

石川工業高等専門学校		開講年度	令和02年度 (2020年度)		授業科目	ソフトウェア工学	
科目基礎情報							
科目番号	17050			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	電子情報工学科			対象学年	4		
開設期	通年			週時間数	2		
教科書/教材	講義スライドを配布する / 教材等：玉井「ソフトウェア工学の基礎」岩波書店 / 参考書：鶴保他「ずっと受けたかったソフトウェアエンジニアリングの授業1, 2」翔泳社						
担当教員	越野 亮,富田 堯						
到達目標							
1. ソフトウェアの開発工程を理解できる 2. ソフトウェアの設計技法を理解できる 3. ソフトウェアのテスト技法を理解できる 4. ソフトウェアの開発プロジェクト管理技法を理解できる 5. ソフトウェアを開発することができる							
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	
到達目標項目1	ソフトウェアの開発工程を詳細まで理解できる			ソフトウェアの開発工程の基礎的な内容を理解できる		ソフトウェアの開発工程を理解できない	
到達目標項目2	ソフトウェアの設計技法を詳細まで理解できる			ソフトウェアの設計技法の基礎的な内容を理解できる		ソフトウェアの設計技法を理解できない	
到達目標項目3	ソフトウェアのテスト技法を詳細まで理解できる			ソフトウェアのテスト技法の基礎的な内容を理解できる		ソフトウェアのテスト技法を理解できない	
到達目標項目4	ソフトウェアの開発プロジェクト管理技法を詳細まで理解できる			ソフトウェアの開発プロジェクト管理技法の基礎的な内容を理解できる		ソフトウェアの開発プロジェクト管理技法を理解できない	
到達目標項目5	ソフトウェア工学の技法を用いて計画通りに開発できる			ソフトウェア工学の技法を用いて部分的に開発できる		ソフトウェア工学の技法を用いて開発できない	
学科の到達目標項目との関係							
本科学習目標 1 本科学習目標 2 本科学習目標 3 本科学習目標 4 創造工学プログラム A1 創造工学プログラム B1専門(電気電子工学&情報工学)							
教育方法等							
概要	ソフトウェア工学では、ソフトウェア開発の各工程及びその管理の技法などの基礎学力と専門知識を養う。前期はソフトウェア工学の考え方にに基づき、ソフトウェア開発の各工程（要求分析、設計、開発、テスト、運用・保守）及びプロジェクト管理の技法を学ぶ。後期は実際に開発プロジェクトを実施し、前期で学んだ知識・技法を体験することで理解を深める。 この科目は企業でシステムエンジニアとして勤務していた教員が、その経験を活かし、ソフトウェア工学の技術について講義形式で授業を行うものである。						
授業の進め方・方法	授業内容の理解を深めるため、作品制作の演習を行います。 【関連科目】プログラミングI, プログラミングII, データベース						
注意点	【評価方法・評価基準】 前期中間試験、前期末試験および後期末試験を実施する。 前期末：前期中間試験（40％）と前期末試験（40％）の総合（80％）、課題（20％） 学年末：前期成績（50％）、後期末試験（25％）、演習（25％） 成績の評価基準として、60点以上を合格とする。						
テスト							
授業計画							
		週	授業内容		週ごとの到達目標		
前期	1stQ	1週	ソフトウェア工学とは		ソフトウェア工学の授業内容について理解できる		
		2週	ソフトウェア開発工程		ソフトウェア開発工程について理解できる		
		3週	モデリング(1)		モデリングについて理解できる		
		4週	モデリング(2)		モデリングについて理解できる		
		5週	モデリング(3)		モデリングについて理解できる		
		6週	要求分析・設計(1)		要求分析・設計方法について理解できる		
		7週	要求分析・設計(2)		要求分析・設計方法について理解できる		
		8週	実装(1)		実装方法について理解できる		
	2ndQ	9週	実装(2)		実装方法について理解できる		
		10週	検証・テスト(1)		検証・テストの方法について理解できる		
		11週	検証・テスト(2)		検証・テストの方法について理解できる		
		12週	検証・テスト(3)		検証・テストの方法について理解できる		
		13週	プロジェクト管理		プロジェクト管理方法について理解できる		
		14週	著作権・特許		ソフトウェアに関する法律（著作権・特許など）を理解できる		
		15週	前期復習				
		16週					
後期	3rdQ	1週	ソフトウェア設計・開発の演習についてのガイダンス		演習内容について理解できる		
		2週	オブジェクト指向プログラミング(1)		オブジェクト指向の考え方にに基づいて、クラスを作成することができる		
		3週	オブジェクト指向プログラミング(2)		オブジェクト指向の考え方にに基づいて、クラスを作成することができる		
		4週	オブジェクト指向プログラミングの開発演習(1)		オブジェクト指向でプログラミングできる		

		5週	オブジェクト指向プログラミングの開発演習(2)	オブジェクト指向でプログラミングできる
		6週	成果物の発表会(1)	開発したソフトウェアを発表できる
		7週	成果物の発表会(2)	開発したソフトウェアを発表できる
		8週	ソフトウェア設計演習(1)	実社会を考慮したソフトウェアの設計ができる
	4thQ	9週	ソフトウェア設計演習(2)	実社会を考慮したソフトウェアの設計ができる
		10週	オブジェクト指向設計(1)：カプセル化・継承・ポリモーフィズム	カプセル化・継承・ポリモーフィズムについて理解できる
		11週	オブジェクト指向設計(2)：クラス図・シーケンス図	クラス図・シーケンス図が作成できる
		12週	ソフトウェアテストの目的, ソフトウェアテストのプロセス, バグ管理図による品質管理	ソフトウェアテストの目的, ソフトウェアテストのプロセス, バグ管理図による品質管理を理解できる
		13週	単体テスト	ホワイトボックステスト(命令網羅, 分岐網羅), ブラックボックステスト(同値分割, 境界値分析)を理解できる
		14週	結合テスト	ボトムアップテスト, トップダウンテストを理解できる
		15週	後期復習	
		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
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	
			コンピュータシステム	プロジェクト管理の必要性について説明できる。	4	
				WBSやPERT図など、プロジェクト管理手法の少なくとも一つについて説明できる。	4	
				ER図やDFD、待ち行列モデルなど、ビジネスフロー分析手法の少なくとも一つについて説明できる。	4	
	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】	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。	4	
				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。	4	
				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。	4	

評価割合

	試験	演習・発表	課題	合計
総合評価割合	65	25	10	100
基礎的能力	0	0	0	0
専門的能力	65	25	10	100
分野横断的能力	0	0	0	0