

2025年度 授業シラバスの詳細内容

○基本情報			
科目名	C言語プログラミング（C Programming）		
ナンバリングコード	P20302	大分類 / 難易度 科目分野	情報メディア学科 専門科目 / 標準レベル プログラミング
単位数	2	配当学年 / 開講期	2 年 / 前期
必修・選択区分	コース必修:情報工学コース 選択:メディアデザインコース、こども・情報教育コース、情報コミュニケーションコース ※入学年度及び所属学科コースで異なる場合がありますので、学生便覧で必ず確認してください。		
授業コード	P030201	クラス名	-
担当教員名	吉森 聖貴		
履修上の注意、履修条件	理解を確実なものとするために、授業以外の時間の予習復習も含めて、課題の意味と、手順を十分に理解しながら数多くのプログラミング演習を行うことを勧めます。 遅刻・欠席をしないように心がけてください。やむを得ない事情で欠席した場合は、必ず次の授業の前までに内容の確認するとともに授業資料の受取に来てください。		
教科書	やさしいC第5版(ソフトバンククリエイティブ)高橋麻奈著		
参考文献及び指定図書	Cの絵本 第2版 C言語が好きになる新しい9つの扉 翔泳社 新・明解C言語 入門編 ソフトバンククリエイティブ 新・解きながら学ぶC言語 ソフトバンククリエイティブ		
関連科目	(先修科目)プログラミング基礎 (後修科目)Javaプログラミング, アルゴリズム, 画像処理及び演習, ソフトウェア開発入門, 組み込み演習, 情報メディア総合演習		

○基本情報		
授業の目的	本科目は、情報技術者として、情報処理・情報の創造や伝達・情報通信技術を活用した企業活動の各分野に従事するために必要となるプログラミングの知識と技術の修得を目的としています。 C言語はハードウェアの制御、組込みシステム、オペレーティングシステム(OS)など幅広い分野で利用されています。また、C言語は近年多く利用されているプログラミング言語(C++, Java、C#など)の基礎となった言語でもあり、この科目で学習する内容は、以後に履修するプログラミング、ソフトウェア開発関連科目の基礎となる大切な内容が数多く含まれます。 上記のような特徴をもつC言語について、基礎的な知識と技術を習得することが本科目の目的です。この科目の到達目標は、C言語の文法を理解し、簡単なプログラムを作成できるようになることです。	
	なお、C言語プログラミングの習得を目的とする本科目は、今日の社会において幅広い分野で必C言語は最も普及しているプログラム言語の一つであるとともに、現在主流となっているオブジェクト指向型のC系列言語(C++, Java、C#など)の元となった言語であるため、C言語の文法(構文)を学習することは、他のC系列言語の学習にも大いに役立ちます。また、特定のOSやハードウェアに依存しない移植性の高いプログラムを記述できることから、現在でもLinux上のプログラミングや、組込みソフトウェアの世界では主流のプログラム言語として使われています。この科目ではこのC言語の文法を解説するとともに、実際にプログラムを作成してその理解を深めます。	
授業の運営方法	(1)授業の形式	「演習形式」
	(2)複数担当の場合の方式	「該当しない」
	(3)アクティブ・ラーニング	「該当なし」
地域志向科目	該当しない	
実務経験のある教員による授業科目	該当しない	

○成績評価の指標		○成績評価基準(合計100点)		
到達目標の観点	到達目標	テスト (期末試験・中間確	提出物 (レポート・作品等)	無形成果 (発表・その他)
【関心・意欲・態度】	① 自発的に学習し、各課題に積極的に取り組むことができる			10点
【知識・理解】	② 基本的なC言語の文法を理解している	50点		
【技能・表現・コミュニケーション】				
【思考・判断・創造】	③ 簡単なC言語プログラムを作成できる ④ 与えられたC言語プログラムからその動作を理解できる		40点	

