KENNESAW STATE
UNIVERSITY

SYLLABUS
COLLEGE OF SCIENCE AND MATHEMATICS
DEPARTMENT OF PHYSICS
PHYS3500K: COMPUTATIONAL PHYSICS I
SPRING 2022

Course Information

Class meeting time and Location:
Tuesday and Thursday at 15:30 -16:45 Marietta Campus, Academic Bldg, Room 250.
Tuesday and Thursday at 17:00 -18:15 Marietta Campus, Atrium Bldg, Room 212.
Modality: Face to Face course;
Syllabus: posted in D2L

Instructor Information

Name: Dr. Marco Guzzi
E-mail: mguzzi@kennesaw.edu
Website: http://facultyweb.kennesaw.edu/mguzzi/
Office Location: Kennesaw Campus, Science Bldg, 4th floor, Room SC436
Office phone: 470-578-4783
Office Hours: By appointment
Preferred method of communication: e-mail
(When e-mailing, please put "PHYS3500K" in the subject line along with the subject of your message. Do Not use D2L or other email providers to send emails, you will not get a reply. We all must adhere to the KSU safety protocols.)

Course Description

2 Class Hours 3 Laboratory Hours 3 Credit Hours
Prerequisite: Grades of "C" or better in PHYS 2212/2212L

This course utilizes introductory computer programming to analyze situations that are unique to physics. Students will enhance their computational thinking by using these methods and learn to obtain high-accuracy approximate solutions to physics problems that are not solvable by analytic means. No prior programming knowledge will be assumed and the basics of one or more of the standard programming languages C/C++, Fortran, Python will be included in the course instruction.

Course Materials

The material is presented in multiple ways: Projector, Whiteboard, Power Point presentations. Videos will also be shown. Students are expected to take notes in class, save their work on a flash drive during each class meeting. Students are responsible to catch up with the material if they miss lectures.

Required Textbook:
For C/C++: http://www.cplusplus.com/ (free of charge)

Recommended Textbooks (For Syntax):
``C Programming: A Modern Approach'', 2nd Edition, by K. N. King.
``C++ Primer'', 5th Edition, by Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo.
``The C++ Programming Language", by Bjarne Stroustrup.
Recommended Textbooks (For numerical methods):
``An Introduction to Numerical Methods and Analysis'' by J.F. Epperson.
``Numerical methods for Physics'' by A. Garcia
``Computational Physics Problem Solving with Computers'', by R.H. Landau, M.J. Paez, C.C. Bordeianu.
``Numerical Methods'' Jeffrey R. Chasnov

Technology requirements: Class time consists of both lectures and programming exercises. Students are encouraged—but not required—to bring a computer to each class. There are computer labs on the KSU campus. Students are expected to save copies of their programs and back them up to other media (such as a flash drive or external hard drive).

Learning Outcomes

C/C++/Fortran are a general-purpose, procedural computer programming languages that are especially suited to numeric computation and scientific computing. The main learning outcomes are listed below:
1) Students demonstrate knowledge of language syntax and the ability to structure code.
2) Students will be able to solve physics problems involving linear equations using numerical methods.
3) Students will learn basic numerical techniques to solve integral and differential equations in physics.
4) Students will be able to perform numerical integration and discuss the accuracy of the result for applications in physics.

Course Requirements and Assignments

Students are expected to attend all lectures, take all tests and exams, and complete all homework assignments.

Evaluation and Grading Policies

Three tests will be given during the semester. Grades will be determined according to student's performance on the three tests.

Homework: **10%**
Tests: **60% (3 tests, 20% each)**
Final Exam: **30%**
Grading Scale: **A: 90% - 100%; B: 80% - 89%; C: 70% - 79%; D: 60% - 69%; F: 0 - 60%.**

Tests and exams are graded by assigning points for:

- writing a code that compiles, runs, and generate the correct output.
- writing a code using an optimized syntax implementation.

D2L (Internet-based utility)
Course information, homework solutions and announcements will be available "D2L".
PHYS3500K course information system is accessible from http://d2l.kennesaw.edu/
To sign on, use your KSU Local Area Network (LAN) username and password.
Students are expected to check D2L for announcements at least once a day.

Course Policies (Important!)

