

## SCIENTIFIC COMPUTING – ACMS20210

### Course Summary

---

An introduction to solving mathematical problems using computer programming in high-level languages such as C.

PREREQUISITES – Math 10560 -or- Math 10860 -or- Math 10360

BOOK – Programming Principles and Practice Using C++, by Stroustrup. Second edition. 2014.

TIME AND PLACE – MWF 2-2:50pm, DeBartolo Hall 136.

INSTRUCTOR – Daniel Brake. Office 146 Hayes-Healey. [dbrake@nd.edu](mailto:dbrake@nd.edu)

OFFICE HOURS – Mondays 12-1, Wednesdays 3-4, and by appointment. Subject to revision.

### Tutorial

---

ASSISTANT – Alan Liddell. Office Hayes-Healy B02. [liddell.10@nd.edu](mailto:liddell.10@nd.edu)

ALAN'S OFFICE HOURS – Mondays 10-11, Fridays 10-11, and by appointment. Also subject to revision (I will try to give 24 hours' notice).

MEETING TIMES – Thursdays, 12:50-1:40 (Hayes-Healey 231), 2-2:50 (DeBartolo 201)

### Grade

---

Lab	100
Homework	100
Project	100
Exams	100
Total	400

LAB – Weekly worksheets helping you focus on particular aspects of computing. **No late submissions accepted.** Submission method will vary by assignment. These are created, graded, and managed by the TA. The number of points on homework assignments will probably not sum to 100. Rather, you will be given a number of points toward your final score, which are the percentage you earned out of the total number available.

HOMEWORK – Assigned weekly, and turned in by Sakai, generally on Monday afternoons. Points scaled to 100. See the POLICIES section for information about group collaboration. Problems will be a combination coming from the book, and those which the instructor makes up. **No late submissions accepted.**

PROJECT – A term project, which will allow you to explore some facet of computing which interests you. It will consist of a topic proposal, two rough drafts, a final code deliverable, a final project writeup, and a short presentation. The project will be largely graded by your peers.

EXAMS – Two midterms, each 50 points. The questions are open ended, and challenging. Dates appear in schedule on page 3.

LETTER GRADE – Will be assigned on the percentage of points you earned, out of the total available, with a scale no more stringent than the typical ND 90-80-70 scale, including  $\pm$  cutoffs.

## Policies

---

GROUP WORK – Group work to solve homework problems is acceptable. However, **you must write your own programs and type your own solutions**. Copy-paste submission of group work is NOT acceptable, and is a violation of the ND Honor Code, even if you helped write the code in the first place. **You super duper seriously must type your own code**. *Alan adds: There are tools which make it really easy to find out who's submitted the same code and the TA will use them. Don't be that person.*

ATTENDANCE – You are expected to attend every lecture and tutorial session in which you are enrolled. If you must miss a lecture for a valid reason, it is your responsibility to obtain notes from a reliable classmate. Attendance will be taken randomly. Five (5) unexcused no-shows yields a reduction by one full letter grade, e.g. from B+ to C+.

HONOR CODE – The ND Honor Code is in full effect for all work related to this course. The instructor takes this seriously. You may not submit code from online or any other source as your own, including group work. Any code you submit must have been typed in by you, personally, physically, using your own fingers on a keyboard. Submission of code developed as a group without typing it yourself, is a violation of the honor code. You must type your own code. **You must type your own code.**

ON LEARNING C++ – The C++ language is *very* hard to master, and just as difficult to get started with. You will be challenged. Simple tasks can be confusing. Compiler errors can be opaque, and runtime bugs mysterious. To overcome this barrier, you must personally engage in the struggle that is programming. Class sessions cannot make you a good programmer any more than a band teacher can make you a good violinist. Practice and struggle with instruments and tools are what make a person great, not being a passive observer of instruction.

Do not expect to be able to sit down and solve homework problems quickly, even simple ones, merely because you came to class, read the class notes, or saw someone else solve the problem. Classes and slides will be insufficient for completing homework; you will have to get better at independent research, in the book and online. Furthermore, only by starting homeworks soon after their assignment, and allowing yourself plenty of time to complete them, can you hope to succeed in this class.

While the instructor is sympathetic to the challenges associated with learning a compiled programming language, especially as a first exposure to programming, the instructor is not sympathetic to low or late effort. Be active in your learning – ask questions, will to succeed despite repeated failure, and be perseverant.

LATE WORK – **Late work is not accepted**, except for pre-arranged university excused absences.

## Detailed Course Syllabus

---

We plan on covering chapters 1-10, 17, 21, 24. Depending on course pace, this may vary some. We will not cover the graphics part.

week	topic	chapters	notes
1	CRC usage, Hello World!	2	
2	Objects, Types, and Values, Computation	3, 4	do the homework or struggle through the entire class
3	Computation, Errors	4, 5	
4	Errors, Writing a Program	5, 6	
5	Completing a Program	7	Proj. intro
6	Technicalities - Functions	8	
7	Technicalities - Classes	9	
8	Exam 1; Projects		Proposal
Oct 17-21 – Fall Break			
9	Input and Output Streams	10	
10	Vector and Free Store	17	Draft 1
11	Vector and Free Store, Algorithms and Maps	17, 21	
12	Algorithms and Maps	21	Draft 2
13	Exam 2      Nov 23-25 – Thx Break		
14	Numerics	24	
15	Project Presentations		Presentation
Final Time, Thursday, Dec 15, 4:15-6:15 pm			Presentation, Project due

If there are topics which you would like to cover in particular depth, or what you want to learn isn't on this topic list, please ask! You can help determine the focus of the class!