東京工業高等専門学校	開講年度	令和03年度 (2021年度)	授業科目	プログラミング言語II

科目基礎情報

科目番号	0044	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	情報工学科	対象学年	2
開設期	後期	週時間数	2
教科書/教材	Webテキスト使用　http://coskx.webcrow.jp/mrr/for_students/CIntro/CIntro.html		
担当教員	西村 亮		

到達目標

C言語初級者がC言語プログラミングの修得を目標とする。具体的には，プログラムを読んでどのような動作が行なわれるか流れを追える力をつける。簡単な課題に関しては，自分で処理手順を考え，それをCプログラムで記述し，コンパイル・実行し，実行結果について検討ができるようになる。演習においては，PCでのエディタ・コンパイラ操作に慣れ，自分で解説を読んで，課題プログラムが作れること，ファイルの管理・ファイルの提出などもできるようになることが必要である。

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	標準的な到達レベルの目安（可）	未到達レベルの目安
変数とデータ型の概念を説明できる。	グローバル変数，スタティック変数，配列，構造体を説明でき，プログラミングに支障がない。	オート変数とデータ型が説明でき，それらを使ったプログラミングに支障がない。	オート変数とデータ型が説明でき，それらを使ったプログラミングにおおむね支障がない。	変数について説明できない。プログラムの振る舞いを追えない。
代入や演算子の概念を理解し，式を記述できる。	＝を使った代入の概念，演算子の優先，論理演算式を理解し，プログラミングに支障がない。	＝を使った代入の概念，論理演算式を理解し，プログラミングに支障がない。	＝を使った代入の概念，論理演算式を理解し，プログラミングにおおむね支障がない。	＝の意味を理解していないため，プログラムの振る舞いを追えない。
制御構造の概念を理解し，条件分岐や反復処理を記述できる。	制御構造の概念を理解し，条件分岐や反復処理やそれらの複合動作を理解し，意図通りのプログラムを作成できる。	制御構造の概念を理解し，条件分岐や反復処理やそれらの複合動作を理解し，プログラムの振る舞いを追うことができる。	制御構造の概念を理解し，条件分岐や反復処理やそれらの複合動作を理解し，プログラムの振る舞いをおおむね追うことができる。	制御構造の概念を理解できず，条件分岐や反復処理やそれらの複合動作も理解できていないため，プログラムの振る舞いを追うことができない。
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。	関数の概念，引数，関数の戻り値を理解し，意図通りにこれらを含むプログラムを作成できる。さらに与えられた課題を解く方法を関数に分解して見通しを良くすることが出来る。	関数の概念，引数，関数の戻り値を理解し，プログラムの振る舞いを追うことができる。	関数の概念，引数，関数の戻り値を理解し，プログラムの振る舞いをおおむね追うことができる。	関数の概念，引数，関数の戻り値を理解していないため，プログラムの振る舞いを追うことができない。
与えられた簡単な問題に対して，それを解決するためのソースプログラムを記述できる。	与えられた問題に対して，解決手順を考え，意図通りにソースプログラムを作成できる。	与えられた簡単な問題に対して，解決手順のヒントや友人の助けを借りて解決するためのソースプログラムを作成できる。	与えられた簡単な問題に対して，解決手順のヒントや友人の助けを借りて解決するためのソースプログラムをおおむね作成できる。	与えられた簡単な問題に対して，解決手順のヒント与えられても解決するためのソースプログラムを作成できない。
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。	様々な開発環境があることを知っており，それぞれの環境でプログラム作成が出来，コンパイル＆実行ができる。	与えられた開発環境でプログラム作成が出来，コンパイル＆実行ができる。	与えられた開発環境でプログラム作成が出来，コンパイル＆おおむね実行できる。	与えられた開発環境でプログラム作成が出来，コンパイル＆実行ができない。
主要な言語処理プロセッサの種類と特徴を説明できる。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。	コンパイラ，インタプリタ，アセンブラなどについて説明でき，C言語に関してはコンパイラ，リンカなどの機能について説明できる。	コンパイラ，インタプリタ，アセンブラについて説明でき，C言語に関してはコンパイラ，リンカの機能について説明できる。	コンパイラ，インタプリタ，アセンブラについて説明でき，C言語に関してはコンパイラ，リンカの機能についておおむね説明できる。	コンパイラ，インタプリタ，アセンブラについて説明できない。またC言語に関してもコンパイラ，リンカの機能について説明できない。
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理解している。主要な計算モデルを説明できる。	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理解している。主要な計算モデルを説明できる。	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理解している。	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おおむね理解している。	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理解していない。