1. Regular lecture and programming exercise attendance is essential for success in this class. If students must miss class, it is their responsibility to get notes/material from another student.
2. Be on time for the lecture.
3. Cellular telephones, pagers, and similar devices must be turned off or placed in silent mode during class. Use of cell phones should be restricted to emergencies.
4. The usage of any other external devices (other than the assigned computer) which can connect to the internet is strictly forbidden. That is considered cheating.
5. During lectures, students must avoid conversations and other disruptions that may distract other students from listening and learning. If students have a question or comment, they should direct it to the instructor.
6. Occasionally, it may be necessary for the instructor to make corrections or changes to the syllabus. Corrections or changes to the syllabus will be announced on the KSU D2L website and in class: students are expected to check D2L for announcements at least once or twice a day.

Homework Assignments

Homework will be assigned during the lectures. Solutions to problems will be discussed in class during lectures and will be posted on D2L. Students must hand in their homework through D2L. Repositories in the ``Assignments'' section in D2L will be created for each assignment.

Withdrawal, Last day of class, and Final Exam

- First day of classes: Tue, January 11, 2022.
- Last day to withdraw without academic penalty: Tue, March 15, 2022, at 11:45pm.
- Last day to withdraw with a WF: April 25, 2022.
- The last day of Class: Mon, May 2, 2022.
- Final exam: Thu, May 5, 2022, 3:30pm - 5:30pm.

(These must be double checked again on the KSU office of registrar website.)

Department or College Policies

The university's withdrawal policy is explained at:

https://registrar.kennesaw.edu/academic-records/academic-standing-appeals/withdrawal-appeal.php

The Academic Standing Appeal policy is explained at:

https://registrar.kennesaw.edu/academic-records/academic-standing-appeals/index.php

Students are solely responsible for managing their enrollment status in a class.
Nonattendance does not constitute a withdrawal.

Make-up Exam policy

Make-up exams will not be given. If students know ahead of time that they have a conflict, they must let the instructor know. If students miss an exam because of an illness (student or a family member) or some other unforeseeable event, students must contact the instructor as soon as possible. They can e-mail the instructor or call the Physics Dep. Office at 470-570-4205. Students must provide documentation showing the reason for missing the exam. Final make-up exam is **ONLY** for documented and excused emergencies or for scheduling conflicts with other final exams.

Institutional Policies

Federal, BOR, & KSU Course Syllabus Policies:
http://curriculum.kennesaw.edu/resources/federal_bor_ksu_student_policies.php

Student Resources:
http://curriculum.kennesaw.edu/resources/ksu_student_resources_for_course_syllabus.php

Academic Integrity Statement:
http://scai.kennesaw.edu/codes.php

Students with Disabilities

Any student with a documented disability or medical condition needing academic accommodations of class-related activities or schedules must contact the instructor immediately. Written verification from the KSU Student Disability Services (http://sds.kennesaw.edu/) is required. No requirements exist those accommodations be made prior to completion of this approved University documentation. All discussions will remain confidential.

KSU Student Resources

This link contains information on help and resources available to students:
https://curriculum.kennesaw.edu/resources/ksu_student_resources_for_course_syllabus.php

Course Delivery

KSU may shift the method of course delivery at any time during the semester in compliance with University System of Georgia health and safety guidelines. In this case, alternate teaching modalities that may be adopted include hyflex, hybrid, synchronous online, or asynchronous online instruction.

Covid-19 Illness

If you are feeling ill, please stay home and contact your health professional. In addition, please email your instructor to say you are missing class due to illness. Signs of COVID-19 illness include, but are not limited to, the following:

- Cough
- Fever of 100.4 F or higher
- Runny nose or new sinus congestion
- Shortness of breath or difficulty breathing
- Chills
- Sore Throat
- New loss of taste and/or smell

COVID-19 vaccines are a critical tool in “Protecting the Nest.” If you have not already, you are strongly encouraged to get vaccinated immediately to advance the health and safety of our campus community. As an enrolled KSU student, you are eligible to receive the vaccine on campus. Please call (470) 578-6644 to schedule your vaccination appointment or you may walk into one of our student health clinics.

For more information regarding COVID-19 (including testing, vaccines, extended illness procedures and accommodations), see KSU’s official Covid-19 website.

Face Coverings

Based on guidance from the University System of Georgia (USG), all vaccinated and unvaccinated individuals are encouraged to wear a face covering while inside campus facilities. Unvaccinated individuals are also strongly encouraged to continue to socially distance while inside campus facilities, when possible.

