

豊田工業高等専門学校		開講年度	令和04年度 (2022年度)	授業科目	プログラミングⅡB	
科目基礎情報						
科目番号	32211		科目区分	専門 / 選択		
授業形態	講義		単位の種別と単位数	履修単位: 2		
開設学科	情報工学科		対象学年	2		
開設期	後期		週時間数	4		
教科書/教材	特に指定しない/教材用プリント配布					
担当教員	稲垣 宏					
到達目標						
<p>(ア)制御構造の概念を理解し、C言語で利用できる様々な制御構造を利用することができる。</p> <p>(イ)代入や演算子の概念を理解し、式を記述できる。初歩的な演算の仕組みを理解できる。C言語特有の演算子を使うことができる。</p> <p>(ウ)変数とデータ型の概念を説明できる。構造体を自分で定義し、それを操作することができる。</p> <p>(エ)関数の概念を理解し、それを利用できる。文字列処理関数を利用することができる。数値計算の基礎が理解できる。</p> <p>(オ)基本的なファイル入出力処理を実現できる。</p> <p>(カ)与えられた簡単な問題を解決するためのソースプログラムを机上で記述できる。さらに、標準的な開発環境上で記述できる。</p> <p>(キ)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実行形式に変換できる。さらにそれを実行し、動作を確認できる。</p>						
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	
評価項目(ア)	変数・データ型・代入・演算子・制御構造・関数の概念を説明でき、応用課題レベルのプログラムを記述することができる。		変数・データ型・代入・演算子・制御構造・関数の概念を説明でき、例題レベルのプログラムを記述することができる。		変数・データ型・代入・演算子・制御構造・関数の概念を説明できない。例題レベルのプログラムを記述できない。	
評価項目(イ)	与えられた応用課題レベルの問題に対して、それを解決するためのソースコードを記述することができる。とともに、プログラムの動作を理論に基づいて予測できる。		与えられた基本レベルの問題に対して、それを解決するためのソースコードを記述することができる。とともに、プログラムの動作を予測できる。		与えられた基本レベルの問題に対して、それを解決するためのソースコードを記述できない。また、プログラムの動作を予測することができない。	
評価項目(ウ)	ソフトウェア開発環境の提供する機能を使いこなし、実行形式のファイルを効率よく生成し、実行できる。		ソフトウェア開発環境を利用して、実行形式のファイルを生成し、実行できる。		ソフトウェア開発環境を利用して、実行形式のファイルを生成し、実行することができない。	
学科の到達目標項目との関係						
本校教育目標 ① ものづくり能力						
教育方法等						
概要	「プログラミングⅡA」に引き続き、より実用的なプログラミング技術を習得するために、C言語を利用したプログラミング教育を行なう。ここでは、「プログラミングⅡA」で紹介できなかった制御構造や演算子を取り上げた後、構造体やファイル操作等までをカバーし、これでC言語の文法事項は一通りマスターしたことになる。					
授業の進め方・方法	プログラミング実習用機器の台数が限られているため、他学科他学年の履修を制限する場合がある。					
注意点	プログラミング実習用機器の台数が限られているため、他学科他学年の履修を制限する場合がある。					
選択必修の種別・旧カリ科目名						
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 type="checkbox"/> ICT 利用		<input 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						
必履修						
授業計画						
	週	授業内容		週ごとの到達目標		
後期	3rdQ	1週	シラバスを用いたガイダンス 関数の作成：ユーザ関数の作り方、値渡し、参照渡し(1)		値渡しによる関数をソースコードとして記述できる。 参照渡しによる関数をソースコードとして記述できる。	
		2週	関数の作成：参照渡し(2)、一次元配列を渡す		参照渡しを使って複数の解を同時に求める関数を記述できる。 一次元配列を渡す関数をソースコードとして記述できる。	
		3週	関数の作成：二次元配列を渡す、ポインタ配列を渡す		二次元配列を渡す関数をソースコードとして記述できる。 ポインタ配列を渡す関数をソースコードとして記述できる。	
		4週	関数の作成：関数プロトタイプ その他の制御構造：do while 文		関数プロトタイプ概念を説明できるとともに、ソースコードとして記述できる。 do while文を使って、後判定の繰り返し処理をソースコードとして記述できる	
		5週	その他の制御構造：switch case 文、else if 文の構造と使い方		各種の制御構造を理解し、記述することができる。	
		6週	C特有の演算子：ビット演算子、条件演算子、sizeof演算子		ビット演算子を使ったソースコードを記述できる。条件演算子を使ったソースコードを記述できる。	
		7週	構造体：構造体の概念と定義方法、構造体配列、ポインタ参照の使い方		構造体の定義と変数宣言について、その概念を説明できるとともに、ソースコードとして記述することができる。また、定義した構造体を使ったソースコードを記述することができる。ポインタを使って、構造体メンバにアクセスすることができる。	
		8週	データ型と記憶クラス：列挙型、符号なし整数型、通用範囲、静的変数		列挙型を使ったソースコードを記述できる。 通用範囲の概念、記憶クラスの概念を理解し、外部変数および静的変数を使ったソースコードを記述できる。	

