# CS 559 Advance Topics in Computer Science

## SPRING 2016

## SYLLABUS AND GENERAL INFORMATION

**CONTENTS**

[GENERAL INFORMATION](#h.1fob9te)

[CONTACT](#h.3znysh7)

[LECTURE, EXAM, OFFICE HOURS](#h.2et92p0)

[PREREQUISITES](#h.tyjcwt)

[REQUIRED TEXT](#h.3dy6vkm)

[COURSE ANNOUNCEMENTS](#h.4d34og8)

[CLASSROOM CONDUCT](#h.2s8eyo1)

[PURPOSE AND FOCUS OF COURSE](#h.17dp8vu)

[GRADING, ASSIGNMENTS, AND EXPECTATIONS](#h.3rdcrjn)

[PROJECT](#h.26in1rg)

[HOMEWORK](#h.lnxbz9)

[EXAMS](#h.35nkun2)

[CLASS ATTENDANCE / PARTICIPATION](#h.1ksv4uv)

[GRADE CUTOFFS](#h.44sinio)

[GRADING PROGRAMS](#h.z337ya)

[BLACKBOARD](#h.3j2qqm3)

[ACADEMIC INTEGRITY](#h.1y810tw)

[RECOMMENDED BLACKBOARD/ SYLLABUS STATEMENT REGARDING STUDENT DISCLOSURES OF SEXUAL MISCONDUCT](#_RECOMMENDED_BLACKBOARD/_SYLLABUS)

[SPECIAL NEEDS](#h.4i7ojhp)

# GENERAL INFORMATION

## CONTACT

**Name*:*** Chinmai Basavaraj

**Email:**[*Chinmai.Basavaraj@indstate.edu*](mailto:Chinmai.Basavaraj@indstate.edu)

**Office:**G104 B, Gillum Hall

**Contact:** *812 237 7914*

# LECTURE, EXAM, OFFICE HOURS

***Lecture:*** Monday, Wednesday and Friday from 12**:00 pm- 12:50 pm** in **Root Hall, Room A017** (one of the large lecture rooms in the basement, all the way to the left).

***Website:*** <http://cs.indstate.edu/~cbasavaraj/cs559.html> (Work in Progress)

***Exam:*** Final Exam for this course will be on **May 2nd, Tuesday 1:00 pm**.

***Instructor Office Hours:*** I am available weekdays 8:30 am to 4:30 pm in my office, Gillum Hall G104B.

***GA Tutoring:*** We have a few graduate assistants who will be in the computer science UNIX lab, room A-015 in the basement of Root Hall, for about 20 hours per week in total. You can go to this lab to work on your programs. The computers are UNIX machines, and you can use the cs559xx login that will be sent to you during the first week of class to use them.

Or, you can bring your laptop to work on. Either way, you can ask the graduate assistants to look at your programs, or you can work with any other CS 559 students that are there (you could use the lab as a regular meeting place to work with your classmates). The regular hours that the lab will be open will be posted to [**here on the department's website**](http://mathcs.indstate.edu/dept/academic/labtimes.php).

# PREREQUISITES

There is no course prerequisite for this class. Things I assume you already know from previous courses or experience: basic computer use (web browsers, email, installing programs on your computer), basic algebra and math. For the math, I assume you can do things like solve an equation for an unknown, or draw a graph/plot of a function such as y = x2. We will review any other math that may be needed (which will not be much).

# REQUIRED TEXT

90% of the course we will cover Programming in C. During the later stages we may do some graphics programming or computer vision.

We will refer to text and material freely available online. For C, We will follow the text [The C Book](http://publications.gbdirect.co.uk/c_book/), second edition by Mike Banahan, Declan Brady and Mark Doran. The text is freely available online from that link, or you can buy a physical copy of the book.Additional readings may be assigned from other online sources.

### More Information about C

* [C Programming](https://en.wikibooks.org/wiki/C_Programming)
* [How to Think Like a Computer Scientist, C++ Version](http://www.greenteapress.com/thinkcpp/)

There are many online resources that you could find for learning to program with C. You could search on youtube for videos, search the web, etc. You could look for other C courses that have their information online, for example [this one from MIT](http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-189-a-gentle-introduction-to-programming-using-python-january-iap-2011/), or [this one from MIT that has videos](http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-00-introduction-to-computer-science-and-programming-fall-2008/index.htm).

