

茨城工業高等専門学校		開講年度	令和03年度 (2021年度)		授業科目	プログラミングⅡ	
科目基礎情報							
科目番号	0039			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: 2		
開設学科	国際創造工学科 情報系			対象学年	3		
開設期	通年			週時間数	2		
教科書/教材	配付資料						
担当教員	小飼 敬						
目的・到達目標							
1. ソフトウェア開発で利用するツールを使って応用プログラムを開発する。 2. 主要な計算モデルについて理解する。 3. コンピュータ上における誤差について理解する。 4. 主要な数値計算アルゴリズムに基づいたプログラムを実装する。 5. オブジェクト指向プログラミングの基礎について理解する。 6. 代表的なソフトウェア開発方法論に基づいてソフトウェアを設計する。							
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	
評価項目1	ソフトウェア開発で利用するツールについて理解できると共に、これらを使って応用プログラムを開発できる。			ソフトウェア開発で利用するツールを使って応用プログラムを開発できる。		ソフトウェア開発で利用するツールを使って応用プログラムを開発できない。	
評価項目2	主要な計算モデルについて理解できると共に、計算モデルに基づいたプログラムを作成できる。			主要な計算モデルについて理解し説明できる。		主要な計算モデルについて理解できず、説明もできない。	
評価項目3	コンピュータ上における誤差について理解できると共に、誤差を考慮したプログラムを作成できる。			コンピュータ上における誤差について理解し説明できる。		コンピュータ上における誤差について理解できず、説明もできない。	
評価項目4	主要な数値計算アルゴリズムについて理解できると共に、これに基づいたプログラムを作成できる。			主要な数値計算アルゴリズムについて理解し説明できる。		主要な数値計算アルゴリズムについて理解できず、説明もできない。	
評価項目5	オブジェクト指向プログラミングの基礎について理解し説明できると共に、これらを活用してアプリケーションを作成できる。			オブジェクト指向プログラミングの基礎について理解し説明できる。		オブジェクト指向プログラミングの基礎について理解し説明できない。	
評価項目6	代表的な開発方法論について理解できると共に、これに基づいたソフトウェア開発を実践できる。			代表的な開発方法論について理解し説明できる。		代表的な開発方法論について理解できず、説明もできない。	
学科の到達目標項目との関係							
学習・教育到達度目標 (A)							
教育方法等							
概要	プログラミングを様々な問題に応用する方法について学ぶ。特に、開発ツール, 計算モデル, コンピュータ上の誤差, 数値計算アルゴリズム, オブジェクト指向プログラミング, ソフトウェア開発方法論の基礎について、コンピュータを使った演習を通じて理解を深める。						
授業の進め方と授業内容・方法	講義は各自のPCにインストールした開発環境等を使いながら進めるので毎回必ずPCを持参すること。						
注意点							
授業の属性・履修上の区分							
<input type="checkbox"/> アクティブラーニング		<input type="checkbox"/> ICT 利用		<input type="checkbox"/> 遠隔授業対応		<input type="checkbox"/> 実務経験のある教員による授業	
授業計画							
		週	授業内容・方法	週ごとの到達目標			
前期	1stQ	1週	ソースコードの管理と開発環境の導入	リポジトリ、バージョン管理の概念、これらを活用した操作について理解し、ソフトウェア開発で利用できる。			
		2週	メモリの動的確保	メモリの動的確保の基本的な考え方を理解し、これを使ったプログラムを作成できる。			
		3週	クラス (1)	クラスとインスタンスの考え方について理解し、これらを使ったプログラムを作成できる。			
		4週	クラス (2)	コンストラクタ、オーバーロードの考え方について理解し、これらを使ったプログラムを作成できる。			
		5週	クラス (3)	クラス型変数について理解し、これらを使ったプログラムを作成できる。			
		6週	クラス (4)	has-a関係について理解し、これらを使ったプログラムを作成できる。			
		7週	(中間試験)				
		8週	アクセス制御	アクセス制御と可視性の考え方について理解し、アクセス修飾子を使ったプログラムを作成できる。			
	2ndQ	9週	カプセル化・情報隠蔽	カプセル化と情報隠蔽の考え方について理解し、これらを使った簡単なプログラムを作成できる。			
		10週	継承 (1)	継承の考え方、継承における可視性、is-a関係について理解し、これらを使ったプログラムが作成できる。			
		11週	継承 (2)	オーバーライド、ポリモフィズムの考え方について理解し、これらを使ったプログラムを作成できる。			

後期		12週	GUI (1)	基本的なGUIの概念について理解し、説明できる。	
		13週	GUI (2)	基本的なGUIの部品について理解し、GUIを持ったプログラムを作成できる。	
		14週	GUI (3)	イベント処理の仕組みについて理解し、イベント処理が伴うプログラムを作成できる。	
		15週	(期末試験)		
		16週	総復習		
	3rdQ	1週	コマンドライン引数	コマンドライン引数の仕組みを理解し、これを活用したプログラムを作成できる。	
		2週	抽象クラス	抽象クラスの仕組みを理解し、これを使ったプログラムを作成できる。	
		3週	インタフェース	インタフェースの仕組みを理解し、これを使ったプログラムを作成できる。	
		4週	例外処理	例外処理の仕組みを理解し、これを使ったプログラムを作成できる。	
		5週	コレクション	コレクションの仕組みを理解し、これを使ったプログラムを作成できる。	
		6週	ジェネリクス	ジェネリクスの仕組みを理解し、これを使ったプログラムを作成できる。	
		7週	(中間試験)		
		8週	計算モデル (1)	計算モデルの考え方について理解し、説明できる。	
		4thQ	9週	計算モデル (2)	オートマトンの考え方、ステートマシンを理解し、これに基づいたプログラムを作成できる。
			10週	数値計算 (1)	コンピュータ上での誤差について理解し、これを考慮したプログラムを作成できる。
			11週	数値計算 (2)	主要な数値計算アルゴリズムについて理解し、これに基づいたプログラムを実装する。
			12週	言語処理系	言語プロセッサの概要と種類、プログラミングパラダイムについて理解し、説明できる。
			13週	ソフトウェア開発方法論 (1)	ソフトウェアライフサイクル、UMLの概要とソフトウェア開発における役割について理解し、これに基づいた代表的なソフトウェア開発方法論について説明できる。
			14週	ソフトウェア開発方法論 (2)	UMLによる静的構造の表記方法について理解し、これを用いてソフトウェアの静的構造を記述し、これに基づいて実装できる。
			15週	(期末試験)	
			16週	総復習	

評価割合			
	試験	課題	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