SYLLABUS

ME4010/6010 – Computational Methods for Mechanical Engineering

3 Credit Hours

Fall Semester, 2016

Instructor: **Prof. Nikolai Priezjev**
430 Russ
Email: nikolai.priezjev@wright.edu
Web: http://www.wright.edu/~nikolai.priezjev
Mr. Roussel Rahman
201 Russ
Email: rahman.31@wright.edu

Pilot Location: the course name in Pilot is "ME-4010-01 - Comp Meth for Mech Engr"

Class time and location: MWF, 12:20 pm - 1:15 pm, 150 Russ

Office Hours: Monday and Wednesday, 2:00 pm-3:00 pm (Nikolai Priezjev)
Tuesday and Thursday, 10:00 am-11:00 am (Roussel Rahman)
And email us to make appointments

Course Outline:

Combines material learned in statics, dynamics, thermodynamics, fluid mechanics, and heat transfer and applied them to the design of mechanical systems using numerical methods

Text:

Course handouts will be distributed. Students have to keep a good and neat course note.

References:

Textbooks for Statics, Dynamics, System Dynamics, Thermodynamics, Fluid Mechanics and Heat Transfer

Course Description:

This course combines the materials your have learned in statics, dynamics, thermodynamics, fluid mechanics and heat transfer and applies them to the designs of mechanical systems using numerical methods. The example illustrated in this course will cover diverse areas in the designs of mechanical and chemical transport processes, such as manufacturing, energy systems, cooling of electronic equipment, power cycles, engines, heat exchangers, hydraulics and pneumatics and fire and explosion protection systems. The basic approach involves (1) develop a suitable model for the system; (2) derive a numerical solution procedure; (3) use computer program to solve the equations and (4) present the solution outcome.

Prerequisites: ME3360 & ME3210

Project Reports:
There will be one introduction assignment and 4 homework projects and all projects will be considered comprehensive. The students have to derive all basic governing equations (using MATHEMATICA or MATLAB when appropriate), write FORTRAN90 programs, plot results using TECPLOT and submit the final report using Microsoft Words. The guidelines to prepare the reports are given at the Appendix.

You need to submit all Fortran files (.f90 or .f95 files) used in each project to Pilot's dropbox (preferably in a compressed folder such as a .zip file). Please see the Project Report Guidelines for the report. All reports must be submitted electronically to the Pilot's dropbox before the deadline.

Exam Policy:
One midterm exam will take place in-class sometime in October between 12:20 pm - 1:15 pm. The final exam will be take place in the classroom on WEDNESDAY, December 14, 2016 from 12:30pm - 2:30pm. All exams are comprehensive.

Grading:
Introduction assignment: 5%
Project 1: 12.5%
Project 2: 12.5%
Midterm: 20%
Project 3: 12.5%
Project 4: 12.5%
Final: 25%
You may submit an incomplete project to get partial credits.

Grading Scale:

85-100	A
75-84	B
65-74	C
55-64	D
below	E

Attendance Policy:
You are responsible for maintaining class notes, assignments and due dates.

Late Assignment Policy:
Assignments may be turned in late at a penalty of 10% per 24 hours from the due date. After three days, you will receive a zero point for your assignment.

Cheating Policy:
You are allowed to discuss your projects with your fellow students but no copying of the assignment and computer program is allowed. The graders have been instructed to identify homework that displays evidence of verbatim copying; all such solutions will receive zero credit regardless of the source of the solution, i.e. the person providing the solution (or computer program) will also receive a zero score. In addition, you will be reported to Office of Community Standards and Student Conduct for violation of student codes.

Cases of academic dishonesty, which include copying of homework or lab assignments, plagiarism of lab abstracts, or cheating on exams, will be dealt with according to the procedures set forth in the university's academic integrity policy at http://www.wright.edu/students/judicial/integrity.html. College of Engineering and Computer Science students found guilty of two violations of the university's academic integrity policy are subject to dismissal.

Anticipated Outcomes:
(1) Capability to develop and apply appropriate forms of conservation laws to various practical design situations
(2) Capability to use software such as MATHEMATICA or MATLAB to derive governing equations in symbolic form
(3) Ability to solve complex algebraic equations using a computer
(4) Ability to solve ordinary differential equations using a compute
(5) Ability to cast partial differential equations into algebraic equations and solve them using a computer
(6) Ability to write your own computer programs to solve complex physical problems
(7) Ability to use computer to present results and draw important conclusions from the results
(8) Ability to perform technical writing

Course Outline

Class	Date	Topic	Reading
1		Preparing for the course –	
		installation of the software	see Pilot
		Introduction to the Conservation Law	Chapter 1
2		First programming lessons	Chapter 1
3		Numerical Solution of Algebraic	
		Equations	Chapter 2
4		Newton Raphson Method	Chapter 2
5		In-class Programming	
6		More examples	
7		Numerical Solution of ODE	Chapter 3
8		Euler and Modified Euler Methods	Chapter 3