# COURSE ANNOUNCEMENTS

Announcements regarding the course will be made both during class and via email to your @sycamores.indstate.edu email address. You should regularly check this email account or have it forwarded to an account that you check regularly. You can set the account to forward by logging into your indstate.edu email from Internet Explorer (the "light" version of the webmail client that opens up from Firefox or Chrome does not give the option to forward email).

# CLASSROOM CONDUCT

You may not use cell phones, iPods/music players, etc. during class. You should be civil and respectful to both the instructor and your classmates, and you should arrive to class a few minutes before the scheduled lecture so you are ready for lecture to begin on time. You may use your computer during class if you are using it to follow along with the examples that are being discussed. You may not check email, Facebook, work on other courses, etc. during class.

# PURPOSE AND FOCUS OF COURSE

This course is an introduction to programming in C. After taking this course (if you put in the time and earn a good grade), you should be able to write programs and solve problems using C, and you should be able to read, understand, and modify programs written by others.

#### Why C?

We have chosen C as the programming language for this course because it is the most basic. Learning C will lay a solid foundation which will help you to grow as a programmer. You can easily learn and adapt to any language once you learn how to go things in C. In your coursework at Indiana State University, you are required to know C.

#### Example topics

In addition to becoming familiar with C programming, and certain computer science ideas, I hope you have fun. Here are some examples of topics we may cover and programs you will personally create!

* Games. We will make a few simple games. For a few of these (e.g., tic-tac-toe), we will write a computer to have the computer play the game optimally. Some potential games: tic-tac-toe, lower/higher, typing game, arithmetic game, checkers ...
* Numbers. Factoring, multiplying, solving equations, sorting ... All these things are very important things computers do for us, and we will see how some of them work.
* Data. How to store data, keep it organized, update it, ...
* Internet/networking/messaging. We can make our own program to process web-pages, make a chat program, make our games internet-enabled, ...
* Animations. We can make cool animations - planetary simulation, fireworks, simple games, ...
* Pictures. We might make some of the functions that are in programs like Adobe Photoshop: grayscale, shading, zooming, rotation, ... Did you ever wonder how these are actually done in Photoshop?
* Audio. We might make some of the functions that are provided by sound-editing programs: making echoes, changing the volume, splicing/clipping (make someone say whatever you want them to!), making music, ...
* Video. Similar to pictures and audio, we can make some simple video editing functions like fading, splicing, stick-figure animations, ...
* Drawing fractals.
* Plotting graphs (make-your-own-graphing-calculator!).
* Make-your-own-paint program.
* Encrypting messages so only you and your secret club can read the messages.
* Financial calculator.
* Computer science. The big goal of computer science is to come up with the best possible computer systems and programs to solve a program. If you want to sort a list of numbers, what is the fastest way to do this? How fast can you factor a 1000-digit number (if you can do it fast enough, you can break the encryption that is used on the internet)? Did you know that there are some problems that no computer, no matter how "super" can solve (e.g., the halting problem)? We will begin to look at how to answer these types of questions, though you'll have to take later CS courses (CS 201, 202 ...) to get into more depth.

We will not cover all of those topics, but that should give you an idea of the types of things we will look at. Hopefully there are some things there that sound exciting!

# GRADING, ASSIGNMENTS, AND EXPECTATIONS

The students of this course have the following responsibilities: read assigned readings before lecture, attend lecture, complete homework assignments, take exams, and complete a project. The final grade consists of:

### FINAL PROJECT

1**0%** of the final grade.

### HOMEWORK

There will be a programming homework assignment most weeks. The total of all homework assignments is worth **40%** of the final grade. If you turn in the assignment late, I will grade it so you know how you did, but it will count as a 0. LATE HOMEWORKS WILL NOT BE ACCEPTED.

NOTE: if you turn an assignment in more than one day early, I will look at it and give you feedback on any mistakes you might have - and if you resubmit before the due date I will count the grade for the corrected version.

