Intermediate Programming for Mechatronics
MTRE 2610 – Fall 2018

Instructor

Kevin McFall (lecture), **David Guerra** (Wednesday lab), **Meiling Sha** (Tuesday and Thursday labs)
E-mail: kmcfall@kennesaw.edu
Office Phone: 470-578-5136
Cell Phone: 610-573-6242
Office Location: Q 322
Office Hours: 11 am - noon MTWRH

Course Description

Catalog Description

This intermediate programming course covers programming topics relevant for Mechatronics Engineering, using tools such as C++, MATLAB, Arduino, and Python. Object-oriented programming techniques are introduced such as encapsulation, classes, inheritance, and operator overloading. Other course components include basic numerical methods and visualization of data in two and three dimensions. Laboratory exercises focus on programming relevant to mechatronics such as acquiring analog, digital, and camera sensor data, motor control, pneumatics, etc.

Course Details

Term: Fall 2018
Course name: Intermediate Programming for Mechatronics
Course number: MTRE 2610
Section number(s): 01, 02, 03
Meeting times: Lecture TR 3:55-4:45 pm, Laboratory T 5:00-7:45 pm (01), R 5:00-7:45 pm (02),
        W 3:30-6:15 pm (03)
Room number: Lecture Q 106, Lab Q 242

Learning Outcomes

By the end of this course, students should:
- Increase proficiency with programming skills
- Understand functional and data encapsulation using object oriented programming
- Apply numerical methods to solve engineering-related problems
- Visualize and manipulate data in two and three dimensions
- Program actuation control using sensor readings

Topics covered

- Encapsulation, overloading, and inheritance in object-oriented programming
- Basic data types, control structures, and functions in MATLAB and Python
- Two and three dimensional plotting in MATLAB and Python
- Symbolic math manipulation in MATLAB
- Structures and cell arrays in MATLAB
- Numpy and matplotlib modules in Python

Textbook

Every student must have their own safety glasses to be worn in laboratory at all times. If the pair you received in MTRE 1000 is lost, they are available for purchase at the bookstore.

Recommended textbooks: How to Think Like a Computer Scientist, C++ version[1] by Allen B. Downey, C++ Annotations[2] by Frank B. Brokken, An Introduction to MATLAB[3] by Krister Ahlersten, Think Python[4] by A. Downey, Matplotlib[5] by J. hunter, et. al., and Guide to NumPy[6] by T. Oliphant are all available for free online. Also consider Udacity's free coding Python courses including Introduction to Computer Science and Intro to Data Analysis.

Technical Requirements

Even more important than the textbooks are the required software MATLAB, available free of charge for KSU students[7], the free Visual Studio Community[8], and the open source Python 3 (see installation instructions posted on D2L).

Grading Policy	
Homework	20%
Test 01	15%
Test 02	15%
Test 03	15%
Laboratories	20%
Final exam	15%
Total	**100%**

Grade Conversion: A: (90-100), B: (80-89), C: (70-79), D: (60-69), F: (0-59)

Homework

Several programming assignments will be collected which can be completed individually or in groups of two. Source code will be submitted in a D2L Dropbox with filenames beginning with "HW" followed by 2 digits for the homework number, followed by the last name(s) of the student(s). Homework written in MATLAB will be submitted as a single .m file with the filename as described. Similarly, Python submissions will be a single appropriately named .py file. C++ homework is submitted instead as a .zip file with the required file name. The .zip file will contain a folder with the required name, which itself contains a folder for each problem housing the entire Visual Studio project. Each folder will contain all the required files and subfolders so that the project can be opened and the code executed. Only a single submission for groups of two is required. When executed, the program will output to the screen the problem numbers and answers. Problems with graphical output will display which figure number contains the solution.

