

仙台高等専門学校		開講年度	令和03年度 (2021年度)		授業科目	ソフトウェア工学基礎	
科目基礎情報							
科目番号	0091			科目区分	専門 / 選択		
授業形態	授業			単位の種別と単位数	学修単位: 2		
開設学科	総合工学科 I 類			対象学年	4		
開設期	前期			週時間数	2		
教科書/教材	「Java 本格入門」谷本心, 阪本雄一郎, 岡田拓也, 秋葉誠, 村田賢一郎 (技術評論社), 自作資料を別途配布						
担当教員	藤原 和彦						
到達目標							
1. オブジェクト指向の基本的概念を理解し, ソフトウェア開発に適用することができる。 2. ソフトウェア開発における様々なモデル化手法を理解し, 実践する素養を身に付ける。 3. ソフトウェア開発におけるテストの重要性を理解し, 様々な問題に対し適切な検証を行うことができる。							
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	
オブジェクト指向プログラミング		オブジェクト指向の基本的概念を正しく理解し, ソフトウェア開発において適切に利用することができる。		オブジェクト指向の基本的概念を理解し, ソフトウェア開発に適用することができる。		オブジェクト指向の基本的概念が理解できず, ソフトウェア開発に適用することができない。	
モデル化		ソフトウェア開発における様々なモデル化手法を正しく理解し, 実際の問題に適切に適用することができる。		ソフトウェア開発における様々なモデル化手法を理解し, 実際の問題に適用することができる。		ソフトウェア開発における様々なモデル化手法を, 問題に適用することができない。	
テスト・検証		ソフトウェア開発におけるテストの重要性を正しく理解し, 様々な問題に対し適切な手法で十分な検証を行うことができる。		ソフトウェア開発におけるテストの重要性を理解し, 様々な問題に対し適切な検証を行うことができる。		ソフトウェア開発におけるテストの重要性が理解できず, 問題に対し適切な検証を行うことができない。	
学科の到達目標項目との関係							
JABEE (A) 実践技術者としての高度でかつ幅広い基本的能力・素養							
教育方法等							
概要		ソフトウェアを開発する際の様々な方法, 考え方について, その基礎的な知識を学習する。オブジェクト指向モデルの実装方法としてはJavaを用いることとし, 3年次までに学習したC言語との違いを中心にプログラミング言語の修得もすすめる。					
授業の進め方・方法		それぞれの授業内容について, 要点をまとめた講義を実施し, 例題や課題などに取り組むことでその知識の定着と実践力の修得を図る。 【事前学習】 次回講義の範囲に相当する教科書や配付資料の内容を確認し, 理解をすすめておくこと。 【事後学習】 授業中に提示された課題に取り組み, 必要な報告書等を作成すること。					
注意点		ソフトウェア開発の手順の重要性を十分に理解し, 「動けばいい」という安易な考えを持ち込まないこと。課題等提出物で未提出のものがある場合は合格とならないので十分に留意すること。					
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	Javaの基礎 1	これまでC言語で学習してきたプログラムの基本的制御構造をJavaに置き換え, クラスの概念を用いて適切にプログラムを実装できる。			
		2週	Javaの基礎 2	配列や様々なコレクションを利用したプログラムを実装できる。			
		3週	Javaの基礎 3	例外処理の取り扱いと, マルチスレッドの記述方法を理解し, 適切に実装できる。			
		4週	Javaの基礎 4	JavaにおけるStreamとラムダ式を理解し, 適切に実装できる。			
		5週	計算量	オーダー記法を理解し, 様々なアルゴリズムの計算量を適切に評価できる。			
		6週	開発プロセスモデル	ソフトウェアの様々な開発モデルを理解し, 説明できる。			
		7週	開発プロセスモデル(2)	ソフトウェアの様々な開発モデルを理解し, 説明できる。			
		8週	要求分析	ソフトウェア開発における要求分析の手法を理解し, 説明できる。			
	2ndQ	9週	モデル化手法(1)	ソフトウェア開発における様々な事柄を抽象化し, UML等を用いて記述することができる。			
		10週	モデル化手法(2)	ソフトウェア開発における様々な事柄を抽象化し, UML等を用いて記述することができる。			
		11週	モデル化手法(3)	ソフトウェア開発における様々な事柄を抽象化し, UML等を用いて記述することができる。			
		12週	テスト(1)	ソフトウェアの検証についての考え方を理解し, 問題の種類にあわせたテスト手法を適用できる。			
		13週	テスト(2)	ソフトウェアの検証についての考え方を理解し, 問題の種類にあわせたテスト手法を適用できる。			

		14週	テスト(3)	ソフトウェアの検証についての考え方を理解し，問題の種類にあわせたテスト手法を適用できる。		
		15週	期末試験			
		16週	答案返却			
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	ソフトウェア	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	前7
				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。	4	前5
				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。	4	前5
		コンピュータシステム	システム設計には、要求される機能をハードウェアとソフトウェアでどのように実現するかなどの要求の振り分けやシステム構成の決定が含まれることを説明できる。	3	前11	
			ユーザの要求に従ってシステム設計を行うプロセスを説明することができる。	3	前8	
評価割合						
		試験		課題		合計
総合評価割合		40		60		100
基礎的能力		15		25		40
専門的能力		25		35		60