○成績評価の補足(具体的な評価方法および期末試験・レポート等の学習成果・課題のフィードバック方法)	
授業中に授業内容についての課題を出題します。かならず提出期限内に提出してください(期限を過ぎて提出された課題については減点します)。成績(無形成果)は授業や課題への取り組みを総合して評価します。なお、授業で出題した課題については、次週の講義中に模範解答を示します。	
成績評価における達成水準の目安は以下の通りです。 [Sレベル]単位を修得するために達成すべき到達目標を満たしている。 [Aレベル]単位を修得するために達成すべき到達目標をほぼ満たしている。 [Bレベル]単位を修得するために達成すべき到達目標をかなり満たしている。 [Cレベル]単位を修得するために達成すべき到達目標を一部分満たしている。	

○その他	
[授業の進め方について] 授業は配布プリントと教科書を併用する形で進めます。授業では必ず教科書を忘れないようにしてください。また、授業中に配布したプリントや授業中に使用した講義資料はWebページにて公開しますので、事前・事後学習に役立ててください。 プログラミングは教科書や資料を読んだり授業の話聞くだけでは上達しません。また、プログラミングはその内容をすべて記憶することよりも、教科書をはじめとする様々な資料を参考にしながらでも必要なプログラムを自身の力で作成できることの方が重要です。授業中はもちろんのこと、授業外でも必ず自身の手を動かしてプログラムを書く習慣を身に着けるとともに、できるだけ多くのプログラムを作成することを勧めます。	
[履修上の注意(補足)] 授業中は指示がない限りスマートフォンをカバンの中に片づけてください。スマートフォンの使用や授業に関係のないWebサイトの閲覧については減点します(これらの行為が授業の大半を占める場合は、欠席となる場合があります)。 プログラミングの知識や技術を学ぶ科目ですので、ソースコードは必ず自身で作成してください。他人のプログラムを複写したものや生成AIを使って生成したもの(またはその可能性が高いもの)については減点もしくは採点しません。 授業中に行う確認問題への解答は、別途指示がない限り授業時間中に行ってください。授業に出席していても確認問題に未解答の場合や授業時間後に解答した場合は、授業中の取り組み不十分として早退扱いとなることがあります。	
[参考文献及び指定図書(補足)] 新・明解C言語 中級編 ソフトバンククリエイティブ 新・明解C言語 実践編 ソフトバンククリエイティブ	
※1「数理データサイエンス」とは、適切な手法でデータを収集し、データの分析を行い、分析結果から得られた解釈を活用するという科学であり、今日の社会で広く必要とされている技術です。	

2025年度 授業シラバスの詳細内容

○授業計画	科目名 担当教員	C言語プログラミング（C Programming） 吉森 聖貴	授業コード	P030201
学修内容				
1. C言語プログラミングの準備				
この科目の目的, 到達目標, 学習内容, 学習方法（受講心得）について説明します。また, C言語とはどのようなプログラム言語であるのかについて説明します。				
予習	教科書の第1章「はじめの一步」を読んで, C言語の特徴を確認する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
2. C言語プログラミングの準備(演習)				
プログラミング作業に必要なテキストエディタの使い方, ファイル管理, OSの基本操作について, 演習を通して学習します。				
予習	教科書の第2章「C言語の基本」を読んで, C言語の特徴を確認する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
3. データ型と変数				
C言語で扱うことのできるデータの種類（データ型）と変数について, 簡単なプログラムを作成しながら学習します。				
予習	教科書の第3章「変数」を読んで, C言語でよく利用されるデータ型と変数を確認する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
4. データ型と変数(演習)				
第3回の授業で学んだデータ型と変数について, 演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。				
予習	データ型と式について, 第3回の授業で配布した資料や教科書を見て復習する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
5. 式と演算子				
C言語で利用可能な式と演算子について, 簡単なプログラムを作成しながら学習します。				
予習	教科書の第4章「式と演算子」を読んで, C言語でよく利用される代表的な演算子を確認する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
6. 式と演算子(演習)				
第5回の授業で学んだ式と演算子について, 演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。				
予習	式と演算子について, 第5回の授業で配布した資料や教科書を見て復習する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
7. 条件分岐処理(1) ～if－else文				
プログラムの流れを制御するしくみの1つ, 「条件分岐」について, if－else文を例に, 簡単なプログラムを作成しながら学習します。				
予習	教科書の第5章「場合に応じた処理」を読んで, 条件分岐(if文, if－else文, if－else if else)の処理の流れを確認			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
8. 条件分岐処理(1) ～if－else文(演習)				
第7回の授業で学んだ条件分岐処理について, 演習問題（プログラム）を自身で作成することでこの分野の理解を深めます。				
予習	条件分岐処理について, 第7回の授業で配布した資料や教科書を見て復習する。			約2時間
復習	授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間

