

大阪情報コンピュータ専門学校 授業シラバス (2024年度)

専門分野区分	プログラミング基礎	科目名	プログラミングⅡ				科目コード	T1112A3		
配当期	後期	授業実施形態	通常				単位数	6 単位		
担当教員名	黒原 恵美	履修グループ	1C(JT/KS/KW)				授業方法	演習		
実務経験の内容	流通系IT企業でシステムエンジニア、プログラマーとしてWindows版POSレジの開発に2年従事。GUIからGUIの変換時期であったため、それまでC言語で作成されていたPOSシステムをもとに、VisualBasicで商品情報、顧客情報などのマスタ登録画面を仕様書作成からプログラミング、単体テストなどを行っていました。実務経験をもとに、実際の現場をイメージしやすいよう具体的な話をまじえて授業をすすめていきたいと思ひます。									
学習一般目標	本校では、IPAが提唱し、企業で人材戦略のために広く利活用されているiCD(iコンピテンシ ディクショナリ)を参照し、「企業現場で遂行される実際のタスク(業務)」をベースとしてカリキュラムを構成している。実務において求められる機能や役割を「課される“仕事”」として定義したものを以下「タスク」という。科目「プログラミングⅠ～Ⅲ」では、アプリケーション開発においてプログラマーが遂行するタスクである「ソフトウェアコード作成・単体テストおよび結合テスト」が実施できることを目指す。そのために、本科目「プログラミングⅡ」では、ウォータフォールモデルを基本としてプログラミング及びテストの工程に必要なスキルを身に付ける。									
授業の概要および学習上の助言	モジュール論理設計の設計書に記述された処理手順を理解し、それに戻つたコーディングが行えるように演習を実施する。また設計書の作成もできるようにしていく。作成したプログラムについては、テスト技法にのっとりホワイトボックステストやブラックボックステストを実施し、単体テストと結合テストが行えるようにする。授業後半では、データアクセス技術を用いたデータベースプログラミングを学び、CRUD操作のコーディングを行う。最終課題の実務システム演習では、初級PGの「タスク」であるモジュール論理設計、プログラミング、単体テストの仕様作成、実施の総合演習を行う。本科目ではウォータフォールモデルを基本とする開発工程を理解し、1年前期の「プログラミングⅠ」で学んだプログラミングに必要な基礎知識やスキルを活用して、各開発工程の成果物に基づく実習課題に取り組んでほしい。									
教科書および参考書										
履修に必要な予備知識や技能	特になし									
使用機器	各自の個人所有パソコンを使用する									
使用ソフト	Visual Studio 2019、C# SQL Server									
学習到達目標	学部DP(番号表記)	学生が到達すべき行動目標								
	1/2/4	モジュール論理設計の設計書を理解・作成でき、設計書に基づいたコーディングができる								
	1/2/4	データベースへのアクセス技術を理解し、CRUD操作のコーディングができる								
	1/2/4	テスト仕様書に基づいて、ホワイトボックステストとブラックボックステストが実施できる								
	1/2/4	ツールを用いたバージョン管理ができる								
	5	専門知識や技能を修得するために自ら継続的に学習する								
達成度評価	評価方法	試験	小テスト	レポート	成果発表(口頭・実技)	作品	ポートフォリオ	その他	合計	
	学部DP	1.知識・理解			50					50
		2.思考・判断			10					10
		3.態度								
		4.技能・表現			10					10

	5.関心・意欲						30	30
	総合評価割合			70			30	100
評価の要点								
評価方法	評価の実施方法と注意点							
試験								
小テスト								
レポート	授業中に出題する課題に対する提出物の完成度を判断する。 ただし、提出物を自ら作成しなかった場合や期限までに提出しなかった場合は、原則として評価しない。							
成果発表(口頭・実技)								
作品								
ポートフォリオ								
その他	授業への出席、取り組みなどを含め総合的に判断する。							

### 授業明細表

授業回数	学習内容	授業の運営方法	学習課題(予習・復習)
第1回	設計書に基づいたプログラミング(復習) ・設計書の理解(擬似言語を用いた処理手順) ・基本文法のコーディングとデバッグの復習	講義・演習	
第2回	設計書に基づいたプログラミング(演習) ・設計書に基づいたコーディングとデバッグの演習	講義・演習	
第3回	テスト技法(単体テスト)を理解する ・テスト仕様書に基づいたテストコードの作成 ・ホワイトボックステストの実施	講義・演習	
第4回	テスト技法(単体テスト、結合テスト)を理解する ・ブラックボックステストの実施	講義・演習	
第5回	課題演習 ・第1回～4回で習得した内容の課題演習	講義・演習	
第6回	課題演習 ・第1回～4回で習得した内容の課題演習	講義・演習	
第7回	課題演習 ・第1回～4回で習得した内容の課題演習	講義・演習	
第8回	データアクセス技術を理解する(データプロバイダ) ・データプロバイダを用いたコーディングの理解 ・CRUD操作(Create(登録)、Read(参照)、Update(更新)、Delete(削除)機能)のコーディング練習	講義・演習	
第9回	データアクセス技術を理解する(データプロバイダ) ・CRUD操作のコーディング演習	講義・演習	
第10回	データアクセス技術を理解する(Entity Framework) ・Entity Frameworkを用いたコーディングの理解 ・CRUD操作のコーディング演習 ツールを用いた基本的なバージョン管理を理解する(Git) ・Gitの基本操作の理解	講義・演習	

第11回	実務システム演習 ・Entity Frameworkの実務的なコーディングを練習する ・Gitを用いたバージョン管理の実施	講義・演習	
第12回	実務システム演習 ・Entity Frameworkの実務的なコーディングを練習する	講義・演習	
第13回	実務システム演習 ・Entity Frameworkの実務的なコーディングを練習する	講義・演習	
第14回	課題解決型授業1 別途提示 (2回後までに自宅で課題に取り組むこと)	遠隔授業 実施時期:6期 (11月後半)	
第15回	課題解決型授業2 別途提示 (2回後までに自宅で課題に取り組むこと)	遠隔授業 実施時期:8期 (12月後半)	