CS 1325 Course Syllabus

COURSE INFORMATION:

Course Title: Introduction to Programming
Section: 002
Course Term: Fall 2022

INSTRUCTOR CONTACT INFORMATION:

Name: Brian Ricks
Office: ECSS 4.701
Email Address: bwr031000@utdallas.edu
Phone: (972) 883-2674

Communication: For course related questions, please email me from your UTD account with a subject that begins with: *CS1325.002*. There is also an MS Teams channel which can be used for general course questions / discussion. I will give course announcements on eLearning.

Office Hours: In *MS Teams* as follows:
Monday: 1pm – 2:30pm
Tuesday: 10:30am – 11:30am
Wednesday: 1pm – 2:30pm
Other times and in-person meetings by appointment.

Do not procrastinate. I will not schedule meetings to assist students with assignments that are due within the next 24 hours.

TA CONTACT INFORMATION:

Kapilesh Yadav Bongu
Email: kxb210026@utdallas.edu

Office Hours:
Monday: 11:30am - 12:45pm (in-person - Lab ECSS 2.104A1)
Tuesday: 10:00am – 12:00pm (MS Teams)

COURSE PREREQUISITES AND COREQUISITES:

Prerequisite or Corequisite: MATH 2413 or MATH 2417

COURSE DESCRIPTION:

Computer programming in a high-level, block structured language. Basic data types and variables, memory usage, control structures, functions/procedures and parameter passing, recursion, input/output. Language used is C. Programming assignments related to engineering applications, numerical methods.

This course may *not* be used to satisfy degree requirements for majors in Computer Engineering, Computer Science, and Software Engineering. Prerequisite or Corequisite: MATH 2413 or MATH 2417 (3-0) S

STUDENT LEARNING OBJECTIVES/OUTCOMES:

1. Ability to use fundamental programming constructs: assignment, loops, conditions
2. Ability to process data in arrays
3. Ability to develop programs in a functional form
4. Ability to perform sequential file input and output
5. Ability to express algorithms that solve elementary engineering and scientific problems

---

COURSE MODALITY:

This course is scheduled to be taught in the traditional (in-person) mode.

Should it become necessary to meet online, meetings will be held in MS Teams in the appropriate lecture channel. Check the course announcements and your UTD email account for updates.

Online meetings will be recorded and posted in MS Streams.

REQUIRED TEXTBOOKS AND MATERIALS:

Required Textbook:
P. Deitel and H. Deitel, C: How to Program, 8th Edition, 2015
ISBN: 978-0-13-397689-2

See the TENTATIVE COURSE CALENDAR later in this syllabus for chapters which we will cover, and when the reading of those chapters should be completed.

IDE Information:
Use of laptop or other computer for completing programming assignments and exercises with software for creating and compiling C programs. The computer lab in ECSS 2.104 has various IDEs you can use for writing programs. The hours of operation are posted at: https://engineering.utdallas.edu/about/facilities/open-access-labs/

We will use a browser-based IDE during lecture: https://www.onlinegdb.com/

CLASS MATERIALS:

Slides, Sample Programs, etc.:
Other materials including the syllabus, slides, sample programs, quizzes, assignments, and participation exercises etc. will be posted in eLearning.

https://elearning.utdallas.edu

TENTATIVE COURSE CALENDAR:

Date	Lecture Material	Reading Completion
August 22	eLearning Access and Syllabus Review	
August 24	Introduction to Computers, the Internet and the Web	Chapter 1
August 29	Introduction to C Programming and IDE Demo	Chapter 2
August 31	Introduction to C Programming	
September 5	**Labor Day – no class**	
September 7	Structured Program Development in C	Chapter 3
September 12	Structured Program Development in C	
September 14	C Program Control	Chapter 4
September 19	C Program Control, and Exam #1 Discussion	
September 21	**Exam #1 – In the testing center, we will *not* have lecture today**	
September 26	C Program Control	
September 28	Functions	Chapter 5
October 3	Functions	
October 5	Arrays	Chapter 6
October 10	Arrays	
October 12	Pointers	Chapter 7
October 17	Pointers	
October 19	Characters and Strings	Chapter 8
October 24	Formatted Input/Output	Chapter 9
October 26	Unions, Bit Manipulation and Enumerations	Chapter 10
October 31	Unions, Bit Manipulation and Enumerations	
November 2	Unions, Bit Manipulation and Enumerations, and Exam #2 Discussion	
November 7	**Exam #2 – In the testing center, we will *not* have lecture today**	
November 9	C File Processing	Chapter 11
November 14	C File Processing	
November 16	C Data Structures	Chapter 12
November 21	**Fall Break – no class**	
November 23	**Fall Break – no class**	
November 28	C Preprocessor	Chapter 13
November 30	Other C Topics	Chapter 14
December 5	Other C Topics, and Exam #3 Discussion	
December 14-16	**Exam #3 – In the testing center**	

This calendar is tentative and may change as the semester progresses. Please check eLearning for announcements regarding changes to the calendar.

