ a.out
1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
2  4  6  8 10 12 14 16 18 20 22 24 26 28 30
3  6  9 12 15 18 21 24 27 30 33 36 39 42 45
4  8 12 16 20 24 28 32 36 40 44 48 52 56 60
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75
6 12 18 24 30 36 42 48 54 60 66 72 78 84 90
7 14 21 28 35 42 49 56 63 70 77 84 91 98 105
8 16 24 32 40 48 56 64 72 80 88 96 104 112 120
9 18 27 36 45 54 63 72 81 90 99 108 117 126 135
10 20 30 40 50 60 70 80 90 100 110 120 130 140 150
11 22 33 44 55 66 77 88 99 110 121 132 143 154 165
12 24 36 48 60 72 84 96 108 120 132 144 156 168 180
13 26 39 52 65 78 91 104 117 130 143 156 169 182 195
14 28 42 56 70 84 98 112 126 140 154 168 182 196 210
15 30 45 60 75 90 105 120 135 150 165 180 195 210 225
joe1@phobos:~/cs241$
```

i (vertical arrow pointing to row index)

*i*k* (diagonal arrow pointing to value 77)

k (horizontal arrow pointing to column index)

6

6

Code for 15 x 15 multiplication table

```
#include <stdio.h>

void main(void)
{
    for (int i=1; i<=15; i++)
    {
        for (int k=1; k<=15; k++)
        {
            printf("%3d ", i*k);
        }
        printf("\n");
    }
}
```

7

7

Pretest Results

G (good): Perfect or at most minor errors.

OK: At least one significant error, but shows the basic idea.

U (unprepared): Withdraw yourself, see me with a good reason for under performance or **you will be withdrawn** by Thursday, Aug 29.

0: Did not take the pretest or forgot your name or I could not read your name. See me to figure out which paper is yours or set a time for a make up or **you will be withdrawn** by Thursday, Aug 29.

8

8

Pretest Results: Score of Unprepared

- “Unprepared” does not mean you are being “weeded out”.
- None of you are “weeds”.
- A very big part of success in college is taking the right class in the right sequence with the right preparation.
- Taking a course you are not ready for does not help you graduate: It wastes time. It wastes money. It destroys self-esteem.

9

9

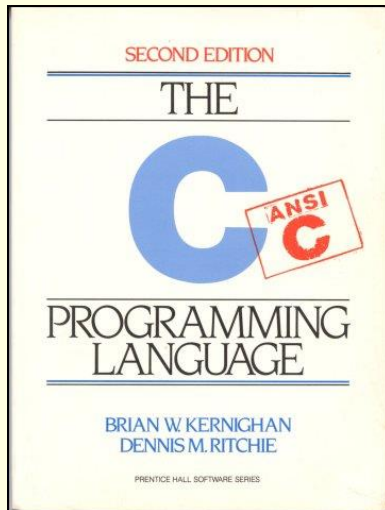
Unprepared Possibilities

- If you are a transfer student trying to align your past classes with UNM classes, then, please, meet with me very soon. I want to take a detailed look at your background and skills and figure out where the best place is for you at UNM (which may be in CS-241 with a bit of extra catch up work).
- If you have not yet completed CS-152 and CS-251, then I will not listen to any excuses: the designed order of taking these classes is 151 → 251 → 241. This score of U is just one bit of evidence to support that.
- If you have taken CS-152 and CS-251 and have gotten at least a B- in the two courses, then let's talk, figure out what happened and what is the best path forward. Maybe you just froze up and will be fine. Maybe stay in 241, but with plans for MUCH TIME of homework and self-study. Maybe take an online course,

10

10

Course Resources



11

Class website:

<http://cs.unm.edu/~joel/cs241/>

- Syllabus
- Projects
- Lecture Notes
- Supplemental readings
- Videos

Blackboard Learn:

- Assignment Drop-box
- Assignment discussions
- Grades

11

CS-241 Course Grading

- 40% Programming Projects.
- 30% Exams (midterm and final)
- 20% Laboratory Programming Assignments.
 - Attendance is required. After missing two labs, -2 points per missed lab from final course grade.
 - The minimum grade on a lab turned in with reasonable effort will be a 10/20 if the student attended the associated lab class. Does not apply to projects.
- 10% Lecture quizzes
 - Approximately 30, i-clicker, attendance required.

12

12

Turning In Assignments

- All assignments must be submitted into Blackboard Learn.
- e-mailing code is useful for getting help with debugging, **but an assignment turned in by e-mail counts for nothing** 😞.
- Late projects/assignments receive a 5% per day penalty.
- Assignments are not accepted after two class periods past the due date (since solutions will be posted and explained).
- An assignment is “**turned in**” on the Blackboard Learn date stamp of the **final version** submitted.
- Every project will have extra-credit options that can be used to cover missed work - or to boost your grade.
- If you do extra stuff that you think is cool, **show me!**

13

13

Assignment Grading: 1 of 2

- In general, assignments are graded by your lab instructor.
