

津山工業高等専門学校		開講年度	令和05年度 (2023年度)		授業科目	プログラミング応用	
科目基礎情報							
科目番号	0094			科目区分	専門 / 選択		
授業形態	講義			単位の種別と単位数	学修単位: 2		
開設学科	総合理工学科(機械システム系)			対象学年	4		
開設期	後期			週時間数	2		
教科書/教材	教科書：特に指定しない。授業資料はPDF等で配布する。						
担当教員	川波 弘道						
到達目標							
学習目的： プログラミングの基礎を踏まえ、プログラム作成能力、プログラム読解能力を身に付ける。							
到達目標：理論的な裏付けに留意しながらプログラムが記述でき、他者の書かれたプログラムを読みこなせる。 1. 適切な変数、データ型を使ってプログラムが記述できる。 2. プロシージャ(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が記述できる。 3. 与えられた簡単な問題に対して、それを解決するためのソースプログラムが書ける。 4. 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。							
ルーブリック							
	優		良		可		不可
評価項目1	適切な変数、データ型を使ってプログラムが記述できる。		適切な変数またはデータ型を使ってプログラムが記述できる。		変数、データ型を使ってプログラムが記述できる。		変数、データ型を使ってプログラムが記述できない。
評価項目2	プロシージャ(関数、サブルーチンなど)の概念を理解し、これらを含む可読性の高いプログラムが記述できる。		プロシージャ(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が記述できる。		プロシージャ(関数、サブルーチンなど)の概念を理解している。		プロシージャ(関数、サブルーチンなど)の概念を理解していない。
評価項目3	与えられた簡単な問題に対して、それを解決するための可読性の高いソースプログラムが書ける。		与えられた簡単な問題に対して、それを解決するためのソースプログラムが書ける。		与えられた簡単な問題に対して、それを解決するためのソースプログラムのフローチャートが書ける。		与えられた簡単な問題に対して、それを解決するためのソースプログラムのフローチャートが書けない。
評価項目4	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可読性の高いプログラムを設計できる。		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。		要求仕様に従って、標準的な手法によりプログラムを設計できる。		要求仕様に従って、標準的な手法によりプログラムを設計できない。
学科の到達目標項目との関係							
教育方法等							
概要	一般・専門の別：専門 学習の分野：情報・制御 基礎となる学問分野：情報学／計算基盤／ソフトウェア 学習の分野：情報システム・プログラミング・ネットワーク 学習教育目標との関連：本科目は「③基盤となる専門性の深化」に相当する科目である。 授業の概要：はじめにプログラミングに必要な文法規則を確認し、簡単なプログラムの読み方や書き方について学ぶ。信号処理やモデル学習といった応用的なプログラム作成に取り組む。また、他人が読みやすいプログラム作成、プログラムの説明能力の育成のため、プレゼンテーションや意見交換なども授業に取り入れる。						
授業の進め方・方法	授業の方法：プレゼンテーションによる講義とプログラミング演習を中心に授業を進める。理解が深まるようレポートを課す。学生自身による発表を指示する場合がある。前半は提携校教員による英語によるオンライン授業を予定している。 成績評価方法：定期試験の結果（60％）と学習に取り組む態度（演習、とりくみ姿勢、発表）（40％）で評価する。割合については調整することがある。 期末段階の成績が60点未満の者には、出席状況や授業態度が良好であれば、事前指示を与えた上で再試験を実施する。実施後の評価は60点を超えないものとする。						
注意点	履修上の注意：プログラミングに関する基礎知識があることが望ましい。本科目を選択した者は、学年の課程修了のために履修（欠課時間数が所定授業時間数の3分の1以下）が必須である。また、本科目は「授業時間外の学修を必要とする科目」である。当該授業時間と授業時間外の学修を合わせて、1単位あたり45時間の学修が必要である。 履修のアドバイス：授業資料と課題はPDFファイルで公開する。授業中の課題について自分で調査し、既存研究についての理解を深め自分でその課題に対するプログラムを作成する。授業時間外にもプログラムを記述する時間が必要である。 基礎科目：情報リテラシー（1年）、総合理工基礎（1）、計算科学（3）、アルゴリズムとデータ構造（3） 関連科目：システムプログラミング（5年）、卒業研究（5） 受講上のアドバイス：プログラミングは自主学習によって実力をつけることが可能である。自宅等でもプログラミングできる環境があることが望ましい。課題は自分で行うこと。友人のコピーなどの不正が発覚した場合は厳しい措置をとる。授業開始前に行う出席確認に遅れた者は遅刻として扱う。遅刻は半時限分の欠課として扱う。ただし、授業開始から45分以上を経過しての遅刻については欠席として扱う。						
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	
履修選択							
授業計画							
		週	授業内容			週ごとの到達目標	
後期	3rdQ	1週	ガイダンス プログラミング環境の確認とC言語の簡単な復習			プログラミング環境と知識の確認	

