Numerical Method

Basic Course Information

Course Number	01005117	**Subject Category**	Compulsory (M)
Class Format	Lecture	**Credit Type and Number of Credits**	1
Department	Mechatronics	**Student Category**	Year 5
Period of Study	Semester 1	**Classes per Week**	1
Required Materials	Google Collaboratory and GitHub. Internet connection is required.		
Instructor	Takehisa Yamamoto	Wutipong Preechapoolaku	

Course Objective

The course provides students with introduction and basis of numerical calculation. Students willl develop their programs for numerical solutions of differential and integral calculus. In addition, students will implement programs for engineering problems in this course.

Evaluation (Rubric)	Ideal Level of Achievement (Very Good)	Standard Level of Achievement (Good)	Unacceptable Level of Achievement (Fail)
	Ideal Level of Achievement (Very Good)	Standard Level of Achievement (Good)	Unacceptable Level of Achievement (Fail)
can implement programs accurately based on algorithms that use arrays and other basics of numerical calculation.	can implement programs accurately based on algorithms that use arrays and other basics of numerical calculation.	can implement simple programs based on algorithms that use arrays and other basics of numerical calculation.	Can't implement programs based on algorithms that use arrays and other basics of numerical calculation.
can explain the solution of nonlinear equations and use it to solve engineering problems.	can explain the solution of nonlinear equations accurately and use it to solve engineering problems.	can explain the solution of nonlinear equations	can't explain the solution of nonlinear equations.
can explain numerical integration methods and use them to solve engineering problems.	can explain numerical integration methods and use them to solve engineering problems precisely.	can explain numerical integration methods	can't explain numerical integration methods
Can explain the solution of simultaneous equations based on matrix operations and use it to solve engineering problems.	can explain the solution of simultaneous equations based on matrix operations accurately and use it to solve engineering problems.	can explain the solution of simultaneous equations based on matrix operations.	can't explain the solution of simultaneous equations based on matrix operations.
Can explain the solution methods for differential equations and use them to solve engineering problems	can explain the solution methods for ordinary differential equations precisely and use them to solve engineering problems correctly.	can explain the solution methods for ordinary differential equations	can't explain the solution methods for ordinary differential equations

Relationship with Learning Outcomes
M(4) Ability to design and develop the software for control robotic/ mechatronic systems.
Please change
Please change

Teaching Method	
Outline:	Lecture and Practice.
Class Format:	Lecture, practice and homework assignments.
Please Note :	Students require to ask any questions after sufficient self-learning.

Course Plan

Semester 1	Contents and Method of Course	Goals	Related MCC
	Guidance and review of "Programming 1-5"	Review of Python programming studied in "Programming 1-5"	\(I\-D 1 1 \(I\-D 1 2 \(I\-D 1 3 \(I\-D 1 4 \(I\-D 1 5
	Numerical error / Numerical integration	Explaining what numerical error is in numerical calculation and implementing numerical integration algorithms (trapezoidal method)	V-A 7 161 V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Root-finding of non-linear equation (1)	Explaining what bisectional method and Regula Falsi method to obtain roots of equations python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Root-finding of non-linear equation (2)	Explaining the algorithms of the Newton-Raphson method and implementing python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Root-finding of non-linear equation(3)	Solving engineering problems using root-finding methods and explaining the differences among the root finding methods.	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical methods for systems of linear algebraic equation: Iterative method (1)	Explaining the algorithm of the Gauss elimination method and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Preparing mid-term examination		
	Mid-term examination		
	Numerical methods for systems of linear algebraic equation: Iterative method (2)	Explaining the algorithm of the LU decomposition method and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical methods for systems of linear algebraic equation: Iterative method (3)	Explaining the algorithm of pivoting in Gauss elimination and LU decomposition methods and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical methods for systems of linear algebraic equation: Iterative method (4)	Explaining the algorithm of Doolittle's decomposition (tridiagonal coefficient matrix) and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical methods for systems of linear algebraic equation: Direct method	Explaining the algorithms of Gauss-Seidel and SOR methods and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical interpolation (1)	Explaining the algorithm of Lagrange's interpolation method and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical interpolation (2)	Explaining the algorithm of Lagrange's interpolation method and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Curve fitting	Explaining the algorithm of least-square fitting and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Numerical differentiation	Explaining the algorithms of finite differential approaches and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Partial differential equation	Explaining the algorithms of partial differential equations using finite differential approaches and implementing Python code	V-A 7 162 V-A 7 163 V-A 7 164 V-D 7 89 V-D 7 90 V-D 7 91
	Preparing final examination	Review and summarize learning after the mid-term exam	
	Final examination		
	Return exam papers and feedback	Review and summarize learning during this course	

Do not

	Examination	Quiz	Mutual Evaluations between students	Report	Portfolio	Other
Basic Ability	20	15		15		
Technical Ability	20	15		15		
Interdisciplinary Ability						