Please have the mentioned chapter read (reading completion) before the respective class begins.

INSTRUCTOR POLICIES:

Grading Policy:
Your course average will be calculated as follows:

Exam #1 – 20%
Exam #2 – 20%
Exam #3 – 20%
Programming Assignments – 20%
Online quizzes – 5%
Participation Exercises – 15%

The instructor intends to assign letter grades as shown below. An average with a fractional portion of five tenths or above will be rounded up to the next whole number for determining the letter grade.

Averages	Letter grade
97+	A+
93-96	A
90-92	A-
87- 89	B+
83-86	B
80-82	B-
77-79	C+
73-76	C
70-72	C-
67-69	D+
63-66	D
60-62	D-
Below 60	F

Grading Concerns:
If you think there is a mistake in the grading of your *assignment or exercise*, you must notify *both* the grader and the instructor by email *within two weeks* after the date the grade is posted in the gradebook on eLearning. Keep in mind that a regrade may result in an increase or in a reduction of the original grade.

Most deductions are made because students did not fully read the assignment instructions, disregarded the instructions, did not adequately test their programs, or did not follow the style guidelines provided. You may not change the problem to suit your purposes. Most assignments restrict the use of programming constructs and library functions not covered in lecture, others require that you use particular constructs or functions. To get the maximum credit you MUST read the directions carefully and test your programs thoroughly.

If you think there is a mistake in the grading of your *quiz or exam* and would like to request that it be regraded, you must notify the instructor of this by email *within two weeks* after the date the grade is posted in the grade book on eLearning. Your request for any regrade must describe in detail what you perceive as the problem with the grading. Keep in mind that a regrade may result in an increase or in a reduction of the original grade.

Student Responsibilities:

- You are responsible for all the material in the assigned reading in the required course textbook.
- You are responsible for all the material in the slides.
- You are responsible for all material discussed during course lectures.
- You are responsible for all material supplied on eLearning (including announcements and discussion postings).
- Students are expected to be respectful of each other and of the course instructor. Disruptive behavior will not be tolerated.

Exams:

Course Exams will be administered in the testing center.

We will *not* have lecture on the day of the exams as your exam window will likely be during or overlap our normal lecture time.

Make-up exams will be administered *only for well-documented emergencies*, and you must make every attempt possible, via email, MS Teams private message, or phone, to notify the instructor *prior* to the scheduled date and time or *immediately thereafter*. If notification is not received in a timely manner, no make-up exam will be given.

Programming Assignments:

All assignments will be announced and submitted using eLearning. You will be given at least one week to complete each assignment. Each assignment will include a due date and time. You may submit an assignment up to 24 hours after the due date and time with a 10-point penalty. No excuses will be entertained for late assignments.

You may *not* send your source code to the grader or instructor unsolicited by email expecting us to debug/fix it. This is not reasonable, as our course size is large. Also, part of learning to program is developing your own debugging skills. It is your responsibility to develop your code in a manner that minimizes errors. You should only ask for help with debugging as a last resort. I will help you find errors during office hours, but you should have narrowed down the problem before asking for help. When I look at your code, I will expect to see debug statements, to see that you have done your best to locate the error(s) in question. If I do not see these, I will ask you to check back after you have added them (you may discover the fix after adding appropriate debug statements).

The instructor will drop your lowest assignment grade *if* you submit all assignments as scheduled and receive a grade of 60 or higher on each of the assignments.

Participation Exercises:

Small exercises will be started/given frequently during lecture. These will be due at 11:59pm the day of lecture. There are no make-ups for participation exercises. The instructor will drop the lowest exercise score. As these exercises are announced during lecture, it may be beneficial to attend all class sessions so you do not miss an exercise.

Online Quizzes:
Quizzes will be given in eLearning to ensure that students are keeping up with reading and course preparation. Online quizzes will be announced in eLearning at least 1 week before they are due. There are no make-ups for online quizzes. The instructor will drop the lowest quiz score.

Academic Integrity:
All assignments, quizzes, exercises, and exams are to be individual efforts. You are *not* to collaborate with other students. Prior to the assignment due date, you are not to: discuss assignment solutions with other students, distribute your code to others, or publish your code. Copying of programming assignments, quizzes, exercises, or exams, in whole or in part, from other students will be considered an act of scholastic dishonesty. Copying of assignments from previous semesters will also be considered an act of scholastic dishonesty.

For programming assignments, you may use source code provided by the instructor. You are *not* to view, copy, or distribute code from any other sources, including code from other students, code from assignments submitted in past semesters, or code from the Internet. Plagiarism detection software will be employed to detect copying of code.

Do not share your code with your peers so they can "learn from it". You are guilty of academic dishonesty if you give your code to others or publish it in chat rooms or on websites and other students submit it as their own work.

