

一関工業高等専門学校		開講年度	令和02年度 (2020年度)	授業科目	応用プログラミング
科目基礎情報					
科目番号	0009		科目区分	専門 / 必修	
授業形態	演習		単位の種別と単位数	履修単位: 2	
開設学科	未来創造工学科 (情報・ソフトウェア系)		対象学年	3	
開設期	通年		週時間数	2	
教科書/教材	教科書 新訂 新C言語入門-シニア編-, 林晴比古, SBクリエイティブ, 2700(+税)円				
担当教員	豊田 計時, 小林 健一				
到達目標					
<p>1. 様々なプログラミング言語の基礎ともなっているC言語を実用的なレベルで使えるようになる。</p> <p>2. 基本的なアルゴリズムを理解することで、応用的なプログラムを自ら作成できるようにする。</p> <p>3. アルゴリズムの一般性を理解することで、C言語以外のプログラミング言語に応用することができる。</p> <p>【キーワード】プログラミング、C言語、アルゴリズム、データ構造</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
関数とライブラリ	与えられた問題に対し、適切な入出力を設定し自作関数を作成できる。また、標準ライブラリや外部ライブラリの利用法を自分で調査し、それを利用した応用的なプログラムを作成出来る。		与えられた問題に対し、適切な入出力を設定し自作関数を作成できる。また、標準ライブラリや外部ライブラリを利用してプログラムを作成出来る。		与えられた問題に対し、適切な入出力を設定し自作関数を作成できる。また、標準ライブラリや外部ライブラリを利用してプログラムを作成することが出来ない。
ポインタの応用	ポインタを使ったメモリへのアクセス、メモリの動的利用などポインタを用いた応用的なプログラムを自ら作成することができる。		ポインタを使ったメモリへのアクセス、メモリの動的利用などポインタを用いたプログラムを作成することができる。		ポインタを使ったメモリへのアクセス、メモリの動的利用などポインタを用いたプログラムを作成することが出来ない。
構造体・基本的なデータ構造	構造体やそれを応用したリストなどの基本的なデータ構造を、与えられた問題に対し適切に設計でき、与えられた問題を解決する応用的なプログラムを作成出来る。		構造体やそれを応用したリストなどの基本的なデータ構造を適切に設計できる。		構造体やそれを応用したリストなどの基本的なデータ構造を適切に設計出来ない。
C言語の仕組み	コンパイラ・プリプロセッサなどC言語周辺の知識を利用したプログラムを自ら設計し作成出来る。		コンパイラ・プリプロセッサなどC言語周辺の知識が理解でき、その知識を利用したプログラムを作成できる。		コンパイラ・プリプロセッサなどC言語周辺の知識が理解できない。
構造化プログラミング	自作関数や再起処理を利用し、与えられた問題に対する適切なプログラムを自ら設計し作成することができる。		自作関数や再帰処理を利用したプログラムを作成出来る。		自作関数や再帰処理について理解できない。
微分方程式への応用	減衰振動を表現したプログラムを自ら設計し作成できる。		減衰振動を表現したプログラムを作成できる。		減衰振動について理解できない。
プログラム開発過程、プログラム作成演習	与えられた問題に関する適切な問題解決法を導き、プログラム開発過程に基づいたグループ開発により、問題へのソリューションとなるプログラムを作成できる。また、制作物に関する適切なプレゼンテーションができる。		与えられた問題に関する適切な問題解決法を導き、グループ開発により、問題へのソリューションとなるプログラムを作成できる。また制作物に関する適切なプレゼンテーションができる。		与えられた問題に関する適切な問題解決法を導くことが出来ない。
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・情報処理の内容を更に進め、C言語を用いたプログラミング及び、情報処理のより実践的な技術を身につける。				
授業の進め方・方法	プログラミングの技術力向上のためには多くの演習問題を解く必要がある。したがって、本講義では多くの演習を行いながら進め、最後にまとめとしてグループによるプログラム制作演習をしてもらう。また、ほぼ毎週課題を課すので提出日を守って提出すること。				
注意点	<p>【注意】本授業では、多くの課題が課されるが、それらは全て以下の評価基準に含まれていることに注意すること。</p> <p>【事前学習】教科書をよく読み2年生までの学習内容を理解してから望むこと。</p> <p>【評価法・評価基準】</p> <ul style="list-style-type: none"> ・前期については、全面的に遠隔授業として実施する関係上、以下の通りとする：課題(80%)、小テスト(20%)で評価する。小テストは授業時間の一部を利用し、Moodleを用いて合計2回実施する。 ・後期については以下の通りとする：試験結果(60%)、課題(40%)で評価する。 <p>期末試験と作品制作によりC言語によるプログラミングの基礎の理解の程度を評価する。ただし、演習時間を多く取っているため授業態度の著しく悪いものや課題の未提出が25%を超えたものについては評価対象としないものとする。</p> <p>なお、前期と後期を総合した、総合成績60点以上を単位修得とする。</p>				
授業計画					
		週	授業内容	週ごとの到達目標	
前期	1stQ	1週	授業に関するガイダンス、機械語とプログラミング言語、C言語の基本(1)	開発環境や授業の進め方について理解できる。機械語とプログラミング言語の関係や違いが分かる。C言語の基本構文について理解できる。	
		2週	C言語の基本(2)[変数のスコープ、変数の宣言、型・定数、初期化、型変換とキャスト、条件式、真と偽、if、switch、三項演算子]	変数の詳細な使い方(宣言や初期化、型とキャストなど)と、条件分岐について理解ができ、利用できる。	
		3週	C言語の基本(3)[sizeof演算子、コメント、代入演算子等、関数、標準入力・出力、標準ライブラリ関数]	関数や標準入出力の使い方がわかり、利用できる。	
		4週	ポインタ(1)[メモリとアドレスとポインタ、ポインタを使った変数値の参照/書き換え、参照渡しによる関数呼び出し]	ポインタとアドレスの関係、メモリ内における変数の配置等について、基本的な概念が理解できる。	

