

石川工業高等専門学校		開講年度	令和05年度 (2023年度)	授業科目	プログラミング基礎 I I
科目基礎情報					
科目番号	20325	科目区分	専門 / 必修		
授業形態	講義	単位の種別と単位数	履修単位: 2		
開設学科	電子情報工学科	対象学年	2		
開設期	通年	週時間数	2		
教科書/教材	高橋麻奈「やさしいC (第5版)」SB Creative/LMSやMicrosoft Teams等で適宜配布する(参考書籍はシラバス内を参照のこと)				
担当教員	三吉 建尊				
到達目標					
<ol style="list-style-type: none"> 1. C言語の基本文法について理解し, 説明できる. 2. 変数の記法について理解し, 説明できる. 3. データ型について理解し, 説明できる. 4. C言語の制御文の動作を理解し, 各種制御文を用いたプログラムを作成できる. 5. 関数の記法と動作について理解し, 関数を用いたプログラムを作成できる. 6. 変数のスコープについて理解し, 適切なプログラムを作成できる. 7. ポインタ変数の記法と動作を理解し, 説明およびプログラムを作成できる. 8. 配列について理解し, 説明できる. 9. 配列とポインタの関係について理解し, 説明できる. 10. ポインタを用いたプログラムを作成できる. 11. 文字と文字列を理解し, それを用いたプログラムを作成できる. 12. 構造体を理解し, 説明できる. 13. 簡単なファイル入出力処理の記法を理解し, 説明およびプログラムを作成できる. 14. メモリの動的確保の記法と動作を理解し, 適切なプログラムを作成できる. 					
ループリック					
	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		
到達目標 項目 1 ~ 4	C言語の基本文法について理解し, プログラムを作成できる.	C言語の基本文法について理解し, 簡単なプログラムを作成できる.	C言語の基本文法について理解し, プログラムを作成できない.		
到達目標 項目 5, 6	関数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.	関数などの概念を理解し, これらを含む簡単なプログラムを記述できる.	関数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ない.		
到達目標 項目 7 ~ 14	配列, ポインタ, 構造体の概念を理解し, これらを含むプログラムを記述できる.	配列, ポインタ, 構造体の概念を理解し, これらを含む簡単なプログラムを記述できる.	配列, ポインタ, 構造体の概念を理解し, これらを含むプログラムを記述できない.		
学科の到達目標項目との関係					
本科学習目標 1 本科学習目標 2					
教育方法等					
概要	<p>この授業では, C言語によるプログラミングの基礎とコンピュータの仕組みに詳しくなることを目指す。そのために, C言語の基本構文, ロジックの組み立て, 関数, ポインタ, 構造体, ファイル入出力などの他のプログラミング言語の修得にも応用可能な知識を一通り習得する。</p> <p>本講義では企業においてシステムエンジニア(SE)としてシステム開発を担当していた教員がその経験を活かして講義を行うものである。</p>				
授業の進め方・方法	<p>各自のノートパソコンを利用し, 講義と演習を同時に実施する。 【事前事後学習など】講義内容の復習および理解を深めるために, 適宜, 課題を課す (長期休業中の課題を含む)。 【関連科目】プログラミング基礎 I, 電子情報工学基礎 I・II, アルゴリズムとデータ構造, コンピュータアーキテクチャ 【MCC 対応】V-D-1プログラミング, VI-D情報系分野 (実験・実習能力), 情報教育対応科目</p>				
注意点	<p>・ 講義には充電が完了したノートパソコンを持参すること。 ・ 授業中に適宜演習を行い学習到達の確認を行う。 ・ 演習課題の配点が高いため, しっかりと取り組み, 提出期限を厳守すること。</p> <p>プログラミング上達のコツは, 「多くのプログラムを構築すること」「エラーやトラブルを自己解決すること」の2つです。 エラーやトラブルが発生することは悪いことではありません。 たくさんの知識を身に付けてどんどん新しいことに挑戦し, 色々なプログラムを作りましょう!</p> <p>【評価方法・評価基準】成績の評価基準として50点以上を合格とする。 前期・後期ともに中間試験・期末試験を実施する。 前期成績: 中間試験 25%, 期末試験 25%, 前期課題 50% 後期成績: 中間試験 25%, 期末試験 25%, 後期課題 50% 学年末: 前期成績 50%, 後期成績 50%</p>				
テスト					
授業の属性・履修上の区分					
<input type="checkbox"/> アクティブラーニング		<input checked="" type="checkbox"/> ICT 利用		<input checked="" type="checkbox"/> 遠隔授業対応	
<input checked="" type="checkbox"/> 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	C言語の基本文法について理解し, 説明できる	
		2週	C言語の基本 (1)	C言語の基本文法について理解し, 説明できる	
		3週	C言語の基本 (2)	C言語の基本文法について理解し, 説明できる	
		4週	変数と型	変数の記法やデータ型について理解し, 説明できる	
		5週	式と演算子	変数の記法やデータ型について理解し, 説明できる	
		6週	条件分岐	C言語の制御文の動作を理解し, 各種制御文を用いたプログラムを作成できる	

後期	2ndQ	7週	論理演算子	C言語の制御文の動作を理解し、各種制御文を用いたプログラムを作成できる	
		8週	関数	関数の記法と動作について理解し、関数を用いたプログラムを作成できる	
		9週	変数とスコープ	変数のスコープについて理解し、適切なプログラムを作成できる	
		10週	ポインタ (1)	ポインタ変数の記法と動作を理解し、説明およびプログラムを作成できる	
		11週	ポインタ (2)	ポインタ変数の記法と動作を理解し、説明およびプログラムを作成できる	
		12週	配列とポインタ (1)	配列について理解し、説明できる	
		13週	配列とポインタ (2)	配列とポインタの関係について理解し、説明できる	
		14週	計算機演習	ポインタを用いたプログラムを作成できる	
	15週	前期復習			
	16週				
	後期	3rdQ	1週	構造体 (1)	構造体を理解し、説明できる
			2週	構造体 (2)	構造体を理解し、説明できる
			3週	構造体 (3)	構造体を理解し、説明できる
			4週	メモリの動的確保 (1)	文字と文字列を理解し、それを用いたプログラムを作成できる。
			5週	メモリの動的確保 (2)	メモリの動的確保の記法と動作を理解し、適切なプログラムを作成できる
			6週	計算機演習	メモリの動的確保の記法と動作を理解し、適切なプログラムを作成できる
7週			計算機演習	メモリの動的確保の記法と動作を理解し、適切なプログラムを作成できる	
8週			ファイル入出力 (1)	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
4thQ		9週	ファイル入出力 (2)	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
		10週	ファイル入出力 (3)	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
		11週	計算機演習	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
		12週	計算機演習	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
		13週	計算機演習	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
		14週	計算機演習	簡単なファイル入出力処理の記法を理解し、説明およびプログラムを作成できる	
		15週	後期復習		
		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	
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1,前2,前3,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8,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4
				制御構造の概念を理解し、条件分岐を記述できる。	4	前6,前7
				制御構造の概念を理解し、反復処理を記述できる。	4	前6
				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3
	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。	4	前14
				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。	4	前14
				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14
				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。	4	後11,後12
				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。	4	後13,後14

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