- When you have a question about how something was graded, **first**, ask the person who graded it.
- The lab instructors must follow strict grading rubrics to score your assignments.
- Sometimes the rubrics miss-fire and cause more points to be taken off than should be for an error I did not think of. Such miss-fires need to be brought to my attention so that I can let all the graders know of any change for uniform grading.

14

14

Assignment Grading: 2 of 2

Work the system:

- Assignments have grading rubrics and are graded by compiling them and running them on given data files.
- If your program does not compile or expects the wrong number of spaces or commas, then your program will miss most of the points on the rubric.
- Therefore, you get a much higher grade if you turn in a program that works perfectly with some of the input data than a program that works almost correctly with all of the input..
- Some of the tests are easy to pass and some are hard. Some tests just test different requirements.

15

15



- We will use i-clicker for quizzes in lecture only.
- We will use i-clicker every lecture.
- If you have already registered your i-clicker on the web for a different class **this semester**, then you do not need to register again.
- Register your iclicker in Blackboard Learn under CS-241.
- Some classes may require you to register in a different way or with a different ID number. If so, register again for this class in this way.
- One i-clicker can be registered to more than one person (as long as no two of them are in the same i-clicker class at the same time).

16

16



Quiz Question #1:

Do you have your i-clicker?

- a) Yes – I am ready to go.
- b) I bought one from the bookstore, but forgot it.
- c) My dog ate it.
- d) No – I did not get one yet.
- e) What is an i-clicker anyway?

17

17

Forgotten i-clicker?

- If you forget your i-clicker, you may borrow one of my loner i-clickers.
- Each loner i-clicker has an animal picture.
- If you borrow an i-clicker, then to get credit, you must ***e-mail me on the same day.***
 - **Subject:** CS-241 borrowed i-clicker
 - **Body:** Name, Date, and Animal.
- There is a limit to how often an i-clicker can be borrowed.

18

18

Working Together but do not Cheat

- Working together and helping one another on all projects is highly encouraged. This includes discussion of project:
 - *specification*,
 - *algorithms*,
 - *data structures*,
 - and *test cases*.
- Do *not* share code.
- It is considered cheating to leave your code (paper or electronic copies) where others can find it. You responsible for the security of your intellectual property.

19

19

Computer Access

- Need to work on a CS Linux machine.
- See: [Instructions for obtaining a CS account and its Terms of Service \(TOS\)](#) on the class website.
- Use Putty (or some other SSH) to connect:
`moons.cs.unm.edu`
- Putty is free and runs from a single executable.
- Therefore, in addition to your UNM account, you will need a cs.unm.edu computer account.
- With a CS computer account you can access
* `.cs.unm.edu` and use the CS Linux lab on the third floor of FEC.

20

20

Programming vs Natural Language

- The entire C vocabulary consists under 40 reserved words.
- There are many libraries, such as `math` and `stdio`. However, these are the *proper nouns* of the language.
- A person can be fluent in a language without knowing the vast majority of its proper nouns.
- Proper nouns are learned as needed.
- Like natural languages, programming languages have punctuation and syntax rules (e.g. In C, every statement is ended with a semicolon). Programming languages, however, have much fewer rules than natural languages.

21

21

Small Language with Complex Usage

- Programming Languages are much smaller and easier to learn than natural languages.
- However, programming languages are primarily used to express complex branchings of conditional logic that far surpass common uses of natural languages.
- Logic skills have strong carryover from one programming language to another.



22

22

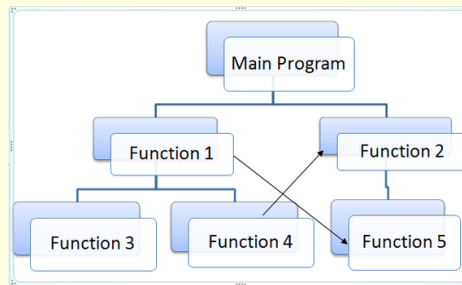
The C language is:

Strongly Typed: Identifiers must have strictly defined before used.

Imperative: Statements change the program's state.

Structured: Makes extensive use of block and loop structures - in contrast to using simple tests and jumps such as the goto statement.

Procedural: Based upon the concept of breaking a complex program into multiple, named, functional units call procedures, routines, subroutines, or functions.



23

23

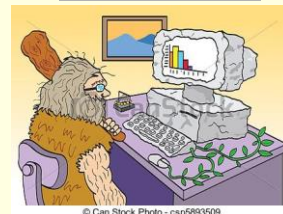
C and Command line: *Old & Outdated?*

Embedded Systems and Firmware:

C is very lightweight and much easier to port than C++ or Java.

Compilers and Device Drivers:

- C is "close to the machine".
- Need to control every bit of every byte.
- Need to specifically use a particular level of cache, main memory, graphics memory,
- Need to do a "hard write" verses "buffered write".



Command-Line interface:

- Very small computers Supercomputers and remote (space) computers and not have IDEs or graphical interfaces.

24

24