

大阪情報コンピュータ専門学校 授業シラバス (2022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 I | | | | 科目コード | T1111A2 | | |
| 配当期 | 前期 | 授業実施形態 | 通常 | | | | 単位数 | 4 単位 | | |
| 担当教員名 | 谷本 成周 | 履修グループ | 1D(KS/KW) | | | | 授業方法 | 演習 | | |
| 実務経験の内容 | | | | | | | | | | |
| 学習一般目標 | <p>本校では、IPA(独立行政法人 情報処理推進機構)が提唱し、企業で人材戦略のために広く活用されているiCD(アイコンビテンシ 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。</p> <p>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 I」では、初心者が学習を始めるにあたり必要となる「プログラミングの基礎的知識・スキル」および「実習環境の基本機能」の修得を目指す。プログラミング言語はC#を使用する。</p> | | | | | | | | | |
| 授業の概要および学習上の助言 | <p>まず、視覚的なプログラミングの体験をとおしてプログラムの直感的理解を得るとともに、プログラム開発の全体像の概要を理解する。その後、様々なフォームアプリケーションやコンソールアプリケーションの作成・演習をとおしてC#の基本文法を修得するとともに、基本アルゴリズムやデータ構造を使用したコーディング・デバッグの方法を学ぶ。また、授業期間の後半では、プログラミングにおいて重要な概念となるメソッド(関数)の作成や使用について学習する。</p> <p>プログラムの開発環境を各自のノートPCに構築するので、授業時間内だけでなく、授業時間外でも自宅等で継続した学習を進めることが可能である。プログラミングスキルの向上には実習の量が重要な要素となるため、授業時間外でも自主的・自発的な学習を行うことが望ましい。</p> | | | | | | | | | |
| 教科書および参考書 | 「3ステップでしっかり学ぶ C#入門[改訂2版]」 技術評論社 | | | | | | | | | |
| 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# | | | | | | | | | |
| 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 | | | | | | | | |
| | 3/5 | プログラミングに興味関心を持ち、学習に取り組むことができる | | | | | | | | |
| | 1/2/4 | 実習環境の基本的な機能を利用し、フォームアプリやコンソールアプリを作成することができる | | | | | | | | |
| | 1/2/4 | 複数のメソッド(関数)を利用してプログラムを作成することができる | | | | | | | | |
| | 1/2/4 | 標準クラスを利用することや、簡単なクラスを作成することができる | | | | | | | | |
| | 1/2/4 | 作成したプログラムの不具合を発見し、デバッグなどを利用して修正することができる | | | | | | | | |
| 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 | | | 20 | | | | | 20 |
| | | 2.思考・判断 | | | 20 | | | | | 20 |
| | | 3.態度 | | | | | | | 15 | 15 |
| | | 4.技能・表現 | | | 30 | | | | | 30 |
| | | 5.関心・意欲 | | | | | | | 15 | 15 |
| | 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 | 授業中に出题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 | 実習環境を準備する ・学内Wi-Fi、メール等を設定する ・Visual Studioのインストール方法を学ぶ (次回までに自宅でインストールしてくる) プログラミング体験(Scratch) プログラム開発の全体像の概要を理解する | 講義・演習 | 外部サイト ・Scratch ・TechFUL |
| 第2回 | 初歩的なWindowsフォームを作成する | 講義・演習 | |
| 第3回 | C#ソースコードの構成、書き方などを理解する | 講義・演習 | |
| 第4回 | 演算と変数、およびデータ型を理解する 代入や四則演算など順次処理を作る | 講義・演習 | |
| 第5回 | if文による条件分岐を作る 制御構造として選択処理の構造を理解する | 講義・演習 | |
| 第6回 | switch文による多分岐を作る 制御構造として選択処理の構造を理解する | 講義・演習 | |
| 第7回 | for文による繰り返し処理を作る 制御構造として反復処理の構造を理解する | 講義・演習 | |
| 第8回 | do文、while文による繰り返し処理を作る 制御構造として反復処理の構造を理解する | 講義・演習 | |
| 第9回 | ソースコードを分割する(メソッド・関数) | 講義・演習 | |
| 第10回 | 配列を用いてプログラムを作成する 基本アルゴリズムをコーディングする ・合計と平均、最大値と最小値 ・探索(線形探索、二分探索) | 講義・演習 | |
| 第11回 | 基本アルゴリズムをコーディングする ・整列(選択法、隣接交換法) | 講義・演習 | |
| 第12回 | オブジェクト指向プログラミングの考え方の基礎を理解する ・クラス | 講義・演習 | |

| | | | |
|------|---------------------|-----------------|--|
| 第13回 | 簡単なクラスを作成する | 講義・演習 | |
| 第14回 | 課題解決型授業1 プログラム演習 | 遠隔授業 実施時期:2期 | |
| 第15回 | 課題解決型授業2 プログラム演習 | 遠隔授業 実施時期:4期 | |