Course (Tentative) Schedule

Week1

Tuesday
0. Introduction to Programming Languages
1. Why C/C++ ?
2. Resources for this course
3. History
4. Programs and Programming languages
5. MobaXterm
6. Basic line commands in Linux/Unix
7. Structure of a program: Hello World!
8. Compiling and Linking
9. Running the GNU compiler from command line
10. Running executables

Thursday
0. Recap
1. Analysis of first.cpp
2. More on Variables, Identifiers, and Data Types
3. Declaration of variables
4. Initialization and deduction
5. Strings, constants, Operators
Hands-on examples

Week2

Tuesday
0. Recap
1. More on Operators:
-Increment and decrement (++, --)
-Relational and comparison operators
-Logical Operators (!, &&, ||) aka (NOT, AND, OR)
-Conditional ternary operator (?)
-Comma operator (,)
-Bitwise operators (&, |, ^, ~, <<, >>)
-Explicit type casting operators
-*sizeof* and precedence of operators

Thursday
0. Recap

2. Basic Input/Output: Standard Input
3. Stringstream
4. Statements and Flow control:
-Selection statements: if and else,
-Iteration statements (loops): while loop
-Iteration statements (loops): do while loop
-Iteration statements (loops): for loop
-Hands-on examples

Week 3

Tuesday
0. Recap
1. Range-based for loop
2. Jump Statements (break,goto, switch)
3. Functions in C++
-Functions with no type (the use of void)
-The return value of main
-Arguments passed by value and by reference
-Passing values by reference
-Efficiency considerations and const references
-Hands-on examples

TEST 1 Thursday Feb 3, 2022

Week 4

Tuesday
0. Recap
1. More on Functions:
-Arguments passed by value and by reference
-Default values in parameters
-Declaring functions
-Recursivity
2. Overloaded Functions
3. Function Templates
-Non-type template arguments
-Name Visibility
4. Namespaces
-Hands-on examples

Thursday
0. Recap
1. About debugging your codes
2. More on Namespaces
-Using std namespace
3. Storage Classes
4. Arrays
-Accessing the values of an array
-Multidimensional arrays
-Arrays as parameters to pass to functions
5. Passing arrays as arguments to functions
-Multidimensional arrays as parameters
-Hands-on examples

Week5

Tuesday
0. Recap
1. Library arrays
2. Character sequences
-Initialization of null-terminated character sequences
-Strings and null-terminated character sequences

3. Pointers
-Dereference operator (*)
-Declaring pointers
-Pointers and arrays
-Examples: Cramer’s Rule
-Hands-on examples

Thursday
0. Recap
1. Pointer initialization
2. Pointer Arithmetics
3. Pointer and Const
4. Pointers and string literals
5. Pointers to pointers
-Void pointers
-Hands-on examples

Week 6

Tuesday
0. Recap
1. Invalid pointers and null pointers
2. Pointers to functions
3. Dynamical memory allocation
-Mechanisms to check if the allocation was successful
-Operators delete and delete[]
-Examples
4. Data structures
-Accessing members
-Examples: structures in action
-Hands-on examples

Thursday
0. Recap
1. Pointers to Structures
-Examples
2. Nesting structures
3. More examples before we move forward (here is from King and C++ Primer by Lippman, Lajoie, Moo)
-More examples on Pointers: Pointers as Iterators
-Pointers as Iterators
-Initializing elements of a Multidimensional Array
-Multidimensional arrays and pointers
-Passing a Multidimensional Array to a function
-Hands-on examples

Week 7

Tuesday
0. Recap
1. Types Aliases
-Other data types: Type aliases (typedef / using)
2. Unions
Anonymous unions
3. Enumerated Types
4. Classes I
-Examples
5. Constructors
-Hands-on examples

Thursday
0. Recap
1. More on Constructors

2. Overloading Constructors
-Uniform Initialization
3. Member initialization
4. Pointers to classes
5. Classes I. defined with struct and union
6. Dynamic allocation: more examples/applications
-Hands-on examples

Week 8

Tuesday
0. Recap
1. Classes II: Overloading operators
2. The keywork ``this''
3. Static members
4. Const member functions
5. Class templates
-Hands-on examples

