

久留米工業高等専門学校		開講年度	令和06年度 (2024年度)		授業科目	計算数理リテラシー	
科目基礎情報							
科目番号	3E13			科目区分	専門 / 必修		
授業形態	講義			単位の種別と単位数	履修単位: 1		
開設学科	電気電子工学科			対象学年	3		
開設期	前期			週時間数	2		
教科書/教材	教科書: [1] 河村哲也 著『数値計算 (第2版)』, インデックス出版. [2] 杉本大一郎 著『使える数理リテラシー』, 勁草書房.						
担当教員	山口 崇						
到達目標							
1. Pythonコーディングを数学などさまざまな計算に応用して活用できる. 2. NumPyとSciPyを公式ドキュメントから理解し活用できる. 3. Pythonの計算ツールや可視化ツールを活用できる.							
ルーブリック							
	理想的な到達レベルの目安			標準的な到達レベルの目安		未到達レベルの目安	
Pythonの計算への活用	問題に応じて十分に適切なコードを作成できる.			問題に応じてある程度適切なコードを作成できる.		適切なコードを作成できない.	
NumPy, SciPyの活用	公式ドキュメントを理解して適切に活用できる.			公式ドキュメントをあまり理解できないが活用できる.		公式ドキュメントを理解できず活用できない.	
計算・可視化ツールの活用	計算ツールと可視化ツールを十分に知り活用できる.			計算ツールと可視化ツールをある程度活用できる.		計算ツールと可視化ツールを活用できない.	
学科の到達目標項目との関係							
1 JABEE B-1							
教育方法等							
概要	「計算数理」と「数理リテラシー」のアウトラインを理解しながら、「計算の道具としてのPython」すなわち、計算を必要とするさまざまな問題解決の場面でPythonを活用できるようになることが目標である。この科目では、数学、物理、電気回路などの問題を例に、NumPyとSciPyを基本ツールとして、場面に応じてそのほかの計算や可視化のパッケージも組み合わせて活用する技法を身につける。						
授業の進め方・方法	教科書は要点のみ参考にしつつあまり深入りせず、概略の理解により実用性重視で進める。数学、物理、電気回路などからも題材を採る。Pythonで計算するなら「車輪の再発明」で足踏みせず、基本はNumPyとSciPyを活用することである。SymPyや可視化ツールも知るべきである。ネット上では直接「公式ドキュメント」を読みに行くことを習慣づけよう。時間の限られた授業で学べることは少なく、あえて基本的なガイドと簡単なヒントしか提示しない。NumPyもSciPyも公式ドキュメント、中でも「ユーザーガイド」「APIリファレンス」を直接読まずには、いつまでも正しく使えるようにはならない。全回を通じて情報のモラルとサイバーセキュリティの基本を学ぶ。						
注意点	評価方法 (1) 課題 (小試験, レポート, 演習など) 100%で評価する。 (2) 評価基準: 60点以上を合格とする。 (3) 定期試験, 再試験は実施しない。 前回までの授業内容について復習しておくこと。課題等は必ず期限までに提出すること。 参考書 [1] Veit Steinkamp 著『Der Python-Kurs für Ingenieure und Naturwissenschaftler』, Rheinwerk Verlag. [2] Svein Linge, Hans Petter Langtangen 著『Programming for Computations – Python』, Springer. (オープンアクセス https://doi.org/10.1007/978-3-030-16877-3) [3] Jake VanderPlas 著『Pythonデータサイエンスハンドブック』, オライリー・ジャパン. [4] Bill Lubanovic 著『入門 Python3』, オライリー・ジャパン. [5] 高専機構 情報セキュリティ教材『K-SEC情報モラル教材』(高専機構「情報システムユーザーガイドライン」「低学年教材」)。						
授業の属性・履修上の区分							
☐ アクティブラーニング		☒ ICT 利用		☒ 遠隔授業対応		☐ 実務経験のある教員による授業	
授業計画							
		週	授業内容	週ごとの到達目標			
前期	1stQ	1週	Pythonにおける数値の取扱い	整数, 実数, 複素数と数学関数の取扱いを理解する.			
		2週	NumPy [1] 関数電卓としてのPython	NumPyのなりたちを理解し, 数学関数の使い方を理解する. ⇨ “API reference” → ‘Mathematical functions’			
		3週	NumPy [2] NumPyの配列	Python標準「リスト」とNumPy「配列」の共通点と違いを理解し使い分ける. ⇨ “User guide” → ‘NumPy fundamentals’			
		4週	NumPy [3] 複素数と電気数学	NumPyにおける複素数の取扱いを理解し, 電気数学に応用する. ⇨ “API reference” → ‘Mathematical functions’			
		5週	NumPy [4] 配列の生成と操作	NumPyの多次元配列を生成し操作するさまざまな方法を理解する. ⇨ “User guide” → ‘NumPy fundamentals’, “API reference” → ‘Array creation routines’, ‘Array manipulation routines’			
		6週	NumPy [5] 配列の論理演算	NumPyの配列に対する強力な論理演算機能を理解し活用する. ⇨ “User guide” → ‘NumPy: the absolute basics for beginners’, “API reference” → ‘Logic functions’			
		7週	NumPy [6] ベクトル, 行列	NumPyの配列を数学のベクトルや行列として取り扱う方法を理解し活用する. ⇨ “User guide” → ‘NumPy fundamentals’, “API reference” → ‘Arithmetic, matrix multiplication, and comparison operations’			

