

仙台高等専門学校		開講年度	令和03年度 (2021年度)		授業科目	プログラミング基礎	
科目基礎情報							
科目番号	0024		科目区分		専門 / 必修		
授業形態	授業		単位の種別と単位数		履修単位: 2		
開設学科	総合工学科Ⅰ類		対象学年		2		
開設期	通年		週時間数		2		
教科書/教材	なし（資料配付）						
担当教員	藤原 和彦,力武 克彰,張 暁勇						
到達目標							
1. コンピュータプログラミングの基礎を理解し、与えられた課題に対し、定められた手順に従い小規模なプログラムを実装、作成することができる。 2. プログラムの論理構造と処理の流れを理解し、言語に依らない処理手順を考えることができる。							
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		
コンピュータプログラミング	コンピュータプログラミングの基礎を正しく理解し、小規模なプログラムを作成、動作検証することができる。		コンピュータプログラミングの基礎を理解し、小規模なプログラムを作成することができる。		コンピュータプログラミングの基礎を理解し、小規模なプログラムを作成することができない。		
プログラミング設計	プログラミング言語に依らない処理手順を考えることができ、その手順を正しく説明できる。		プログラミング言語に依らない処理手順を考えることができる。		プログラミング言語に依らない処理手順を考えることができない。		
学科の到達目標項目との関係							
教育方法等							
概要	プログラミング言語としてはC言語を取り上げる。プログラミングは本科目と3年次開講のプログラミングとあわせ2年間で完成させることを目指す。本科目では、データ型、標準入出力、演算、制御構造、記憶クラス、標準ライブラリ関数、ユーザ関数等の基本文法について講義と実習を行い、小規模なプログラムを作成していく。						
授業の進め方・方法	C言語を対象に、プログラミングに必要な基本的な知識を学んだ後、理解を深めるために演習課題に取り組む。この手順を、一年間通して繰り返して知識・技術を習得していく。演習時にレポート用紙、5mm方眼紙、テンプレート、英和・和英辞典を利用することが多いので、常に持参してくること。 【事前学習】 講義回の前には、事前配布された授業資料から自作ノートを作成し、その内容を確認しておくこと。 【事後学習】 講義回の後に、授業で学んだことを振り返り知識として定着させる努力をすること。 課題がある際は、提出期限に十分間に合うよう取り組むこと。						
注意点	コードを入力するだけがプログラミングではないことに留意し、きちんと手順を踏んで課題に取り組むこと。また、教科書は副読本として位置づけており、次年度以降も利用するので留意すること。 指定された課題等のうち未提出のものがある場合は合格評価とならないので、計画的に取り組むこと。 【参考書等】 「C言語によるプログラミング 基礎編」システム計画研究所・内田智史（オーム社） 「C実践プログラミング」Steve Oualline（オライリー・ジャパン） 「C言語によるプログラミング 基礎編」内田智史（オーム社） 「C言語によるプログラミング スーパーリファレンス編」内田智史他（オーム社） 「苦しんで覚えるC言語」MMGames（秀和システム） 「Cクイックリファレンス」Peter Prinz, Tony Crawford（オライリー・ジャパン） など						
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	序論		コンピュータとプログラムについて基本的な概念を理解し、プログラミングに必要な操作と生成されるファイルについて理解できる。		
		2週	C言語の基礎1		C言語によるプログラミングの決まり事を理解し実践できる。 フローチャートを用いて処理手順を記述することができる。		
		3週	C言語の基礎2		C言語によるプログラミングの決まり事を理解し実践できる。 フローチャートを用いて処理手順を記述することができる。		
		4週	変数とデータ型		変数とデータ型の概念を説明できる。		
		5週	標準入出力		標準入力のための基本的な使い方を理解し、キーボードから数値や文字を入力できる。 標準出力のための基本的な使い方を理解し、任意の書式でデータを出力することができる。		
		6週	演算子1		代入や演算子の概念を理解し、式を記述できる。		
		7週	演算子2		代入や演算子の概念を理解し、式を記述できる。		
		8週	分岐処理1		制御構造の概念を理解し、条件分岐を記述できる。		
	2ndQ	9週	分岐処理2		制御構造の概念を理解し、条件分岐を記述できる。		
		10週	繰り返し処理1		制御構造の概念を理解し、反復処理を記述できる。		
		11週	繰り返し処理2		制御構造の概念を理解し、反復処理を記述できる。		

