

一関工業高等専門学校		開講年度	平成30年度 (2018年度)		授業科目	プログラミング演習	
科目基礎情報							
科目番号	0003			科目区分	専門 / 必修		
授業形態	演習			単位の種別と単位数	履修単位: 2		
開設学科	未来創造工学科 (情報・ソフトウェア系)			対象学年	2		
開設期	通年			週時間数	2		
教科書/教材	確かな力が身につくPython「超」入門、確かな力が身につくC#「超」入門 (SBクリエイティブ株式会社)						
担当教員	小保方 幸次,佐藤 建,小池 敦,水津 俊介						
到達目標							
(1) 統合開発環境を使ってプログラムを開発できる。 (2) Pythonを利用した基本的なプログラミングができる。 (3) C#言語を用いたプログラムを作成できる。 (4) オブジェクト指向を用いたプログラムを作成できる。							
【教育目標】C, D							
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		
(1) プログラム作成に関する基礎知識		用途に沿った適切なプログラムを設計、開発できる。	統合開発環境を使ってプログラムを開発できる。		プログラムの開発ができない。		
(2) Pythonプログラミング		Pythonの標準、外部モジュールを利用して自らプログラムの開発ができる。エラー箇所の分析ができる。	Pythonを利用した基本的なプログラミングができる。		Pythonを利用したプログラミングができない。		
(3) C#プログラミング		C#言語の応用的なプログラムを自ら作成できるようになる。	C#言語を用いたプログラムを作成できる。		C#言語を用いたプログラムを作成できない。		
(4) オブジェクト指向プログラミング		オブジェクト指向を用いて応用的なプログラムを自ら作成できるようになる。	オブジェクト指向を用いたプログラムを作成できる。		オブジェクト指向を用いたプログラムを作成できない。		
学科の到達目標項目との関係							
教育方法等							
概要	本授業ではプログラミングの入門者を対象にプログラミングの演習を行う。前期はスクリプト言語Pythonによりプログラミングの基礎的な演習を行う。Linux OS上での操作や演習により基礎的なプログラムの制御、データ構造を習得する。Pythonの標準ライブラリ、外部ライブラリを利用したプログラム製作を行う。後期はC#言語を用いて言語を理解するための演習やオブジェクト指向を活用できるようになるための演習を行う。						
授業の進め方・方法	演習室において実際のコンピュータ上でプログラムを動作させ、自分で確認する。前期はPythonを利用したLinux OS上でのプログラム開発を行う。統合開発環境を利用して実際にプログラムを動かし、最終的には自主的に計画してプログラム製作を行う。後期はWindows OS上でVisual Studioを用いてプログラム開発を行う。毎週の授業内容を理解するための小課題と総合的なプログラミング能力を養うための総合課題を課す。						
注意点	本授業では多くの課題が課されるが、提出期限を守って提出すること。 【事前学習】 「授業内容・方法」に対応する教科書や資料の内容を事前に読んでおくこと。 【評価方法・評価基準】 総合課題(60%)、小課題(40%)で評価する。詳細は第1回目の授業で告知する。 基本的なプログラミング能力、Python、C#言語を利用したプログラム作成能力、オブジェクト指向プログラミングを実践する能力を評価する。 総合評価60点以上を単位修得とする。						
授業計画							
		週	授業内容		週ごとの到達目標		
前期	1stQ	1週	Linux OSの操作、Pythonの開発環境構築 (1)		Linux OS、CUIを操作ができる。Pythonのインタラクティブシェルや開発環境を操作できる。		
		2週	Linux OSの操作、Pythonの開発環境構築 (2)		Linux OS、CUIを操作ができる。Pythonのインタラクティブシェルや開発環境を操作できる。		
		3週	データ型と演算、ファイル操作		算術演算、比較演算、変数を使用したプログラミングができる。ファイル操作を行える。		
		4週	リスト、タプル、辞書型、集合型 (1)		リスト、タプル、辞書型、集合型を扱うことができる。		
		5週	リスト、タプル、辞書型、集合型 (2)		リスト、タプル、辞書型、集合型を扱うことができる。		
		6週	条件分岐、繰り返し、関数		条件分岐、繰り返し、関数を利用したプログラミングができる。		
		7週	クラス、継承、標準ライブラリ (1)		クラスや継承、標準ライブラリを利用できる。		
		8週	クラス、継承、標準ライブラリ (2)		クラスや継承、標準ライブラリを利用できる。		
	2ndQ	9週	外部ライブラリの活用 (1)		外部ライブラリを利用した画像処理やwebアクセスができる。		
		10週	外部ライブラリの活用 (2)		外部ライブラリを利用した画像処理やwebアクセスができる。		
		11週	GUIプログラミング		Pythonによる簡単なGUIプログラミング、アプリケーション開発ができる。		
		12週	プログラミング開発実習 (1)		Pythonの標準ライブラリ、外部ライブラリを利用したプログラミングができる。		
		13週	プログラミング開発実習 (2)		Pythonの標準ライブラリ、外部ライブラリを利用したプログラミングができる。		
		14週	開発したプログラムの発表		開発したプログラムについて設計や機能などを説明できる。		

後期		15週	前期のまとめ	Pythonによるプログラム開発を行える。
		16週		
	3rdQ	1週	C# 言語の開発環境構築	Visual Studio上で新規プロジェクト作成やプログラム実行ができる。
		2週	制御構造を用いたプログラミング	制御構造を利用したプログラムを作成できる。
		3週	複数のデータを扱うプログラミング	リストやディクショナリーを利用したプログラムを作成できる。
		4週	メソッドを活用したプログラミング	メソッドを利用したプログラムを作成できる。
		5週	迷路ゲームの作成（１）	与えられた課題を解決するために配列やコレクションを活用できる。
		6週	迷路ゲームの作成（２）	与えられた課題を解決するためにメソッドを活用できる。
		7週	オブジェクト指向プログラミング（１）	クラスを利用したプログラムを作成できる。
		8週	オブジェクト指向プログラミング（２）	コンストラクタやプライベート変数を用いたプログラムを作成できる。
	4thQ	9週	オブジェクト指向プログラミング（３）	ポリモーフィズムを利用したプログラムを作成できる。
		10週	描画アプリの作成	クラスを活用した描画オブジェクトの設計ができる。
		11週	windowsアプリケーション	簡易的なwindowsアプリケーションを作成できる。
		12週	総合演習（１）	オブジェクト指向を活用したプログラム設計ができる。
		13週	総合演習（２）	オブジェクト指向を活用したプログラム設計ができる。
		14週	制作物のプレゼンテーション	制作物についてのプレゼンテーションができる。
		15週	後期のまとめ	後期に学んだ事項を俯瞰的に理解できる。
		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	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	
				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。	3	
				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	
				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	
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	
				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	
				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。	2	
				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。	2	
				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	
			ソフトウェア	アルゴリズムの概念を説明できる。	2	
				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	
				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	
				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	
				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。	2	
				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。	2	
				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。	2	

				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。	2	
				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。	2	
				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。	2	
				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。	2	
			システムプログラム	コンパイラの役割と仕組みについて説明できる。	3	
	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。	3	
				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。	3	
				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。	3	
				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。	3	
				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。	3	
				標準的な開発ツールを用いてプログラミングするための開発環境構築ができる。	3	
				要求仕様にあったソフトウェア(アプリケーション)を構築するために必要なツールや開発環境を構築することができる。	3	
				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。	3	
評価割合						
			総合課題	小課題	合計	
総合評価割合			60	40	100	
基礎的能力			60	40	100	