- Submission requirements satisfied (1 point)
- Significant progress made toward solution (1 point)
- Correct output generated (1 point)
- All elements of problem description properly addressed (1 point)

---

[1] http://greenteapress.com/thinkcpp/
[2] http://cppannotations.sourceforge.net/annotations/html/
[3] http://bookboon.com/en/an-introduction-to-matlab-ebook
[4] http://www.greenteapress.com/thinkpython/thinkpython.pdf
[5] http://matplotlib.org/Matplotlib.pdf
[6] http://csc.ucdavis.edu/~chaos/courses/nlp/Software/NumPyBook.pdf
[7] https://apps.kennesaw.edu/files/pr_app_uni_cdoc/doc/Matlab_DownloadInstructions.pdf
[8] https://www.visualstudio.com/downloads/download-visual-studio-vs

Tests

Three in-class tests will be administered where students have access to internet-connected computers to compile and execute programs. Test grades may be curved in an attempt to maintain the class average in the C range. Additionally, when calculating the final grade, the lowest test score will be set to the second lowest score (including the final exam) to eliminate the effects of one bad test grade. No assistance from any human may be solicited during the test period. Source code solving each problem will be uploaded to a D2L Assignment dropbox before the end of the test in the same manner as homework but with "HW" replaced with "Test". Problems will be graded according to the following rubric:

0	No content relevant to solving the problem
30 (F)	Some relevant content but no indication of how to solve the problem
60 (D)	Some indication that the correct solution method is being followed
70 (C)	Significant work showing understanding of how to approach the problem
80 (B)	Most details of problem solution complete but significant mistakes made
90 (A)	Correct solution except for minor mistakes such as typos, sign, or algebra errors
100	Problem is answered completely with the correct answer

Laboratory Exercises

Laboratory exercises will generally be graded during the scheduled laboratory time and groups not completing in time will be assessed on what is finished. Grades will be assigned according to the fraction of the lab content correctly addressed.

Final Exam

A cumulative final exam will follow the same format as the tests, although twice as long, with "Final" in the submitted filename.

Course Expectations

Attendance Policy

Forcing everyone to come to every class is not practical. Each student bears responsibility for material covered in class. If students choose to miss class, that is their decision. Class time will be spent explaining the day's content and working problems, under the assumption that all students have read and attempted to understand the reading assignment. In general, late assignments are not accepted nor can make-up tests be administered. Extenuating circumstances can result in exceptions to these rules, but agreement must be reached with the instructor in advance of the assignment or test that will be missed.

Course Communication

Course material will be disseminated in D2L including lecture notes, homework solutions, etc. All official course announcements, including instructions when class may be cancelled, will be posted in the D2L course news. Be sure to check D2L regularly.

Course Outline

Day	Date	Description	Reading	HW due	Lab exercise
Tue	Aug 14	0[9]: Standard vectors	10[10]		C++ review
Thu	Aug 16	1: Structure data and functions	8, 11[10], 3.2[11]		
Tue	Aug 21	2: Classes	14[10], 7,9[11]	HW01	Arduino basics
Thu	Aug 23	3: Function/operator overloading			
Tue	Aug 28	4: Recitation		HW02	Complex class
Thu	Aug 30	5: File I/O	11[11]		
Tue	Sep 04	6: Inheritance	13[11]	HW03	H-bridge motor drivers
Thu	Sep 06	7: Constructors/destructors			
Tue	Sep 11	8: Recitation			Controlling motors
Thu	Sep 13	Review		HW04	
Tue	Sep 18	Test 01			Pneumatics and sensors
Thu	Sep 20	9: MATLAB environment	1, 2, 3, 8, 10, 13[12]		
Tue	Sep 25	10: Vector and matrix indexing	4[12]		Color sorting robot
Thu	Sep 27	11: 2D plotting	7[12]		
Tue	Oct 02	12: User defined functions	9[12], 12[12]		Image processing
Thu	Oct 04	13: Surface and contour plotting			
Tue	Oct 09	14: Symbolic math		HW05	2D tracking
Thu	Oct 11	15: Structures and cell arrays			
Tue	Oct 16	16-17: File I/O	6[12]		Mandelbrot
Thu	Oct 18	18: Review		HW06	
Tue	Oct 23	Test 02			Distance sensor
Thu	Oct 25	19: Tuple/List/Array/Matrix	1, 2, 8, 10, 11, 12[13]		
Tue	Oct 30	20: Control structures	5, 7[13]		PI Control
Thu	Nov 01	22-23: NumPy and matplotlib	1-8[14], 1-3[15]		
Tue	Nov 06	21: User defined functions	3, 6[13]		Serial communication
Thu	Nov 08	24: Recitation		HW07	
Tue	Nov 13	Recitation			3D tracking
Thu	Nov 15	Review		HW08	
Tue	Nov 27	Test 03			Accelerometer
Thu	Nov 29	25: Review			

---

[9] These numbers indicated which recorded lectures should be reviewed before coming to class.