○授業計画		科 目 名 担当教員	C言語プログラミング（C Programming） 吉森 聖貴	授業コード	P030201
学修内容					
9. 条件分岐処理(2) ～switch文 プログラムの流れを制御するしくみの1つ, 「条件分岐」について, switch文を例に, 簡単なプログラムを作成しながら学習します。					
予習		教科書の第5章「場合に応じた処理」を読んで, 条件分岐（switch文）の処理の流れを確認する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
10. 条件分岐処理(2) ～switch文(演習) 第9回の授業で学んだ条件分岐処理について, 演習問題（プログラム）を自身で作成することでこの分野の理解を深めます。					
予習		条件分岐処理について, 第9回の授業で配布した資料や教科書を見て復習する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
11. 繰り返し処理(1) ～for文 プログラムの流れを制御するしくみの1つ, 「ループ(繰り返し)」について, for文を例に, 簡単なプログラムを作成しながら学習します。					
予習		教科書の第6章「何度も繰り返す」を読んで, for文の処理の流れとネスト構造について確認する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
12. 繰り返し処理(1) ～for文(演習) 第11回の授業で学んだ繰返し処理について, 演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。					
予習		繰り返し処理について, 第11回の授業で配布した資料や教科書を見て復習する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
13. 繰り返し処理(2) ～while文、do-while文 プログラムの流れを制御するしくみの1つ, 「ループ(繰り返し)」について, while文, do-while文を例に, 簡単なプログラムを作成しながら学習します。					
予習		教科書の第6章「何度も繰り返す」を読んで, while文とdo-while文の処理の流れについて確認する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
14. 繰り返し処理(2) ～while文、do-while文(演習) 第13回の授業で学んだ繰返し処理について, 演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。					
予習		繰り返し処理について, 第13回の授業で配布した資料や教科書を見て復習する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
15. 配列 実用的なプログラムを作成する際に欠かせないデータ構造である「配列」について, 簡単なプログラムを作成しながら学習します。					
予習		教科書の第7章「配列」を読んで, 配列の仕組みを確認する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
16. 配列(演習) 第15回の授業で学んだ配列について, 演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。					
予習		配列について, 第15回の授業で配布した資料や教科書を見て復習する。			約2時間
復習		授業の内容を振り返るとともに, 講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間

