

2022年度 授業シラバスの詳細内容

○基本情報			
科目名	プログラミング入門（Introduction to Programming）		
ナンバリングコード	E21103	大分類 / 難易度 科目分野	経営経済学科 専門科目 / 標準レベル IT・システム
単位数	2	配当学年 / 開講期	2 年 / 前期
必修・選択区分	選択 ※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	E004901	クラス名	-
担当教員名	本村 裕之		
履修上の注意、 履修条件	プログラミング言語を併せて受講することが望ましいです。 席数による受講者数制限を行うことがあります。 出席状況を直接の評価基準とするわけではありませんが、遅刻と欠席による授業内容の把握の遅れは、そのまま、課題提出の困難さを招きますので注意してください。		
教科書	教科書は使用しません。授業に必要な情報は、授業用ホームページを利用して配布します。		
参考文献及び指定図書	千葉則茂・村岡一信・小沢一文・海野啓明共著、Cアルゴリズム全科、近代科学社		
関連科目	プログラミング言語、情報システム論、システム分析論		

○基本情報		
授業の目的	プログラミング入門は、プログラム作成を通じ、コンピュータとプログラミングの基礎を学習する科目です。コンピュータがどんな機械であるのかを理解し、その上で、コンピュータにより問題を解決していくための方法を学ぶ基礎となる部分です。プログラムに対する理解・認識を確認し、プログラムの実行による数値計算、制御構造、配列、モジュールによるプログラムの作成を行い、併せて、デバッグの方法、データの処理・入出力法を学んでいきます。 本科目は、「数理・データサイエンス・AI 関連科目」である。 対応DP：自然や文化・伝統など幅広い視野に立って、産業界の要請に応える各分野の専門知識と実践的応用力を身につけていること。	
授業の概要	本科目は、コンピュータ関連科目の導入部分として重要な位置にあります。	
授業の運営方法	(1) 授業の形式	「演習等形式」
	(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】	統計に対する嫌悪感の払拭		20点	
【知識・理解】	講義で学ぶ知識と実際の生活の上で見えてくる新聞・テレビなど各種メディアから得られる情報との整合	20点		
【技能・表現・コミュニケーション】	評価対象外			
【思考・判断・創造】	C言語に対する理解と利用法に対する初歩的能力の取得	50点	10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)				
完全な形で提出をすること。提出に際しては対面による口頭試問を行い到達度を確認することでフィードバックとする。				

○その他
プログラミング入門は、「数理データサイエンス」(下記※1)について学ぶ 今日の社会において幅広い分野で必要とされるAI(人工知能) 関連の基礎科目、特にプログラミング能力の育成科目としても重要である。 ※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る技能です。

2022年度 授業シラバスの詳細内容

○授業計画		科 目 名 担当教員	プログラミング入門 (Introduction to Programming) 本村 裕之	授業コード	E004901
学修内容					
1. プログラミング入門					
コンピュータがただの機械であり、入力された情報に対してしか答えを返してこないことを理解します。同じ意味で、そのアプリケーションで何が出来るか、何が出来ないかを理解しておかなければいけません。					
予習		シラバスを熟読する			約2時間
復習		システム環境設定の確認を行う			約2時間
2. C言語のための基礎知識					
C言語の登場の背景、C言語を使って何が出来るかを理解します。また、C言語を使うために必要な環境について理解し、ハードウェアとソフトウェアの関係を学んでいきます。					
予習		前回の指導内容を確認の上、アプリケーションになれるための作業をする。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
3. C言語の基本型と表示					
ソースプログラム、コンパイル、デバッグの一連の作業を通して、プログラムを完成させていくことを学び、エラーメッセージに 対し適切な対応ができるようになるための準備を行います。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
4. 変数・演算子					
変数、または引数の意味を理解し、変数の宣言の方法とその分類の必要性を理解します。また、四則演算やインクリメント演 算子・デクリメント演算子等、一般によく使われる演算子を用い、変数の代入、出力等を学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
5. 関数 I					
私たちが実際に計算を行う際に、こういった構造で計算を行っているかを確認し、それをプログラム上に再現するためにどの ような構造を作らなければならないかを試すことで、関数の仕組みを理解していきます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
6. 関数II(標準関数)					
C言語には、予め用意された標準関数(組み込み関数)が存在します。そのような関数を用い計算を行う方法を学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
7. 入力					
データをキーボードから入力し、それを出力する方法を学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
8. 条件判断(if-else)					
if-else構文を使った条件判断を学び、その判断の真偽によってプログラムを分岐させていく方法を理解します。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間

○授業計画		科 目 名	プログラミング入門 (Introduction to Programming)	授業コード	E004901
		担当教員	本村 裕之		
学修内容					
9. 論理演算					
and、or、notといった論理演算子を学び、複雑な条件の表現を可能にする方法を学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
10. 繰り返し(for)					
イタレーション(繰り返し)の作業は一般の生活の中にもよく出てくるものですが、第10回と第11回の2回に渡って、そのやり方を学びます。第10回は、繰り返し回数が予めわかっている場合について学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
11. 繰り返し(while)					
予め繰り返し回数がわかっていない場合の方法について学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
12. ファイル処理					
コンピュータで作業を行う際には、プログラムの作成と同時に、そのプログラム上で行う処理に必要なデータの存在が不可欠です。予め他の記憶装置に保存されているデータの呼び出しと書き込み方法を学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
13. ポインタ・構造体					
記憶された変数のアドレスを記憶するポインタという機能と、複数のデータ型を組み合わせで独自のデータのリストを作る構造体という機能について学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
14. アルゴリズム構造					
プログラムを簡素化し、処理速度をあげるためには、アルゴリズムを理解することが必要です。各個人の能力に応じた問題解決のための道筋作り(アルゴリズム)の基本を学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
15. アルゴリズム構造2					
前回到引き続き、アルゴリズム構造について学びます。					
予習		前回の講義を受け講義内容の再生を行う。			約2時間
復習		配布資料を基に講義内容の精査を行う。			約2時間
16. 期末試験					
予習					
復習					