

東京工業高等専門学校		開講年度	平成29年度 (2017年度)	授業科目	プログラミング言語I
科目基礎情報					
科目番号	0052	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	前期	週時間数	2		
教科書/教材	https://xythos.tokyo-ct.ac.jp/web/j/usr/kosaka/studentreports/index.html				
担当教員	小坂 敏文				
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でもできるようになることが必要である。					
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		
変数とデータ型の概念を説明できる。	整数型、浮動小数点型、文字型変数を宣言して使用でき、計算上の特性を理解し、コンソールとの値のやり取りができる。またそれらの配列(文字列を含む)も自由に操作できる	整数型、浮動小数点型、文字型変数を宣言して使用でき、コンソールとの値のやり取りができる。簡単な配列操作ができる。	整数型、浮動小数点型、文字型変数を宣言も使用もできない、コンソールとの値のやり取りも出来ない。		
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を理解し、プログラムの振る舞いを追うことができる。	=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。		
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、解決手順のヒントを元に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&実行ができない。		
少なくとも一つの具体的なコンピュータシステムについて、起動・終了やファイル操作など、基本的操作が行える。	複数のコンピュータシステムについて、起動・終了やファイル操作など、基本的操作が行える。	演習室のコンピュータシステムについて、起動・終了やファイル操作など、基本的操作が行える。	演習室のコンピュータシステムについて、起動・終了やファイル操作など、基本的操作が行えない。		
少なくとも一つのWebブラウザを使って、Webブラウジングを行うことができる。	少なくとも一つのWebブラウザを使って、Webブラウジングを行うことができる。ユーザIDとパスワードを管理し、課題提出システムにアクセスできる。	少なくとも一つのWebブラウザを使って、Webブラウジングを行うことができる。ユーザIDとパスワードを管理し、課題提出システムにアクセスできる。	Webブラウザを使って、Webブラウジングを行うことができない。(これでは演習できない)		
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している。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(インテン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、ケアレスミスは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・Webテキストの紹介を行う。エディタによるプログラム作成・コンパイル&実行、課題のWeb提出を指導し、実際に提出させる。	この授業・授業方法の狙いを理解する。エディタでのプログラムの記述、コンパイル&実行、課題提出方法を理解する。	
		2週	Webテキストを自分で読んでprintfを使ったプログラムを作成し、実行結果を貼り付けて提出形式ファイルを作成し、Web提出する。	printfと書式制御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。int型、double型変数を使う。「=」の意味を理解でき、プログラムの流れを追うことができる。	
		3週	scanfの動作を理解し、それを使ったプログラムを作成し、Web提出する。	scanfで、実行時にデータを入力できる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。	

2ndQ	4週	forループの繰り返しを理解し、それを使ったプログラムを作成し、Web提出する。	forループを使った繰り返しのある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。
	5週	二重forループの繰り返しを理解し、それを使ったプログラムを作成し、Web提出する。	二重forループを使った繰り返しのある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。
	6週	forループを用いた連続加算のプログラムを作成し、Web提出する。whileループ、dowhileループの繰り返しを理解し、それを使ったプログラムを作成し、Web提出する。	forループを使った連続加算などの繰り返しのある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。
	7週	各種ループの使い分けを理解し、適切なループを用いたプログラムを作成し、Web提出する。	各種ループを使い分けて繰り返しのある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。
	8週	if, if-else, if-else if-else, switch caseによる、判断分岐構造を使ったプログラムを作成し、Web提出する。	判断分岐構造を持つ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。
	9週	ループと判断分岐構造を持つプログラムを理解し、その構造を用いたプログラムを作成し、Web提出する。	ループと判断分岐構造を持つ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。
	10週	ループと判断分岐構造を持つプログラムを理解し、その構造を用いたプログラムを作成し、Web提出する。	ループと判断分岐構造を意図通りに使い分け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。
	11週	配列の使い方を理解し、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。
	12週	char型変数の使い方、文字コードの扱い、printfにおける%cの使い方を理解し、それを使ったプログラムを作成し、Web提出する。	char型変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。
	13週	char型変数の配列である文字列の使い方を理解し、それを使ったプログラムを作成し、Web提出する。	char型変数の配列である文字列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。
	14週	二次元配列の使い方を理解し、それを使ったプログラムを作成し、Web提出する。	二次元配列を使ったプログラムを作成の動作を理解でき、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。
	15週	前期末定期試験（前期中間試験を含む）	
	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	変数とデータ型の概念を説明できる。	3	
				代入や演算子の概念を理解し、式を記述できる。	3	
				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。	3	
				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。	3	
				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。	3	
				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。	3	
				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。	2	
		その他の学習内容	少なくとも一つの具体的なコンピュータシステムについて、起動・終了やファイル操作など、基本的操作が行える。	3		

評価割合

	試験	発表	相互評価	態度	ポートフォリオ	その他	合計
総合評価割合	70	0	0	0	0	30	100
基礎的能力	0	0	0	0	0	0	0
専門的能力	70	0	0	0	0	30	100
分野横断的能力	0	0	0	0	0	0	0