We may have in-class quizzes from time to time; these would be added into the homework grade.

### EXAMS AND QUIZ

30% total. We will have three exams in total.

# CLASS ATTENDANCE / PARTICIPATION

20% total. Attendance will be taken at the beginning of each class. Half of your attendance/participation score will consist solely of whether you were present when attendance was taken each day - the total number of days present divided by the number of lectures in the semester. The other half of your attendance/participation grade will be assigned at the end of the semester based on how attentive you were in class throughout the semester.

The class attendance grade serves the purpose of giving you credit for coming to class. You benefit from coming to class by seeing me present the new material, getting to ask questions, interacting with your classmates, keeping up on what is going on in the course, etc.

The exams serve as benchmarks of your ability to relatively quickly solve problems related to the material. This helps me assign a grade, and also gives you motivation to pay attention and keep up with the assignments.

The weekly programming homework assignments are designed to solidify your knowledge by having you write programs.

# GRADE CUTOFFS

#### GRADING SCALE

|  |  |
| --- | --- |
| Percentage | Minimum Grade |
| 90 and up | A |
| 85-89 | A - |
| 80-84 | B + |
| 70-79 | B |
| 65-69 | B - |
| 60-64 | C + |
| 50-59 | C |
| 45-49 | C - |
| 40-44 | D + |
| 35-39 | D |
| 30-34 | D - |
| 29 and below | F |

I make no promise ahead of time what the exact cutoff will be in terms of the number of points to achieve an A+, A, A-, etc. These will depend on how the course goes. I will use the guidelines below in assigning letter grades. After the first few weeks, I will include a "letter grade if the semester ended today" in your grades and for taking attendance. You can keep track of how you are doing in the course with the grades on the [blackboard](http://cs.indstate.edu/~jkinne/cs151-f2012/syllabus.html#blackboard) site for this course.

The following is roughly what I would expect by the end of the semester to earn a particular grade. This may not mean too much to you now, so you may want to check this again a few weeks into the semester.

* **A+/A** Superior. You have a good understanding of all of the concepts. You can write programs from scratch on your own by the end, and can trace through what even modestly complicated programs do just by working it out on paper. You have a good understanding of all the aspects of the C language we talk about in class. An A+/A student should be able to continue learning about C (and other programming languages) on their own after completing the course.
* **A-/B+** Excellent. You understood most of the concepts, but may have one or two gaps. Still, you can normally write your own program from scratch and follow along in most of the program examples from class. But you may need a slight bit of guidance getting started working on a program. You also could probably keep learning more C and programming on your own, though it will probably take you more effort.
* **B/B-** Good. You have done well on some assignments but not as well on others. You may have trouble getting started on a program without guidance, but given hints and an outline you can normally write a correct program. Likewise with understanding a program given to you, you may need some help in understanding it. You would likely have difficulty learning more programming after the course on your own; you should be fine to keep learning in a classroom setting but should make sure to get a good start on your next CS/programming course.
* **C+/C/C-** Adequate. You have basic understanding of C programming but have modest gaps in your knowledge of the language syntax and how to use it effectively. You may have difficulty tracing through even a simple program that you have not seen before, but you are able to trace through a program that is similar to one you have seen before. You are minimally prepared to take another course in CS/programming. You should definitely consult the instructor of the next course you take about whether you are prepared to take the course, and you should consider repeating some of the homework assignments to continue to improve your understanding.
* **D+/D/D-** Poor. You have some shown some ability to write programs and trace through code given to you, but you are often unable to write a program from scratch and produce programs that are not correct. You are not prepared to take another CS/programming course. You should retake the course.
* **F** Failing. For much of the material in the course, you do not even know the basic information or have not turned in assignments. Most of your programs probably do not run at all, let alone correctly.

#### GRADING PROGRAMS

For the later programming assignments, I will assign a grade based on correctness *and* style. The exact breakdown may vary slightly from one assignment to another. In general, 60-70% of the points for an assignment will be given based on whether it is correct. The remaining 30-40% will be given based on good programming style:

1. Choosing variable and function names that are descriptive/appropriate
2. Writing code that is easy to understand and efficient
3. Including documentation at the top of each file about what is in that file and how to use it
4. Including documentation with each function describing what the function does (including what should be input to the function and what the function outputs)
5. Documentation throughout each function describing the flow of the program

For portions of the code that are given to you, you do *not* need to add documentation to those parts of the code; if I give you partially completed code, you are responsible only for documenting the code that you add.

