

沼津工業高等専門学校		開講年度	令和04年度 (2022年度)		授業科目	プログラミング基礎演習	
科目基礎情報							
科目番号	2022-355			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: 1		
開設学科	電子制御工学科			対象学年	3		
開設期	前期			週時間数	2		
教科書/教材	やさしいC（第5版）、高橋麻奈著、SB Creative						
担当教員	香川 真人						
到達目標							
1. C言語を学ぶ理由を説明できる。 2. C言語の基本的な文法を理解することができる。 3. 学習した文法・アルゴリズムに基づくプログラムを作成できる。 4. 作成したプログラムの動作を確認し、デバッグを行うことができる。							
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		
1. C言語を学ぶ理由と統合開発環境について説明できる。	C言語を学ぶ理由と統合開発環境について明確に説明できる。		C言語を学ぶ理由と統合開発環境について説明できる。		C言語を学ぶ理由と統合開発環境について説明できない。		
2. C言語の基本的な文法を理解することができる。	C言語の基本的な文法を正確に理解することができる。		C言語の基本的な文法を理解することができる。		C言語の基本的な文法を理解することができない。		
3. 学習した文法・アルゴリズムに基づくプログラムを作成できる。	学習した文法・アルゴリズムに基づくプログラムを正しく作成できる。		学習した文法・アルゴリズムに基づくプログラムを作成できる。		学習した文法・アルゴリズムに基づくプログラムを作成できない。		
4. 作成したプログラムの動作を確認し、デバッグを行うことができる。	作成したプログラムの動作を確認し、漏れなくデバッグを行うことができる。		作成したプログラムの動作を確認し、デバッグを行うことができる。		作成したプログラムの動作を確認し、デバッグを行うことができない。		
5. アルゴリズムとフローチャートを作成できる。	アルゴリズムとフローチャートを正確に作成できる。		アルゴリズムとフローチャートを作成できる。		アルゴリズムとフローチャートを作成できない。		
学科の到達目標項目との関係							
【本校学習・教育目標（本科のみ）】 3							
教育方法等							
概要	本講義では、C言語を学ぶ理由を理解し、C言語の基礎的な文法全般を理解しそれに基づくプログラムを作成（コーディング）できるようにする。また、コーディングしたプログラムの動作確認を行い、不具合を見つけ修正するデバッグ作業を行えるようにすることを目標とする。授業では演習と課題レポートを実施し、知識の定着とプログラミングとデバッグ技術向上を図る。						
授業の進め方・方法	授業は、プレゼンテーション形式の座学の時間と演習を実施する時間で構成される。座学では、オンライン上に資料を置き、授業の要点をまとめ、理解を促す。演習では、時間内に課題のプログラムを完成させる。時間が足りない場合は宿題とする。						
注意点	1.授業内容は教科書に準拠して行うため、事前に予習をしておく。						
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	ガイダンス		到達目標や学習方法、評価方法について理解できる。C言語を学ぶ理由を理解できる。		
		2週	統合開発環境とプログラムの作成・コンパイル・実行方法		統合開発環境について理解する。プログラムの作成・コンパイル・実行方法について		
		3週	コードの基本、画面出力、文字と数値		ソースコード、画面出力、文字数値の扱いについて説明できる。		
		4週	変数とキーボード入力		変数およびキーボード入力の扱いを理解し、対応するプログラムを作成できる。		
		5週	式と演算子		式と演算子の扱いを理解し、対応するプログラムを作成できる。		
		6週	条件分岐（if文、switch文）		条件分岐処理を理解し、対応するプログラムを作成できる。		
		7週	繰り返し（for文 while文）		繰り返し処理を理解し、対応するプログラムを作成できる。		
		8週	配列		配列の扱いを理解し、対応するプログラムを作成できる。		
	2ndQ	9週	関数		関数の扱いを理解し、対応するプログラムを作成できる。		
		10週	ポインタ		ポインタの扱いを理解し、対応するプログラムを作成できる。		
		11週	配列・ポインタの応用		配列とポインタの関係を理解し、対応するプログラムを作成できる。		
		12週	構造体、列挙型		構造体、列挙型の扱いを理解し、対応するプログラムを作成できる。		
		13週	ファイル入出力		ファイルへのデータの入出力の扱いを理解し、対応するプログラムを作成できる。		
		14週	アルゴリズム		ソートのアルゴリズムを理解し、対応するプログラムを作成できる。		

		15週	フローチャート		プログラムのアルゴリズムをフローチャートとして表現できる。	
		16週				
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	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。	2	前14
				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。	2	前14
				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。	2	前14
専門的能力	分野別の専門工学	機械系分野	情報処理	プログラムを実行するための手順を理解し、操作できる。	3	前1
				定数と変数を説明できる。	3	前4
				整数型、実数型、文字型などのデータ型を説明できる。	3	前4
				演算子の種類と優先順位を理解し、適用できる。	3	前5
				算術演算および比較演算のプログラムを作成できる。	3	前5
				データを入力し、結果を出力するプログラムを作成できる。	3	前4,前13
				条件判断プログラムを作成できる。	3	前6,前7
				繰り返し処理プログラムを作成できる。	3	前6,前7
				一次元配列を使ったプログラムを作成できる。	3	前8
		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5
				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	前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13
				制御構造の概念を理解し、条件分岐を記述できる。	4	前6
				制御構造の概念を理解し、反復処理を記述できる。	4	前7
				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。	4	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。	4	前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。	4	前15
				主要な言語処理プロセッサの種類と特徴を説明できる。	3	前1
				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2
				プログラミング言語は計算モデルによって分類されることを説明できる。	3	前1
				主要な計算モデルを説明できる。	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設計できる。	4	前15
				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。	4	前15
				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。	4	前15
				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。	4	前15
			ソフトウェア	アルゴリズムの概念を説明できる。	3	前14
				与えられたアルゴリズムが問題を解決していく過程を説明できる。	3	前14
				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説明できる。	3	前14
				整列、探索など、基本的なアルゴリズムについて説明できる。	3	前14
				時間計算量によってアルゴリズムを比較・評価できることを説明できる。	3	前14
				領域計算量などによってアルゴリズムを比較・評価できることを説明できる。	3	前14
				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。	3	前11
				同一の問題に対し、選択したデータ構造によってアルゴリズムが変化しうることを説明できる。	3	前11
				リスト構造、スタック、キュー、木構造などの基本的なデータ構造の概念と操作を説明できる。	3	前11
				リスト構造、スタック、キュー、木構造などの基本的なデータ構造を実装することができる。	3	前11
				ソフトウェアを中心としたシステム開発のプロセスを説明できる。	2	前15
				ソースプログラムを解析することにより、計算量等のさまざまな観点から評価できる。	2	前15
				同じ問題を解決する複数のプログラムを計算量等の観点から比較できる。	2	前15

			情報数学・ 情報理論	離散数学に関する知識をアルゴリズムの設計、解析に利用することができる。	3	前15
				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15
				コンピュータ上で数値計算を行う際に発生する誤差の影響を説明できる。	3	前15
				コンピュータ向けの主要な数値計算アルゴリズムの概要や特徴を説明できる。	4	前14
	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	前2
				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。	3	前2
				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。	4	前2
				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	前15
				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。	3	前15
				要求仕様に従って標準的な手法によりプログラムを設計し、適切な実行結果を得ることができる。	2	前15

評価割合

	試験	課題	合計
総合評価割合	50	50	100
基礎的能力	0	0	0
専門的能力	50	50	100
分野横断的能力	0	0	0