TEST 2: Thursday March 3, 2022

Week 9

Tuesday
0. Recap
1. More on class templates
2. The vector class (C++ Primer by Lippman, Lajoie, Moo, and ``The C++ Programming Language'', by Bjarne Stroustrup)
-Library vector Type
-Defining and Initializing a vector
-List Initializing a vector
-Creating a Specified Number of Elements
-Restrictions to value initialization
-List Initializer or Element Count?
-Adding Elements to a vector: push_back
-Other vector operations
-Little recap: Range-based for loops
-The empty and size members
-Computing a vector Index
-Example
-Hands-on examples

Thursday
0. Recap
1. Iterators: intro and definitions
-Examples
3. Operations with iterators
4. The function template Copy() in C++ STL
-Hands-on examples

Week 10

Tuesday
0. Recap
1. More on the function template copy_n() in C++ STL
2. Handling files: Input/output with files
-ifstream, ofstram, fstream
-Open and Closing a file
-Text files: writing
-getline
-Examples

Thursday

0. Recap
1. Numerical Methods (here I use An Introduction to Numerical Methods and Analysis by J.F. Epperson)
-Intro, motivation, dealing with errors,
2. Roor Finding
-Bisection method
-Convergence and Errors
-Bisection method: Pros and Cons
-Hands-on examples

Week 11

Tuesday
0. recap
1. Newton-Raphson's method
Analytic, Limitations, Where to Stop
2. Secant Method
Convergence
Tools from calculus: Taylor Expansions
Taylor's theorem with reminder
3. Difference approximation to derivative
Example: truncation and round-off errors
4. Discrete derivatives: 5-point stencil
Hands-on examples

Thursday
0. Recap
1. System of linear Equations
Linear Algebra review
Recap: Linear systems and the Cramer's rule
Recap: Explicit formulas for small systems e.g. 3 x 3
2. Gaussian Elimination
Example of implementation
backward solution algorithm
3. Pivoting
Hands-on examples

Week 12

Tuesday
0. Recap
1. Linear Systems of equations: Physics applications
-Steady States
-Coupled harmonic oscillators
-Kirchhoff's law applied to circuits
-Least squares fit method
-Quadratic Fit (Extension)
2. Non-linear systems of equations
-Hands-on examples

Thursday
1. Operation counts
2. Numerical Integration: Intro
3. Linear Interpolation
4. Lagrange Interpolation
5. Trapezoid rule
6. Simpsons's rule
-Hands-on examples

Week 13
Tuesday
0. Recap
1. Numerical Methods for Ordinary Differential Eqns
-Examples
2. Euler's method (and variants)

-Examples
3. Implicit Methods and Predictor-Corrector Schemes
-Examples
-Hands-on examples

TEST 3: Thursday April 7, 2022

Week 14

Tuesday
0. Recap
1. Runge-Kutta's method
-Examples
2. Non-linear pendulum
3. Renormalization group equations
-Running of Coupling constants: QCD and Asymptotic freedom
-Hands-on examples

Thursday
0. Recap
1. Writing Large Programs
2. Header files
-Sharing macro definitions and Type definitions
-Sharing Functions Prototypes
-Sharing variables declarations
-Building multiple-file programs
3. Makefiles and make
-Examples (5 step-by-step examples of increasing level of complexity)
-Hands-on examples

Week 15

Tuesday
0. Recap
-Runge-Kutta for a 2nd order ODE
1. More on makefiles
-Basics of Variable References: $
2. Automatic Variables
3. Data visualization
-Plotting tools: Gnuplot
-Hands-on examples

Thursday
0. Recap
1. More on Gnuplot
2. Plotting data from file
-Data vs Theory comparisons: example
3. gnuplot Surface (3-D) Plots
-Vector fields in gnuplot
-Hands-on examples

Week 16

Tuesday
1. Partial Differential Equations (PDEs)
2. Classification of PDEs
3. A survey on main methods: Basic Problem
4. Explicit methods and Stability
-Example
5. Implicit methods: Crank-Nicolson (briefly)

6. More on parabolic PDEs (Here Numerical methods for Physics by A. Garcia)
7. Heat diffusion
-Analytic expansion method
Algorithm for numerical solutions
8. Critical Mass: Neutron diffusion and the nuclear fission process
-Solutions
-Critical Mass for a chain reaction
-Hands on applications

Thursday
1. Hyperbolic PDEs: Wave equation
-Initial Conditions, boundary conditions
-Solutions of the wave equation
-Implementation examples
2. Time stepping
3. 1+1 dim Schrödinger equation
4. 1+2 dim Schrödinger equation
-Final remarks

Final Exam: Thursday, May 5, 2022, 3:30pm - 5:30pm