9	In-class Programming	
10	Runge Kutta Methods	Chapter 3
11	Examples	
12	Curve fitting – Least square methods	Chapter 4
13	Examples	
14	Interpolation	Chapter 4
15	Examples	
16	**Mid-term exam**	
17	Multi-dimensional Heat Conduction	Chapter 5
18	Work through an example on paper	Chapter 5
19	Implement of boundary conditions	Chapter 5
20	TDMA	Chapter 5
21	Cylindrical/Spherical coordinates	Chapter 5
22	Veteran’s Day Holiday; University Closed	
23	In-class Programming	
24	Solution of Thin Shear Flows	Chapter 6
25	Control volume approach & upwind differencing scheme	Chapter 6
26	Boundary Layer flows	Chapter 6
27	Thanksgiving Holiday; University Closed	
28	In-class Programming	
29	SIMPLE method	Chapter 6
30	Flow in a tube or channel	Chapter 6

Wright State University
Department of Mechanical and Materials Engineering
ME 4010/6010 – Computational Methods for Mechanical Engineering
Project Report Guidelines
Fall Semester, 2016

Introduction: (3%)
Before a numerical method can be applied, the problem should be understood and well defined. The purpose of this section is to clearly yet succinctly define the problem solved and state what approach was used and why.

To get full credit you need to answer three questions in your introduction: (1) What is the situation that requires a solution? This would include a description of the system to be modeled, governing equations, and any boundary conditions. (2) What do you need to find and why? (3) What approach will you use to find the results and what is the reason for using this approach?
This section should not include any derivations or specific details on the problem setup or methodology. Instead it should introduce the reader to what you will discuss in the rest of the report. ***Do not just provide a list –*** providing a numbered list will not get you full credit.

Methodology: (25%)
It is important to include a detailed description of how the problem was solved so others can follow, reproduce or add to your work. The purpose of this section is to describe in greater detail how the problem was approached and solved and demonstrate an understanding of how to move from a physical system to a numerical model.

You should include a detailed description of the numerical method and how the method was applied to the specific problem. If the same numerical method was used for multiple problems a description of the method itself only needs to be included in the problem in which it first appears with a clear reference to the description in all subsequent problems. However, a description of how the method was applied should be supplied for each problem. This is a full description of how you prepared the problem for finding a numerical solution. If it was necessary to derive any equations or any other significant setup was necessary to generate a solution it should be included in this section.

To receive full credit for this section you need to demonstrate a full understanding of the concepts used to produce a correct solution in your own words and provide a clear description of everything that was done in order to produce a numerical model. Equations take a lot of time to produce electronically -- I recommend leaving blank spaces in your reports where you can write them in by hand. Do not turn in a pile of unordered derivations as your appendix to meet this requirement.

Results: (25%)
By themselves, results are just numbers. As engineers we don’t just write programs to get an answer, we also need to interpret the numbers output by the program and communicate them clearly. In this section you will present the products of your coding that demonstrate your code works properly and produces a correct solution to the problem.
This section should include all plots, figures, or values significant to the problem solution. An explanation is necessary for anything included in the results section; do not just provide figures or numbers! Compare results if multiple input values are required. Plotting your solutions is often helpful and adds to your discussion. If multiple numerical methods were used, compare them and explain which method worked the best and support that claim.

Conclusion: (2%)
The purpose of this section is to summarize your report. Briefly summarize the problem, results, and your interpretation of the results. The conclusion ***should not include any new information*** – it is solely for summarizing what was discussed and found in the other sections of the report. This section should be short and is used as a quick read of the findings without the details. ***Do not answer questions from project handouts in this section!***

Coding: (45%)
This is a copy of the actual code that you wrote. Your code will be checked and run to verify proper operation and output. You may collaborate with other students but your code needs to be your own product. Duplicated codes will not receive any credit. Projects from previous classes have been retained by the course instructors and will be checked for duplications as well.
You need to submit a copy of your code for each problem. If different parts of the project require significant modifications you may need to submit a different copy for each part. This is not necessary if you are just changing boundary conditions or input values. Please submit your code with the appendix of your report as well as the actual files in a compressed format file (.zip file) and **clearly indicate which code belongs to which problem or part of the problem**.

Other Notes:
This format should be followed for each problem. If a single problem includes multiple sections, separate the results by each section. If there is more than one problem, present the first problem completely then start over again for the next problem. You may produce separate reports for each problem.

Handdrawn/written figures/equations are ok. If you need to use a figure to explain how to set the problem up you may draw them by hand in your report.

All figures and equations need to be appropriately labeled. It is much easier to describe an equation by number in your discussion rather than 'that one where I modified something' or a figure by a number rather than distinguishing it from a series of similar plots.

BE SURE TO INCLUDE EACH SECTION! No points will be awarded for a section that is missing.

You will receive no credit for information that is misplaced. For example, if you discuss the results in the conclusion you will lose points in the results section for missing information. If you place derivations in the introduction you will lose points in the methodology section for excluding information.

Comment your code liberally. Comments can demonstrate a better understanding of the program and will save you time for more complicated projects. It will also help your grader follow your program and is a good habit professionally.