

熊本高等専門学校		開講年度	令和04年度 (2022年度)		授業科目	プログラミング入門	
科目基礎情報							
科目番号	0023			科目区分	一般 / 必修		
授業形態	授業			単位の種別と単位数	履修単位: 1		
開設学科	生物化学システム工学科			対象学年	2		
開設期	前期			週時間数	2		
教科書/教材	使用						
担当教員	村田 美友紀,小島 俊輔,藤本 洋一						
到達目標							
1. Processingを用いてプログラムを作成し実行することができる。 2. マウスやキーボード入力、コンソールやグラフィックス出力ができる。 3. 代入・演算・制御構造・関数の処理を記述できる。 4. 探索や整列などのアルゴリズムを理解できる。							
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	
Processingを用いてプログラムを作成し実行することができる		与えられたすべての課題に対して、変数名やインデントなどを考慮したプログラムをProcessingで記述し実行できる。実行モジュールを作成し実行することができる		与えられた課題の6割以上について、Processingを用いてプログラムを記述し実行することができる		Processingを用いてプログラムを記述し実行することができない	
マウスやキーボード入力、コンソールやグラフィックス出力ができる		マウスやキーボード入力などのイベントを複数組み合わせたプログラムを記述できる。プログラムの実行結果をコンソールやグラフィックスに思い通りに出力できる		マウスやキーボード入力などのイベントを記述し、プログラムの動作を変更できる。実行結果を文字やグラフィックスで出力できる		マウスやキーボード入力などのイベントを記述できない。実行結果を文字やグラフィックスで出力できない	
代入・演算・制御構造・関数の処理を記述できる		代入・演算・制御構造・関数を多数組み合わせ合わせたプログラムを記述できる		代入・演算・制御構造・関数のいくつかを組み合わせた基本的な動作のプログラムを記述できる		代入・演算・制御構造・関数を組み合わせることができない	
探索や整列などのアルゴリズムを理解できる		探索や整列などのアルゴリズムの動作を理解し、動きを図などを描いて説明できる		資料で説明されている探索や整列などのアルゴリズムの動作を理解できる		探索や整列などのアルゴリズムを理解できない	
学科の到達目標項目との関係							
教育方法等							
概要		現代はスマホや家庭電気製品はもちろん、ありとあらゆる製品にコンピュータが内蔵されており、プログラムによって制御されている。将来技術者として活躍する学生諸君にとってプログラムの習得は必須である。本科目はプログラムに関する基本的な知識の習得を目的とする。					
授業の進め方・方法		毎回、配布した資料に従って授業を進める。多くの演習問題を取り入れ、学習内容の修得を目指す。授業の内容で分からないところは質問するなどして解決すること。サンプルプログラムを単に入力するだけでなく、なぜそのようにするのか理由を考えながら演習に取り組むことが重要である。					
注意点		自転車の乗り方を口頭で説明されても自転車には乗れるようにはなりません。同様に、プログラムも教員から習っただけでは作成できるようにはなりません。演習問題を含め多くのプログラムを自分で作成し、数多くの失敗することが目標達成につながります。プログラムが正しく動作しない場合でも、誤りを見つけて修正する作業は、皆さんの問題発見・解決能力を磨くこととなります。質問や相談は、直接、あるいはメールで随時受け付けます。また教員室前に所在を示し、在室時間等も掲示しておくので活用してください。					
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	ガイダンス、プログラミングの作成から実行まで、コンソール出力		・ Processingプログラムの作成から実行までの一連の操作ができる ・ プログラムの実行モジュールを作成できる ・ setup(),draw()を利用することができる		
		2週	グラフィック出力		・ 図形のプロパティを指定して任意の場所に基本図形を描画できる ・ 座標変換関数を理解し利用できる		
		3週	変数と型、演算子、クラスの問題		・ 格納するデータに適した変数があることを理解する ・ 代入演算子、演算子を使ったコードが書ける ・ クラスの概念を理解しクラスライブラリを使用できる		
		4週	繰り返し		・ for文をwhile文を使った反復処理のコードが書ける ・ for文、while文の違いを説明できる ・ デバッグを利用できる		
		5週	分岐		・ if-else文を使った選択処理のコードが書ける ・ 比較演算子、論理演算子を使って条件式が書ける ・ フローチャートが読める		
		6週	配列		・ 1次元配列を用いたプログラムを記述することができる、2次元配列の概念を説明することができる		
		7週	インタラクティブプログラミング		・ イベントドリブンの考え方を理解し、マウスやキーボード入力を用いたプログラムが記述できる		
		8週	[中間試験]				
	2ndQ	9週	ファイル入出力とCSVファイル		・ ファイル入出力を介し他のアプリケーションと連携できる		
		10週	関数定義と使い方		・ 関数を定義し使用することができる		

		11週	基本的なアルゴリズムの例	・最大値，合計，平均値などの基本的なアルゴリズムのプログラムを記述できる
		12週	探索アルゴリズム	・線形探索や配列インデックスによる探索手法を理解する ・バイナリサーチアルゴリズムを理解しリニアサーチとの違いを説明できる
		13週	整列アルゴリズム（１）	・バブルソートの動作を理解できる ・プログラムの一部を修正し正順・逆順のソートに変更することができる
		14週	整列アルゴリズム（２）	・クイックソートの考え方を理解し，バブルソートとの差を説明できる
		15週	総合演習	
		16週	期末試験の返却と解説	

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	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知っている。	2	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。	1	前5,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。	1	前11,前12,前13,前14
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3
				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10
				変数の概念を説明できる。	3	前3
				データ型の概念を説明できる。	3	前3
				制御構造の概念を理解し、条件分岐を記述できる。	3	前5,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4,前6,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。	2	前11
				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。	2	前1
				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。	2	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。	2	前15
				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。	2	前15
			ソフトウェア	アルゴリズムの概念を説明できる。	2	前12
				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。	2	前12,前13,前14
				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。	2	前12,前13,前14
				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。	2	前12,前13,前14
	分野別の工学実験・実習能力	情報系分野【実験・実習能力】	情報系【実験・実習】	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、標準的な開発ツールや開発環境を利用して記述できる。	2	前1,前4
				ソフトウェア生成に利用される標準的なツールや環境を使い、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。	2	前1
				ソフトウェア開発の現場において標準的とされるツールを使い、生成したロードモジュールの動作を確認できる。	2	前1,前4
				フローチャートなどを用いて、作成するプログラムの設計図を作成することができる。	2	前15
				問題を解決するために、与えられたアルゴリズムを用いてソースプログラムを記述し、得られた実行結果を確認できる。	2	前15
				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し、適切な実行結果を得ることができる。	2	前15

評価割合

	試験	課題	小テスト	合計
総合評価割合	70	20	10	100
基礎的能力	40	10	5	55
専門的能力	30	10	5	45
分野横断的能力	0	0	0	0