# BLACKBOARD

The course has a blackboard site. Click [here](http://blackboard.indstate.edu/) to go to blackboard. You should see CS 559 listed under your courses for the current term. The blackboard site is used for giving you your grades. The blackboard site can also be used to post questions/comments about the class. You have the option of posting your questions in blackboard or just sending email. If you do not want your question visible to the rest of the class, then send email. It could be good to post the question on blackboard because (1) other students may be having the same problem and could benefit from seeing the answer to your question, (2) other students can respond to your question/comment - and may do so sooner than I do.

Other than the course grades and discussion board on blackboard, all course content, schedule, etc. is kept on the instructor's webpage (which you are currently viewing).

# ACADEMIC INTEGRITY

Please follow these guidelines to avoid problems with academic misconduct in this course:

* Homework’s: You may discuss the homework assignments, but should solve them on your own. To make sure you are not violating this, if you discuss with someone, you should DESTROY any work or evidence of the discussion, go your separate ways, SPEND at least an hour doing something completely unrelated to the assignment, and then you should be able to RECREATE the program/solution on your own, then turn that in. If you cannot recreate the solution on your own, then it is not your work, and you should not turn it in.

Note on sources: if you use some other source, the web or whatever, you better cite it! Not doing so is plagiarism.

* Exams: This should be clear - no cheating during exams. The exams will be closed-book, closed-notes, no computer, and no calculator.
* Projects: You should not copy code from the internet or anywhere else. The project should be your own work. It will be fairly obvious to me if you do copy code from the internet, and the consequences will be at the least a 0 on the project.

If cheating is observed, you will at the least receive a 0 for the assignment (and may receive an F for the course), and I will file a Notification of Academic Integrity Violation Report with Student Judicial Programs, as required by the university's policy on Academic Integrity. A student who is caught cheating twice (whether in a single course or different courses) is likely to be brought before the All-University Court hearing panel, which can impose sanctions up to and including suspension/expulsion. See the [Student Code of Conduct](http://www.indstate.edu/sjp/docs/code.pdf) and [Academic Integrity Resources](http://www.indstate.edu/academicintegrity/) for more information.

Please ask the instructor if you have doubts about what is considered cheating in this course.

# RECOMMENDED BLACKBOARD/ SYLLABUS STATEMENT REGARDING STUDENT DISCLOSURES OF SEXUAL MISCONDUCT

Indiana State University fosters a campus free of sexual misconduct including sexual harassment, sexual violence, intimate partner violence, and stalking and/or any form of sex or gender discrimination. If you disclose a potential violation of the sexual misconduct policy I will need to notify the Title IX Coordinator. Students who have experienced sexual misconduct are encouraged to contact confidential resources listed below. To make a report to the Title IX Coordinator, visit the Equal Opportunity and Title IX website: <http://www.indstate.edu/equalopportunity-titleix/titleix>.

Confidential Resources:

The ISU Student Counseling Center – HMSU 7th Floor; 812-237-3939; [www.indstate.edu/cns](http://www.indstate.edu/cns)

The ISU Victim Advocate – Leah Reynolds; HMSU Room 813; 812-237-3829(office); 812-243-7272; [leah.reynolds@indstate.edu](mailto:leah.reynolds@indstate.edu)

Campus Ministries - <http://www2.indstate.edu/sao/campusinistries.htm>;

For more information on your rights and available resources: <http://www.indstate.edu/equalopportunity-titleix/titleix>

*\*Optional additional statement – Please see me if you want more information about the confidential resources.*

# SPECIAL NEEDS

If you have special needs for the classroom environment, homework’s, or quizzes, please inform the instructor during the first week of classes. If you have any such needs, you should go to the Student Academic Services Center to coordinate this. See [Student Academic Services Center - Disabled Student Services](http://www.indstate.edu/sasc/programs/dss/services.htm) for more information.