2ndQ	5週	ポインタ (2) [配列とポインタ, 文字列リテラルとポインタ]	配列とポインタの関係や, C言語での文字列の扱い方について理解できる。	
	6週	ポインタ (3) [構造体, 動的配列]	構造体の概念が理解できる。 malloc()を用いた動的配列について理解できる。	
	7週	ファイルの入出力, コマンドライン引数	テキストファイルからの読み出し, 書き込みなどの基本的なファイルの入出力と, コマンドライン引数の利用方法が分かる。	
	8週	小テスト①, 課題解説	これまでの課題について振り返り, 理解が足りなかった内容について十分に理解ができる。	
	9週	データ構造 (1) [スタック構造]	スタック構造について理解でき, これまでに学んだことを用いて, 実装ができる。	
	10週	データ構造 (2) [キュー構造とリングバッファ]	キュー構造について理解でき, これまでに学んだことを用いて, 実装ができる。	
	11週	データ構造 (3) [リスト構造①]	リスト構造について理解でき, これまでに学んだことを用いて, 実装ができる。	
	12週	データ構造 (4) [リスト構造②]	リスト構造について理解でき, これまでに学んだことを用いて, 実装ができる。	
	13週	応用プログラミング演習 (1)	これまでに学んできたことを用いて, バブルソート, 挿入ソートなどのアルゴリズムが実装ができる。	
	14週	応用プログラミング演習 (2)	これまでに学んできたことを用いて, バブルソート, 挿入ソートなどのアルゴリズムが実装ができる。	
	15週	小テスト②, 前期のまとめ	これまでの課題について振り返り, 理解が足りなかった内容について十分に理解ができる。	
	16週			
	3rdQ	1週	配列の応用 (入射波)	配列の応用 (入射波) が理解できる。
		2週	配列の応用 (反射波)	配列の応用 (反射波) が理解できる。
		3週	配列の応用 (定在波)	配列の応用 (定在波) が理解できる。
		4週	微分方程式の応用 (減衰振動)	微分方程式の応用 (減衰振動) が理解できる。
5週		微分方程式の応用 (減衰振動)	微分方程式の応用 (減衰振動) が理解できる。	
6週		再帰処理の応用 (IIRフィルタ) 1	再帰処理の応用 (IIRフィルタ) 1 が理解できる。	
7週		再帰処理の応用 (IIRフィルタ) 2	再帰処理の応用 (IIRフィルタ) 2 が理解できる。	
8週		中間試験		
4thQ		9週	プログラム開発過程, プログラム作成演習 (グループに分かれこれまでの学習内容を用いてプログラムの設計, 実装, 評価の演習を行う) 1	プログラム開発過程について理解し, これまで学習してきたことを用いた応用的プログラムの作成ができる。
		10週	プログラム開発過程, プログラム作成演習 (グループに分かれこれまでの学習内容を用いてプログラムの設計, 実装, 評価の演習を行う) 2	プログラム開発過程について理解し, これまで学習してきたことを用いた応用的プログラムの作成ができる。
		11週	プログラム開発過程, プログラム作成演習 (グループに分かれこれまでの学習内容を用いてプログラムの設計, 実装, 評価の演習を行う) 3	プログラム開発過程について理解し, これまで学習してきたことを用いた応用的プログラムの作成ができる。
		12週	プログラム開発過程, プログラム作成演習 (グループに分かれこれまでの学習内容を用いてプログラムの設計, 実装, 評価の演習を行う) 4	プログラム開発過程について理解し, これまで学習してきたことを用いた応用的プログラムの作成ができる。
		13週	プログラム開発過程, プログラム作成演習 (グループに分かれこれまでの学習内容を用いてプログラムの設計, 実装, 評価の演習を行う) 5	プログラム開発過程について理解し, これまで学習してきたことを用いた応用的プログラムの作成ができる。
		14週	制作物のプレゼンテーション	課題制作物についてのプレゼンテーションが出来る。
		15週	期末試験	
		16週	後期のまとめ	期末試験の解答・まとめを自己確認できる。

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	ソフトウェア	アルゴリズムの概念を説明できる。	3	
				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	
				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	
				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	
				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	
				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	
				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	
				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	
				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	
				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	
				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。	3	
	計算機工学	基本的な論理演算を行うことができる。	3			

				基本的な論理演算を組合わせて、論理関数を論理式として表現できる。	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。	4	
				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	
				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	
				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	
				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	
				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。	4	
				要求仕様にあったソフトウェア(アプリケーション)を構築するために必要なツールや開発環境を構築することができる。	4	
				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。	4	

評価割合

	試験	課題	発表	資料作成	合計
総合評価割合	40	50	8	2	100
基礎的能力	40	50	0	0	90
創造的能力	0	0	8	2	10