2025年度 授業シラバスの詳細内容

○授業計画	科目名 担当教員	C言語プログラミング（C Programming） 吉森 聖貴	授業コード	P030201
学修内容				
17. 第1回演習課題				
これまでの授業で学んだ内容について，第1回目の演習課題が出ます。				
予習	第1回演習課題に取り組む準備として，これまでの授業で学んだ内容をもう一度確認する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された課題プログラム作成に取り組む。			約2時間
18. 第1回演習課題(つづき)				
第17回につづき，第1回目の演習課題に取り組めます。				
予習	第1回演習課題に取り組む準備として，第1回～第16回までに学んだ内容をもう一度確認する。			約2時間
復習	授業の内容を振り返るとともに，課題プログラム作成に取り組む。			約2時間
19. 関数				
実用的なプログラムを作成する際に欠かせない機能である，「関数」について，簡単なプログラムを作成しながら学習します。				
予習	教科書の第8章「関数」を読んで，関数の仕組みを確認する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
20. 関数(演習)				
第19回の授業で学んだ関数について，演習問題（プログラム）を自身で作成することでこの分野の理解を深めます。				
予習	関数について，第19回の授業で配布した資料や教科書を見て復習する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
21. ポインタ				
実用的なプログラムを作成する際に欠かせない機能である，「ポインタ」について，簡単なプログラムを作成しながら学習します。				
予習	教科書の第9章「ポインタ」を読んで，ポインタの仕組みを確認する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
22. ポインタ(演習)				
第21回の授業で学んだポインタについて，演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。				
予習	ポインタについて，第21回の授業で配布した資料や教科書を見て復習する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
23. 配列とポインタの応用				
実用的なプログラムを作成する際に欠かせない機能である，「配列」と「ポインタ」の関係について，簡単なプログラムを作成しながら学習します。				
予習	教科書の第10章「配列・ポインタの応用」を読んで，配列とポインタの関係を確認する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
24. 配列とポインタの応用(演習)				
第23回の授業で学んだ配列とポインタの応用について，演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。				
予習	配列とポインタの応用について，第23回の授業で配布した資料や教科書を見て復習する。			約2時間
復習	授業の内容を振り返るとともに，講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間

○授業計画		科 目 名 担当教員	C言語プログラミング（C Programming） 吉森 聖貴	授業コード	P030201
学修内容					
25. 構造体 実用的なプログラムを作成する際に欠かせない機能である、「構造体」について、簡単なプログラムを作成しながら学習します。					
予習		教科書の第11章「いろいろな型」を読んで、構造体の仕組みを確認する。			約2時間
復習		授業の内容を振り返るとともに、講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
26. 構造体(演習) 第25回の授業で学んだ構造体について、演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。					
予習		構造体について、第25回の授業で配布した資料や教科書を見て復習する。			約2時間
復習		授業の内容を振り返るとともに、講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
27. ファイル処理 実用的なプログラムを作成する際に欠かせない処理である、「ファイル処理」(ファイルの入出力)について、簡単なプログラムを作成しながら学習します。					
予習		教科書の第12章「ファイルの入出力」を読んで、ファイルへのアクセス方法について確認する。			約2時間
復習		授業の内容を振り返るとともに、講義中に出題された演習問題に取り組む（課題プログラムの作成）。			約2時間
28. ファイル処理(演習) 第27回の授業で学んだファイル処理について、演習問題を自身で解く（プログラムを作成する）ことでこの分野の理解を深めます。					
予習		ファイル処理について、第27回の授業で配布した資料や教科書を見て復習する。			約2時間
復習		授業の内容を振り返るとともに、講義中に出題された課題プログラム作成に取り組む。			約2時間
29. 第2回演習課題 これまでの授業で学んだ内容について、第2回目の演習課題が出ます。					
予習		第2回演習課題に取り組む準備として、第1回～第28回までに学んだ内容をもう一度確認する。			約2時間
復習		授業の内容を振り返るとともに、講義中に出題された課題プログラム作成に取り組む。			約2時間
30. 第2回演習課題(つづき) 第29回につづき、第2回目の演習課題に取り組めます。					
予習		第2回演習課題に取り組む準備として、第1回～第28回までに学んだ内容をもう一度確認する。			約2時間
復習		授業の内容を振り返るとともに、課題プログラム作成に取り組む。			約2時間
31. 期末試験 この科目で扱った知識項目の理解度を問うために、筆記形式の試験を行います。					
予習		期末試験に向けて、これまでに学んだすべての内容をもう一度確認する。			約2時間
復習		出題された問題の中で解答できなかった内容を再度見直す。			約2時間
32.					
予習					
復習					