		2週	・最大公約数・最小公倍数を求めるプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		3週	・円周率の近似値を求めるプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		4週	・素数判定のプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		5週	・ファイルの読み込みと統計処理を行うプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		6週	・基本的な信号処理を行うプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		7週	成果物の発表	作成したプログラムについて他者に要点を説明できる。
		8週	（後期中間試験）	学習内容の確認
	4thQ	9週	後期中間試験答案返却と解答解説	学習内容の確認と振り返り
		10週	・エルガマル暗号のプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		11週	・RSA暗号のプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		12週	・テーブルゲームのプログラム	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		13週	・テーブルゲームのプログラム（継続）	適切な変数，データ型を使ってプログラムが記述できる。プロシージャ（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が記述できる。与えられた簡単な問題に対して，それを解決するためのソースプログラムが書ける。
		14週	成果物の発表	作成したプログラムについて他者に要点を説明できる。
		15週	（後期末試験）	学習内容の確認
		16週	後期末試験の答案返却と試験解説	学習内容の確認と振り返り

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
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。	4	
				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。	4	
				変数の概念を説明できる。	4	
				データ型の概念を説明できる。	4	
				制御構造の概念を理解し、条件分岐を記述できる。	4	
				制御構造の概念を理解し、反復処理を記述できる。	4	
				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。	4	
				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。	4	
				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。	4	
				主要な言語処理プロセッサの種類と特徴を説明できる。	4	
				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。	4	
				プログラミング言語は計算モデルによって分類されることを説明できる。	4	
				主要な計算モデルを説明できる。	4	
				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。	4	
				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。	4	

				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。	4	
				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実装できる。	4	
			ソフトウェア	アルゴリズムの概念を説明できる。	3	
				与えられたアルゴリズムが問題を解決していく過程を説明できる。	3	
				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うることを説明できる。	3	
				整列、探索など、基本的なアルゴリズムについて説明できる。	3	
				時間計算量によってアルゴリズムを比較・評価できることを説明できる。	2	
				領域計算量などによってアルゴリズムを比較・評価できることを説明できる。	2	
				コンピュータ内部でデータを表現する方法(データ構造)にはバリエーションがあることを説明できる。	3	
				同一の問題に対し、選択したデータ構造によってアルゴリズムが変化しうることを説明できる。	3	
				リスト構造、スタック、キュー、木構造などの基本的なデータ構造の概念と操作を説明できる。	3	
				リスト構造、スタック、キュー、木構造などの基本的なデータ構造を実装することができる。	3	
				ソフトウェアを中心としたシステム開発のプロセスを説明できる。	4	
				ソースプログラムを解析することにより、計算量等のさまざまな観点から評価できる。	3	
				同じ問題を解決する複数のプログラムを計算量等の観点から比較できる。	3	

評価割合			
	試験	演習（発表を含む）	合計
総合評価割合	60	40	100
基礎的能力	0	0	0
専門的能力	60	40	100
分野横断的能力	0	0	0