香川高等専門学校	開講年度	令和02年度 (2020年度)	授業科目	情報処理基礎Ⅱ

科目基礎情報

科目番号	190204	科目区分	専門 / 必修
授業形態	講義，実習	単位の種別と単位数	履修単位: 2
開設学科	電気情報工学科（2018年度以前入学者）	対象学年	2
開設期	通年	週時間数	2
教科書/教材	黒瀬能聿，福田之介，「やさしく学べるC言語」，森北出版		
担当教員	重田 和弘		

到達目標

1．コンピュータとインターネットを利用するうえでのルールとマナーを理解し、対策が説明できる。
2．プログラム開発環境の操作方法を理解し，プログラムを作成できる。
3．入出力処理と演算を用いたプログラムを作成できる。
4．if-else文、for文を用いたプログラムを作成できる。分岐と繰り返し処理を含む簡単なフローチャートを書き，そのプログラムを作成することができる。
5．多次元配列を用いたプログラムを作成できる。
6．文字列の入出力処理を行うプログラム、文字列関数を使用したプログラムを作成できる。

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
コンピュータを利用するうえでのルールとマナー	コンピュータとインターネットを利用するうえでのルールとマナーを理解し、対策が説明できる。	コンピュータとインターネットを利用するうえでのルールとマナーの概略を理解し、対策が説明できる。	コンピュータとインターネットを利用するうえでのルールとマナーの概略を理解し、対策が説明できない。
プログラムの作成方法	プログラム開発環境の操作方法を理解し，プログラムを作成できる。	プログラム開発環境の操作方法を理解し，簡単なプログラムを作成できる。	プログラム開発環境の操作方法を理解し，簡単なプログラムを作成できない。
入出力と演算	入出力処理と演算を用いたプログラムを作成できる。	入出力処理と演算を用いた簡単なプログラムを作成できる。	入出力処理と演算を用いた簡単なプログラムを作成できない。
分岐と繰り返し	if-else文、for文を用いたプログラムを作成できる。分岐と繰り返し処理を含む簡単なフローチャートを書き，そのプログラムを作成することができる。	if-else文、for文を用いた簡単なプログラムを作成できる。分岐と繰り返し処理を含む簡単なフローチャートを書き，そのプログラムを作成することができる。	if-else文、for文を用いた簡単なプログラムを作成できない。分岐と繰り返し処理を含む簡単なフローチャートを書き，そのプログラムを作成することができない。
配列	多次元配列を用いたプログラムを作成できる。	簡単な1次元配列を用いたプログラムを作成できる。	簡単な1次元配列を用いたプログラムを作成できない。
文字列	文字列の入出力処理を行うプログラム、文字列関数を使用したプログラムを作成できる。	文字列の入出力処理を行う簡単なプログラムを作成できる。	文字列の入出力処理を行う簡単なプログラムを作成できない。

学科の到達目標項目との関係

学習・教育到達度目標 B-1

教育方法等

概要	C言語を用いたプログラミングの基礎を習得する。さらに、コンピュータとインターネットを利用するうえでのルールとマナーを理解する。
授業の進め方・方法	プログラミングの説明と実習を交互に行い授業を進める。実習では，4～6時間程度で自主的にプログラミングを行う機会を設け，作成したプログラムについて報告書を提出させる。
注意点	

授業計画

		週	授業内容	週ごとの到達目標
前期	1stQ	1週	1．ガイダンス 2．コンピュータを利用するうえでのルールとマナー	コンピュータとインターネットを利用するうえでのルールとマナーを理解し、対策が説明できる。
		2週	2．コンピュータを利用するうえでのルールとマナー 3．プログラムの作成方法	コンピュータとインターネットを利用するうえでのルールとマナーを理解し、対策が説明できる。
		3週	3．プログラムの作成方法	プログラム開発環境の操作方法を理解し，プログラムを作成できる。
		4週	3．プログラムの作成方法	プログラム開発環境の操作方法を理解し，プログラムを作成できる
		5週	4．入出力と演算	入出力処理と演算を用いたプログラムを作成できる。
		6週	4．入出力と演算	入出力処理と演算を用いたログラムを作成できる。
		7週	4．入出力と演算	入出力処理と演算を用いたプログラムを作成できる。
		8週	前期中間試験	
	2ndQ	9週	5．分岐と繰り返し(if-else文)	if-else文を用いたプログラムを作成できる。
		10週	5．分岐と繰り返し(if-else文)	if-else文を用いたプログラムを作成できる。
		11週	5．分岐と繰り返し(if-else文)	if-else文を用いたプログラムを作成できる。
		12週	5．分岐と繰り返し(for文)	for文を用いたプログラムを作成できる。
		13週	5．分岐と繰り返し(for文)	for文を用いたプログラムを作成できる。
		14週	5．分岐と繰り返し(for文)	for文を用いたプログラムを作成できる。
		15週	5．分岐と繰り返し まとめ	分岐と繰り返し処理を含む簡単なフローチャートを書き，そのプログラムを作成することができる。
		16週	前期期末試験	
後期	3rdQ	1週	5．分岐と繰り返し(While文、do-while文)	while 文，do-while文を用いたプログラムを作成できる。
		2週	5．分岐と繰り返し(switch文)	switch文を用いたプログラムを作成できる。