学科の到達目標項目との関係

教育方法等

概要	C言語の修得ではプログラムのリーディングとライティングが重要である。この2つを交互に行いながら，C言語を修得できるようになっている。この科目は一斉授業がなじまないため，授業中においては自分でWebテキストのサンプルプログラムの解説を読み，文法の要点を理解して50個ほどの課題を演習する。定期試験においては基本的な文法を理解しているかどうか，サンプルプログラムの振る舞いを追跡させる。
授業の進め方・方法	授業中においては自分でWebテキストのサンプルプログラムの解説を読み，文法の要点を理解して50個ほどの課題を1人1台の使って演習する。出来上がった課題は実行結果とともにWeb提出し，課題受付システムによるチェック（インデンテーションなどの書式の整合性，コンパイラによるコンパイルエラーの有無）・担当教員による目視チェックを経て，課題が受理される。将来新しいプログラミング言語を習得する際も，解説を自分で読みながら理解する学習方法は有効であり，その学習方法も習得できる。定期試験においては基本的な文法を理解しているかどうかを，サンプルプログラムの振る舞いを追跡させる。定期試験で用いるサンプルプログラムは，演習中に思い通り動かなかったコンパイルエラーにはならなかったものを題材としている。
注意点	Webテキストを読む時は，文法上の要点に自分で「気づく」ことが大事である。そのため，気づいた要点のまとめの提出してもらう。課題の演習では，処理手順をよく考えるためのヒントもWebテキスト中にあるのでよく読むこと。課題にはどのように出力するか指示がある場合があるので，それに従うこと。課題の実行結果については正しいかどうか検討してから提出すること。課題の提出期限を守ることは当然だが，遅れそうな時は放課後も使って課題演習を行うこと。なお，課題がはやく終わってしまった場合は，自分のペースで先に進んでも良い。定期試験前には過去の問題を試験本番と同じように解いてみて，自分の理解度が正確であったかどうか，ケアレスミスを犯しやすいところはどこかよく検討してから本番に臨むこと。

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
後期	3rdQ	1週	組み込み関数の使い方，関数の作り方・使い方をを理解し，それを使ったプログラムを作成し，Web提出する。	sqrtなどの組み込み関数，自作関数を使ったプログラムの動作を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		2週	関数の作り方・使い方をを理解し，それを使ったプログラムを作成し，Web提出する。	自作関数を使ったプログラムの動作を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		3週	配列を扱う関数の作り方・使い方を理解し，それを使ったプログラムを作成し，Web提出する。	配列を扱う関数を使ったプログラムの動作を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		4週	文字列（char型配列）を扱う関数の作り方・使い方を理解し，それを使ったプログラムを作成し，Web提出する。	文字列（char型配列）を扱う関数の作り方・使い方を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		5週	段階詳細化プログラミングを体験し，段階詳細化手法でプログラムを作成しすることを目指し，関数部品プログラムを作成し，Web提出する。	段階詳細化プログラミングを理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		6週	段階詳細化プログラミングを体験し，段階詳細化手法でプログラムを作成し，Web提出する。	段階詳細化プログラミングを理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		7週	変数の型の混合演算，グローバル変数とオート変数。スタティック変数の振る舞いを理解し，それらを利用したプログラムを作成し，Web提出する。	各種変数の振る舞いを理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		8週	fprintfによるファイルへの出力方法を理解し，それを使ったプログラムを作成し，Web提出する。	fprintfによるファイルへの出力方法を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
	4thQ	9週	fscanfによるファイルへの入力方法を理解し，それを使ったプログラムを作成し，Web提出する。	fscanfによるファイルへの入力方法を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		10週	テキストファイルの入出力方法を理解し，それを使ったプログラムを作成し，Web提出する。	テキストファイルの入出力方法を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		11週	テキストファイルの入出力方法を理解し，それを使ったプログラムを作成し，Web提出する。	テキストファイルの入出力方法を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		12週	バイナリファイルの入出力方法を理解し，それを使ったプログラムを作成し，Web提出する。	バイナリファイルの入出力方法を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		13週	構造体を理解し，それを使ったプログラムを作成し，Web提出する。	構造体を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		14週	構造体を理解し，それを使った簡易データベースプログラムを作成し，Web提出する。	構造体を理解でき，それを使ったプログラムを作成できる。プログラムの各行逐次動作説明ができる。
		15週	学年末定期試験（後期中間試験を含む）	
		16週	プログラミング言語中でのC言語の位置づけ，C言語プロセッサの役割を学ぶ	

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	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5,後6,後7,後8,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7
				データ型の概念を説明できる。	4	後3,後4,後13,後14
				制御構造の概念を理解し、条件分岐を記述できる。	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
				制御構造の概念を理解し、反復処理を記述できる。	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
				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

				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
				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
				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。	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
	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】	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9,後10,後11,後12,後13,後14

評価割合

	試験	発表	相互評価	態度	ポートフォリオ	課題	合計
総合評価割合	75	0	0	0	0	25	100
基礎的能力	0	0	0	0	0	0	0
専門的能力	75	0	0	0	0	25	100
分野横断的能力	0	0	0	0	0	0	0