後期		12週	前期総合演習課題1	与えられた簡単な問題に対して、それを解決するためのプログラムを記述できる。 これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		13週	前期総合演習課題2	与えられた簡単な問題に対して、それを解決するためのプログラムを記述できる。 これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		14週	前期総合演習課題3	与えられた簡単な問題に対して、それを解決するためのプログラムを記述できる。 これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		15週	前期期末試験	前期期末試験の実施
		16週	前期期末試験の返却	前期期末試験の答案返却と解説
	3rdQ	1週	配列と文字列操作1	配列の概念を理解し、必要に応じてデータをまとめて処理することができる。 文字と文字列の違いを理解し、簡単な文字列操作ができる。
		2週	配列と文字列操作2	配列の概念を理解し、必要に応じてデータをまとめて処理することができる。 文字と文字列の違いを理解し、簡単な文字列操作ができる。
		3週	配列と文字列操作3	配列の概念を理解し、必要に応じてデータをまとめて処理することができる。 文字と文字列の違いを理解し、簡単な文字列操作ができる。
		4週	配列と文字列操作4	配列の概念を理解し、必要に応じてデータをまとめて処理することができる。 文字と文字列の違いを理解し、簡単な文字列操作ができる。
		5週	記憶クラス	それぞれの定義域と意味について理解し、適切に変数を宣言できる。
		6週	関数1	関数の概念を理解し、これらを含むプログラムを記述できる。 基本的な標準関数について、引数や戻り値の型を意識して正しく利用することができる。 必要な機能を自ら関数化し、利用することができる。
		7週	関数2	関数の概念を理解し、これらを含むプログラムを記述できる。 基本的な標準関数について、引数や戻り値の型を意識して正しく利用することができる。 必要な機能を自ら関数化し、利用することができる。
		8週	関数3	関数の概念を理解し、これらを含むプログラムを記述できる。 基本的な標準関数について、引数や戻り値の型を意識して正しく利用することができる。 必要な機能を自ら関数化し、利用することができる。
	4thQ	9週	関数4	関数の概念を理解し、これらを含むプログラムを記述できる。 基本的な標準関数について、引数や戻り値の型を意識して正しく利用することができる。 必要な機能を自ら関数化し、利用することができる。
		10週	総合演習課題1	これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		11週	総合演習課題2	これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		12週	総合演習課題3	これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		13週	総合演習課題4	これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		14週	総合演習課題5	これまでに学習した内容を応用し、課題の要求を満たすプログラムを作成することができる。
		15週	後期期末試験	後期期末試験の実施
		16週	後期期末試験の返却	後期期末試験の答案返却と解説

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	プログラミング	代入や演算子の概念を理解し、式を記述できる。	4	前7
				プロシージャ(または、関数、サブルーチンなど)の概念を理解し、これらを含むプログラムを記述できる。	4	後9
				変数の概念を説明できる。	4	前4
				データ型の概念を説明できる。	4	前4
				制御構造の概念を理解し、条件分岐を記述できる。	4	前9
				制御構造の概念を理解し、反復処理を記述できる。	4	前11
				与えられた問題に対して、それを解決するためのソースプログラムを記述できる。	3	前14
				ソフトウェア生成に必要なツールを使い、ソースプログラムをロードモジュールに変換して実行できる。	3	前14
				ソフトウェア開発に利用する標準的なツールの種類と機能を説明できる。	3	前3

				要求仕様に従って、標準的な手法により実行効率を考慮したプログラムを設計できる。	3	後14
				要求仕様に従って、いずれかの手法により動作するプログラムを設計することができる。	3	前14
				要求仕様に従って、いずれかの手法により動作するプログラムを実装することができる。	3	前14
				要求仕様に従って、標準的な手法により実行効率を考慮したプログラムを実装できる。	3	後14
			ソフトウェア	コンピュータ内部でデータを表現する方法(データ構造)にはバリエーションがあることを説明できる。	2	前4
			計算機工学	整数・小数をコンピュータのメモリ上でデジタル表現する方法を説明できる。	2	前4
				基数が異なる数の間で相互に変換できる。	1	前4
			情報数学・情報理論	コンピュータ上での数値の表現方法が誤差に関係することを説明できる。	3	前4
評価割合						
		試験	提出物	合計		
総合評価割合		30	70	100		
基礎的能力		20	35	55		
専門的能力		10	35	45		