4thQ	9週	プリプロセッサ：簡単なマクロおよび引数付きマクロの作り方 標準ライブラリ関数：数学関数、乱数関数の使い方	プリプロセッサの概念を理解し、#define文を使ったソースコードを記述することができる。 数学関数および乱数関数を使ったソースコードを記述することができる。
	10週	標準ライブラリ関数：文字列処理関数の使い方	文字列処理関数を使ったソースコードを記述することができる。
	11週	標準ライブラリ関数：文字処理関数の使い方 標準ライブラリ関数：ソート関数の使い方、サーチ関数の使い方	文字処理関連の関数を使ったソースコードを記述することができる。 ソート関数とサーチ関数を使ったソースコードを記述することができる。
	12週	ファイル操作：ファイルとは、1文字単位のファイル入出力、1行単位のファイル入出力 コマンドライン引数	ファイル入出力の概念を説明できるとともに、ファイル入出力を実現するソースコードを記述できる。 コマンドライン引数の概念を説明できるとともに、それを実現するソースコードを記述できる。
	13週	ファイル操作：書式付きファイル入出力、バイナリファイルの入出力	書式ファイル入出力を行うソースコードを記述できる。 バイナリファイルの概念を説明できるとともに、バイナリファイル入出力を行うソースコードの記述できる。
	14週	関数の再帰呼び出し 動的配列とメモリ操作	再帰呼び出しの概念を説明することができる。また、モンテカルロ法を実装したプログラムを作成することができる。 動的配列の概念を説明することができる。また、メモリ操作関連の標準ライブラリ関数を使ったソースコードを記述できる。
	15週	乱数の応用：モンテカルロ法 プリプロセッサの応用：条件コンパイル 後期の総まとめ	モンテカルロ法を実装したソースコードを記述できる。 プリプロセッサを使って条件コンパイルを実現できる。 後期の内容を振り返って説明することができる。
	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	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ていることを知っている。	4	後10,後11,後14,後15	
			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。	4	後10,後11,後14,後15	
			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。	4	後10,後11,後14,後15	
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	後6,後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1,後2,後3,後4,後9,後10,後11,後12,後13,後14
				変数の概念を説明できる。	4	後2,後3,後6,後7,後8,後12,後13
				データ型の概念を説明できる。	4	後2,後3,後4,後6,後7,後8,後12,後13
				制御構造の概念を理解し、条件分岐を記述できる。	4	後5
				制御構造の概念を理解し、反復処理を記述できる。	4	後4
				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	後1,後2,後3,後4,後5,後6,後7,後8,後9,後10,後11,後12,後13,後14,後15
				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	後1,後2,後3,後4,後5,後6,後7,後8,後9,後10,後11,後12,後13,後14,後15
				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	後1,後2,後3,後4,後5,後6,後7,後8,後9,後10,後11,後12,後13,後14,後15

	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	後1,後2,後3,後4,後5,後6,後7,後8,後9,後10,後11,後12,後13,後14,後15
				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	後1,後2,後3,後4,後5,後6,後7,後8,後9,後10,後11,後12,後13,後14,後15
				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	後1,後2,後3,後4,後5,後6,後7,後8,後9,後10,後11,後12,後13,後14,後15
				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1,後2,後3,後4,後5,後6,後7,後8,後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。	4	後1,後2,後3,後4,後5,後6,後8,後9,後10,後11,後12,後13,後14,後15

評価割合

	定期試験	課題	合計
総合評価割合	65	35	100
基礎的能力	65	35	100