Extra Credit Work:
Extra credit work will not be given to *individual* students.

What you need to do to be successful in this course:
- Attend every course lecture and pay close attention.
- Read your assigned reading *before* the lecture. The instructor expects you have completed relevant readings before lectures.
- Dedicate 9-10 hours per week outside of course lectures for reading, practicing writing code, completing quizzes, assignments, exercises, and studying for exams.
- Do all your work and do it individually. Students who get too much help from others: the instructor, mentors, peers, and the internet may not acquire the knowledge they need to successfully complete the exams.
- Do not procrastinate. Initial exercises and assignments are typically easy, but as we progress through the course the assignments will become more challenging. All assignments are designed to be worked on over a period of *days or weeks*. Start early so that you have time to try alternate approaches, ask questions, and test your program. A program is not necessarily complete when it compiles. Your program must produce correct results under various conditions, as given in the exercise or assignment. You must design test cases in addition to designing your code.
- Learn to debug your programs yourself. Add print statements that print the values of inputs and the results of intermediate calculations. Add print statements to indicate that sections of code have been reached during execution. Learn to use the integrated debugger.
- Do not wait till the end of the semester to seek help. If you wait until late in the semester, it may be difficult to catch up, as the course is constantly moving forward.

- If you do not understand a concept: reread the text or review the slides. Arrange a meeting with your instructor and come to the meeting with a list of specific questions. Visit the Computer Science Mentor Center, prepared with specific questions.
- Write more programs than are required. The more programs you practice with outside of lecture, the better you will do in this course. You learn to program by doing – coding, testing, and fixing (debugging). This course is like a mathematics course – you need to work *many* problems.
  - Enter the sample programs from the text. Experiment by making small changes. Note how the changes affect the program translation and/or execution.
  - Pick a few of the programming exercises presented in each relevant chapter of the textbook and write programs that satisfy the requirements. This is good practice for the types of coding questions I will ask you on the exam.

MORE FROM UTD:

Class Materials

The instructor may provide class materials that will be made available to all students registered for this class as they are intended to supplement the classroom experience. These materials may be downloaded during the course, however, these materials are for registered students' use only. Classroom materials may not be reproduced or shared with those not in class, or uploaded to other online environments except to implement an approved Office of Student AccessAbility accommodation. Failure to comply with these University requirements is a violation of the Student Code of Conduct.

Classroom Conduct Requirements Related to Public Health Measures

UT Dallas will follow the public health and safety guidelines put forth by the Centers for Disease Control and Prevention (CDC), the Texas Department of State Health Services (DSHS), and local public health agencies that are in effect at that time during the Fall 2021 semester.

Class Attendance

The University's attendance policy requirement is that individual faculty set their course attendance requirements. Regular and punctual class attendance is expected. Students who fail to attend class regularly are inviting scholastic difficulty. In some courses, instructors may have special attendance requirements; these should be made known to students during the first week of classes.

Class Participation

Regular class participation is expected. Students who fail to participate in class regularly are inviting scholastic difficulty. A portion of the grade for this course is directly tied to your participation in this class. It also includes engaging in group or other activities during class that solicit your feedback on homework assignments, readings, or materials covered in the lectures (and/or labs). Class participation is documented by faculty. Successful participation is defined as consistently adhering to University requirements, as presented in this syllabus. Failure to comply with these University requirements is a violation of the Student Code of Conduct.

Class Recordings

Students are expected to follow appropriate University policies and maintain the security of passwords used to access recorded lectures. Unless the Office of Student AccessAbility has approved the student to record the instruction, students are expressly prohibited from recording any part of this course. Recordings may not be published, reproduced, or shared with those not in the class, or uploaded to other online environments except to implement an approved Office of Student AccessAbility accommodation. Failure to comply with these University requirements is a violation of the Student Code of Conduct.

The instructor may record meetings of this course. These recordings will be made available to all students registered for this class if the intent is to supplement the classroom experience. If the instructor or a UTD school/department/office plans any other uses for the recordings, consent of the students identifiable in the recordings is required prior to such use unless an exception is allowed by law.

Comet Creed

This creed was voted on by the UT Dallas student body in 2014. It is a standard that Comets choose to live by and encourage others to do the same:

"As a Comet, I pledge honesty, integrity, and service in all that I do."

Academic Support Resources

The information contained in the following link lists the University's academic support resources for all students.

Please see http://go.utdallas.edu/academic-support-resources.

UT Dallas Syllabus Policies and Procedures

The information contained in the following link constitutes the University's policies and procedures segment of the course syllabus. Please review the catalog sections regarding the credit/no credit or pass/fail grading option and withdrawal from class.

Please go to http://go.utdallas.edu/syllabus-policies for these policies.

The descriptions and timelines contained in this syllabus are subject to change at the discretion of the Professor.