		3週	６．総合演習Ⅰ	分岐と繰り返し処理を用いたプログラムが作成できる。
		4週	６．総合演習Ⅰ	分岐と繰り返し処理を用いたプログラムが作成できる。
		5週	７．配列(一次元配列)	一次元配列を用いたプログラムを作成できる。
		6週	７．配列(一次元配列)	一次元配列を用いたプログラムを作成できる。
		7週	７．配列(多次元配列)	多次元配列（3次元以下）を用いたプログラムを作成できる。
		8週	後期中間試験	
	4thQ	9週	８．文字列	文字列の入出力処理を行うプログラムを作成できる。
		10週	８．文字列	代表的な文字列関数を使用して，文字列処理を行うプログラムを作成できる。
		11週	９．総合演習Ⅱ	分岐，繰り返し処理，配列・文字列処理を用いたプログラムが作成できる。
		12週	９．総合演習Ⅱ	分岐，繰り返し処理，配列・文字列処理を用いたプログラムが作成できる。
		13週	９．総合演習Ⅱ	分岐，繰り返し処理，配列・文字列処理を用いたプログラムが作成できる。
		14週	１０．まとめ、復習	分岐，繰り返し処理，配列・文字列処理を用いたプログラムが作成できる。
		15週	１０．まとめ、復習	分岐，繰り返し処理，配列・文字列処理を用いたプログラムが作成できる。
		16週	後期期末試験	

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
基礎的能力	工学基礎	情報リテラシー	情報リテラシー	情報を適切に収集・処理・発信するための基礎的な知識を活用できる。	3	
				論理演算と進数変換の仕組みを用いて基本的な演算ができる。	3	
				コンピュータのハードウェアに関する基礎的な知識を活用できる。	3	
				情報伝達システムやインターネットの基本的な仕組みを把握している。	3	
				同一の問題に対し、それを解決できる複数のアルゴリズムが存在しうることを知っている。	3	
				与えられた基本的な問題を解くための適切なアルゴリズムを構築することができる。	3	
				任意のプログラミング言語を用いて、構築したアルゴリズムを実装できる。	3	
				情報セキュリティの必要性および守るべき情報を認識している。	3	
				個人情報とプライバシー保護の考え方についての基本的な配慮ができる。	3	
				インターネット(SNSを含む)やコンピュータの利用における様々な脅威を認識している	3	
				インターネット(SNSを含む)やコンピュータの利用における様々な脅威に対して実践すべき対策を説明できる。	3	
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。	3	
				変数の概念を説明できる。	1	
				データ型の概念を説明できる。	3	
				制御構造の概念を理解し、条件分岐を記述できる。	3	
				制御構造の概念を理解し、反復処理を記述できる。	3	
				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。	2	
				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。	2	
				与えられたソースプログラムを解析し、プログラムの動作を予測することができる。	2	
				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	
	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】	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、標準的な開発ツールや開発環境を利用して記述できる。	3	
				ソフトウェア生成に利用される標準的なツールや環境を使い、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。	1	
				ソフトウェア開発の現場において標準的とされるツールを使い、生成したロードモジュールの動作を確認できる。	1	
				フローチャートなどを用いて、作成するプログラムの設計図を作成することができる。	3	
				問題を解決するために、与えられたアルゴリズムを用いてソースプログラムを記述し、得られた実行結果を確認できる。	2	
				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。	2	

評価割合

	試験	レポート	合計
総合評価割合	70	30	100

コンピュータを利用するうえでのルールとマナー	5	0	5
プログラミング	65	30	95