

有明工業高等専門学校		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	5I008			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	創造工学科(情報システムコース)			対象学年	5		
開設期	前期			週時間数	前期:1		
教科書/教材	情報工学レクチャーシリーズ ソフトウェア工学入門；高橋直久, 丸山勝久／森北出版						
担当教員	森山 英明						
到達目標							
1. 大規模なソフトウェアの特徴と問題を理解できる 2. ソフトウェアに対する要求の獲得, 仕様化手法を理解できる 3. 要求仕様に応じた分析, 設計, 開発工程を理解できる							
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	
評価項目1	標準的な到達レベルに加え, ソフトウェア危機や小規模ソフトウェアとの開発工程の違いを説明できる.			大規模ソフトウェア開発の特徴を考慮した開発の工程を把握し, ソフトウェア工学の定義を理解している.		大規模ソフトウェア開発の特徴, 開発の工程, ソフトウェア工学の定義に関する知識が不足している.	
評価項目2	標準的な到達レベルに加え, 要求に応じた開発工数の見積もり, プロジェクトの管理, 品質の管理について説明できる.			大規模ソフトウェアに対する要求の獲得, 仕様化, 確認を理解し, 具体的な手法を説明できる.		大規模ソフトウェアに対する要求の獲得, 仕様化, 確認, およびこれらを実現する具体的な手法に関する知識が不足している.	
評価項目3	標準的な到達レベルに加え, システムのテストや保守・再利用に関して説明できる.			要求仕様に対する構造化分析とオブジェクト分析ができ, 各分析に応じた設計を説明できる.		構造化分析, オブジェクト分析, および設計手法に関する知識が不足している.	
学科の到達目標項目との関係							
学習・教育到達度目標 B-2							
教育方法等							
概要	システム開発者に要求されるソフトウェアは, 年々, 大規模化・複雑化・高信頼化している. 大規模で複雑なソフトウェアを開発する場合, ソフトウェアの要求分析や設計を行い, 作成された設計書に基づいて, 多数のプログラマが分担し連携を取りながら, ソフトウェアの開発作業を進めていく. 本授業では, 設計・開発作業を効率よく行うための技術について学習する.						
授業の進め方・方法	教科書を参照しつつ, 黒板を用いて授業を行う. また, 授業内容の理解を深めるために, 定期的に課題レポートを出題する. 課題レポートは, 文献やインターネットを利用して情報を収集し, 計画的に進めること.						
注意点	本授業で学習するソフトウェア開発に関する技術は, ソフトウェア開発者 (システムエンジニア) には必須の内容である. 本授業は, 今まで学習してきた情報工学系の授業科目全般 (プログラミング, システムプログラム, アルゴリズムなど) の知識を必要とする.						
授業計画							
		週	授業内容		週ごとの到達目標		
前期	1stQ	1週	大規模ソフトウェア開発の課題		大規模ソフトウェアの特徴を把握し, 各ステークホルダーによる視点の違いを理解できる. また, 課題に対して, 解決するための技術や考え方を説明できる.		
		2週	ソフトウェアの開発工程1		大規模ソフトウェアの開発工程と各工程の成果物を説明できる.		
		3週	ソフトウェアの開発工程2		ソフトウェアプロセスモデルの概要と具体的な手法を説明できる.		
		4週	プロジェクト管理		プロジェクト管理の概要と管理に役立つ視点, 技術, 管理手法を説明できる. また, 開発工数の見積もりを説明できる.		
		5週	要求分析		ソフトウェアに対する要求と各ステークホルダーの視点による要求の違いを説明できる. また, 要求を獲得するための具体的な手法と要求の取捨選択を説明できる.		
		6週	構造化分析1		構造化分析の概要を説明できる. また, データフロー図を用いてシステムを表現できる.		
		7週	構造化分析2		データフロー図の階層化を理解し, データフロー図を用いてシステムを実際に分析できる.		
		8週	オブジェクト指向分析1		オブジェクト指向分析の概要と基本概念を説明できる.		
	2ndQ	9週	オブジェクト指向分析2		UMLの概要と代表的な図の表記法を理解し, 適切な図を選択できる.		
		10週	アーキテクチャ設計		ソフトウェアアーキテクチャと設計の概要を説明できる. 代表的なアーキテクチャスタイルを把握し, 品質特性を考慮した適切なスタイルを選択できる.		
		11週	ユーザインタフェース設計		人間の認知機能を考慮したインタフェースの設計方法と考慮すべき事柄を理解できる.		
		12週	モジュール設計		モジュール設計の概要と利点を把握し, 良い/悪い構造を理解できる. また, 評価尺度であるモジュール強度と結合度を説明できる.		
		13週	プログラミング		プログラミングの概要, 言語, 開発環境, スタイル, アプローチの手法を説明できる.		
		14週	テストと検証(ISO)		ソフトウェアテストと具体的なテスト技法を説明でき, 検証を行うことができる. また, ISOによる品質管理を説明することができる.		
		15週	期末試験				

		16週	テスト返却と解説					
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		
専門的能力	分野別の専門工学	情報系分野	コンピュータシステム	ネットワークコンピューティングや組込みシステムなど、実用に供せられているコンピュータシステムの利用形態について説明できる。	4	前10		
				デュアルシステムやマルチプロセッサシステムなど、コンピュータシステムの信頼性や機能を向上させるための代表的なシステム構成について説明できる。	4	前10		
				集中処理システムについて、それぞれの特徴と代表的な例を説明できる。	4	前10		
				分散処理システムについて、特徴と代表的な例を説明できる。	4	前10		
				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。	4	前10		
				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。	4	前5		
				プロジェクト管理の必要性について説明できる。	4	前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	前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	前6		
分野横断的能力	汎用的技能	汎用的技能	汎用的技能	特性要因図、樹形図、ロジックツリーなど課題発見・現状分析のために効果的な図や表を用いることができる。	3	前5		
				グループワーク、ワークショップ等による課題解決への論理的・合理的な思考方法としてブレインストーミングやKJ法、PCM法等の発想法、計画立案手法など任意の方法を用いることができる。	3	前5		
				どのような過程で結論を導いたか思考の過程を他者に説明できる。	3	前5		
					工学的な課題を論理的・合理的な方法で明確化できる。	3	前5,前6	
					公衆の健康、安全、文化、社会、環境への影響などの多様な観点から課題解決のために配慮すべきことを認識している。	3	前14	
					要求に適合したシステム、構成要素、工程等の設計に取り組むことができる。	3	前2	
					課題や要求に対する設計解を提示するための一連のプロセス(課題認識・構想・設計・製作・評価など)を実践できる。	3	前2	
					提案する設計解が要求を満たすものであるか評価しなければならないことを把握している。	3	前14	
					経済的、環境的、社会的、倫理的、健康と安全、製造可能性、持続可能性等に配慮して解決策を提案できる。	3	前14	
	評価割合							
		試験	発表	相互評価	態度	ポートフォリオ	その他	合計
	総合評価割合	70	0	0	0	30	0	100
基礎的能力	45	0	0	0	15	0	60	
専門的能力	15	0	0	0	15	0	30	
分野横断的能力	10	0	0	0	0	0	10	