[10] *How to Think Like a Computer Scientist*

[11] *C++ Annotations Version 10.2.0*

[12] *An Introduction to MATLAB*

[13] *Think Python*

[14] *Guide to NumPy*

[15] *Matplotlib*

Help Resources

Contacts to get Help

Student Help Desk studenthelpdesk@kennesaw.edu or call 470.578.3555
KSU Service Desk[16]
D2L Student User's Guide[17]

Additional Resources

Remote access to Library Resources[18]
Student success Services[19]
Tutoring and Academic Support[20]
Academic Advising[21]
University bookstore[22]

University Policies

Academic Honesty

Every KSU student is responsible for upholding the provisions of the Student Code of Conduct, as published in the Undergraduate and Graduate Catalogs. Section II of the Student Code of Conduct addresses the University's policy on academic honesty, including provisions regarding plagiarism and cheating, unauthorized access to University materials, misrepresentation/ falsification of University records or academic work, malicious removal, retention, or destruction of library materials, malicious/intentional misuse of computer facilities and/or services, and misuse of student identification cards. Incidents of alleged academic misconduct will be handled through the established procedures of the University Judiciary Program, which includes either an "informal" resolution by a faculty member, resulting in a grade adjustment, or a formal hearing procedure, which may subject a student to the Code of Conduct's minimum one semester suspension requirement.

Plagiarism Policy

No student shall receive, attempt to receive, knowingly give or attempt to give unauthorized assistance in the preparation of any work required to be submitted for credit as part of a course (including examinations, laboratory reports, essays, themes, term papers, etc.). When direct quotations are used, they should be indicated, and when the ideas, theories, data, figures, graphs, programs, electronic based information or illustrations of someone other than the student are incorporated into a paper or used in a project, they should be duly acknowledged.

Collaboration on assignments among students and other individuals is wholeheartedly encouraged. In order to avoid possible plagiarism issues, limit such collaboration to discussion of how to approach the problem and what strategies, equations, and techniques should be used to solve it. When actually writing down your solution, ensure you are not in the same room as outside collaborators nor referencing a copy of their work. Your solution will then be written in your own words and therefore not plagiarized. Violations of this policy are STRICTLY ENFORCED.

---

[16] https://apps.kennesaw.edu/portal/prod/app_its_ask_stu_publ/student/

[17] https://apps.kennesaw.edu/files/pr_app_uni_cdoc/doc/D2LBrightspaceStudentguide_10.3.pdf

[18] http://library.kennesaw.edu/

[19] http://www.kennesaw.edu/studentsuccessservices/

[20] http://learnonline.kennesaw.edu/student-resources/tutoring.php

[21] http://advising.kennesaw.edu/

[22] http://bookstore.kennesaw.edu/home.aspx

Disability Statement

Kennesaw State University provides program accessibility and reasonable accommodations for persons defined as disabled under Section 504 of the Rehabilitation Act of 1973 and the Americans with Disabilities Act of 1990. Kennesaw State University does not deny admission or subject to discrimination in admission any qualified disabled student.

A number of services are available to help students with disabilities with their academic work. In order to make arrangements for special services, students must visit the Office for Student Disability Services and make an appointment to arrange an individual assistance plan. In most cases, certification of disability is required.

Special services are based on

- medical and/or psychological certification of disability,
- eligibility for services by outside agencies, and
- ability to complete tasks required in courses.

ADA Position Statement

Kennesaw State University, a member of the University System of Georgia, does not discriminate on the basis of race, color, religion, age, sex, national origin or disability in employment or provision of services. Kennesaw State University does not discriminate on the basis of disability in the admission or access to, or treatment or employment in, its programs or activities.

The Americans with Disabilities Act (ADA), Public Law 101-336, gives civil rights protections to individuals with disabilities. This statute guarantees equal opportunity for this protected group in the areas of public accommodations, employment, transportation, state and local government services and telecommunications.

The following individuals have been designated by the President of the University to provide assistance and ensure compliance with the ADA. Should you require assistance or have further questions about the ADA, please contact:

- ADA Compliance Officer for Students: 470-578-6443
- ADA Compliance Officer for Facilities: 470-578-6224
- ADA Compliance Officer for Employees: 470-578-6030

For more information, go to: http://www.kennesaw.edu/stu_dev/dsss.