2ndQ	8週	NumPy [7] 線形代数	NumPyでのより高度な行列計算の方法を理解し、回路方程式の解析などに活用する。☞ “API reference” → ‘Linear algebra’
	9週	NumPy [8] 確率、統計、乱数	NumPyを確率や統計データの解析、乱数を用いるシミュレーションなどに活用する。☞ “API reference” → ‘Statistics’; ‘Random sampling’
	10週	SciPy [1] 科学計算	SciPyの多彩な機能を概観し、基本的な取扱いを理解し活用する。☞ “SciPy User Guide” → ‘Introduction’
	11週	SciPy [2] 行列演算	SciPyの代表的な機能を理解し活用する。☞ “SciPy User Guide” → ‘Introduction’
	12週	SymPy [1] 数式処理の基本	SymPyによる文字式の基本的な取扱いを理解する。
	13週	SymPy [2] 微分・積分	SymPyを微分・積分やその他の高等数学の計算に活用する。
	14週	可視化ツール	Pythonの代表的な可視化パッケージMatplotlibの概要を理解する。
	15週	情報セキュリティ	情報のモラルとサイバーセキュリティの基本を理解する。
	16週		

モデルコアカリキュラムの学習内容と到達目標

分類		分野	学習内容	学習内容の到達目標	到達レベル	授業週
基礎的能力	工学基礎	情報リテラシー	情報リテラシー	情報を適切に収集・処理・発信するための基礎的な知識を活用できる。	3	前1,前2,前10,前12,前14
				論理演算と進数変換の仕組みを用いて基本的な演算ができる。	3	前2,前3,前4,前6
				コンピュータのハードウェアに関する基礎的な知識を活用できる。	3	前2,前3,前4
				同一の問題に対し、それを解決できる複数のアルゴリズムが存在していることを知っている。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				与えられた基本的な問題を解くための適切なアルゴリズムを構築することができる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				任意のプログラミング言語を用いて、構築したアルゴリズムを実装できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				情報セキュリティの必要性および守るべき情報を認識している。	3	前15
				個人情報とプライバシー保護の考え方についての基本的な配慮ができる。	3	前15
				インターネット(SNSを含む)やコンピュータの利用における様々な脅威を認識している	3	前15
専門的能力	分野別の専門工学	情報系分野	プログラミング	代入や演算子の概念を理解し、式を記述できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				プロシージャ(または、関数、サブルーチンなど)の概念を理解し、これらを含むプログラムを記述できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				変数の概念を説明できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				データ型の概念を説明できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14

				制御構造の概念を理解し、条件分岐を記述できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				制御構造の概念を理解し、反復処理を記述できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				与えられた問題に対して、それを解決するためのソースプログラムを記述できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				ソフトウェア生成に必要なツールを使い、ソースプログラムをロードモジュールに変換して実行できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				与えられたソースプログラムを解析し、プログラムの動作を予測することができる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				ソフトウェア開発に利用する標準的なツールの種類と機能を説明できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				要求仕様に従って、標準的な手法により実行効率を考慮したプログラムを設計できる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				要求仕様に従って、いずれかの手法により動作するプログラムを設計することができる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14
				要求仕様に従って、いずれかの手法により動作するプログラムを実装することができる。	3	前2,前3,前4,前5,前6,前7,前8,前9,前10,前11,前12,前13,前14

評価割合							
	試験	発表	相互評価	態度	ポートフォリオ	課題	合計
総合評価割合	0	0	0	0	0	100	100
基礎的能力	0	0	0	0	0	10	10
専門的能力	0	0	0	0	0	90	90
分野横断的能力	0	0	0	0	0	0	0