

2026 年度シラバス

科目分類/Subject Categories			
学部等/Faculty	/工芸科学部 : /School of Science and Technology	今年度開講/Availability	/有 : /Available
学域等/Field	/設計工学域 : /Academic Field of Engineering Design	年次/Year	/1年次 : /1st Year
課程等/Program	/情報工学課程・課程専門科目 : /Specialized Subjects for Undergraduate Program of Information Science	学期/Semester	/後学期 : /Second term
分類/Category	/:/	曜日時限/Day & Period	/木 1 : /Thu.1

科目情報/Course Information				
時間割番号 /Timetable Number	12222401			
科目番号 /Course Number	12260004			
単位数/Credits	2			
授業形態 /Course Type	演習 : Practicum			
クラス/Class				
授業科目名 /Course Title	ソフトウェア演習 I : Software Exercise I			
担当教員名 / Instructor(s)	/澁谷 雄/杜 偉薇/SIRIARAYA PANOTE : SHIBUYA Yu/DU Weiwei/SIRIARAYA PANOTE/			
その他/Other	インターンシップ実施科目 Internship	国際科学技術コース提供科目 IGP	PBL 実施科目 Project Based Learning	DX 活用科目 ICT Usage in Learning
			○	○
	実務経験のある教員による科目 Practical Teacher			
科目ナンバリング /Numbering Code				

授業の目的・概要 Objectives and Outline of the Course	
日	プログラミング I の講義内容を理解するために、問題演習を行う。C 言語処理系を使用して、C プログラムの作成、コンパイル、実行、デバッグを行い、データ構造の操作法やアルゴリズムのコーディング方法について実習し、C プログラムの読解と開発を行う能力を習得する。本演習では計算機を「プログラミングの対象」として扱い、高度な計算機利用法についても学ぶ。
英	This course provides a comprehensive set of exercises designed to learn essential concepts covered in 'Programming I'. The exercise encompasses coding, compiling, executing, and debugging of C programs to develop a special literacy in C language, data-structure handling and coding techniques. In this exercise, computer is used as a target of programming to understand some advanced computer usage methods.

学習の到達目標 Learning Objectives	
日	C コンパイラを用いてプログラムをコンパイルし、文法エラーを修正して実行ファイルを作成できる。 数値や文字列を入力し、変数に代入・加工した後、それらを整形出力するプログラムを読解・開発できる。 条件分岐、繰り返しなどの言語要素を用いた C プログラムを読解・開発できる。 関数を用いて構造化された C プログラムを読解・開発できる。 配列、文字列、構造体、ポインタなどの C 言語特有の言語要素を含む C プログラムを読解・開発できる。 ファイル操作を行う C プログラムを読解・開発できる。

	<p>計算機を「プログラミングの対象」として捉えることができる。</p> <p>To compile C programs, correct syntax errors using a C compiler, and generate the executable files.</p> <p>To grasp/develop programs that assign a numerical value or string to a variable, transform it, and display the result.</p> <p>To grasp/develop C program syntaxes, including conditional operators, loops, and more.</p> <p>To grasp/develop structured C programs that use functions.</p> <p>To grasp/develop some special syntaxes of C programs including arrays, strings, structures, and pointers.</p> <p>To grasp/develop C programs that manipulate files.</p> <p>To grasp that computers are used as 'a target of programming'.</p>
英	

学習目標の達成度の評価基準 / Fulfillment of Course Goals (JABEE 関連科目のみ)

日	
英	

授業計画項目 Course Plan

No.		項目 Topics	内容 Content
1	日	簡単な C プログラムの開発	<ul style="list-style-type: none"> ・ソフトウェア演習 I を受講する上で必要なルールと道具(Eclipse, Moodle)の使い方を学ぶ。 ・C プログラムのコンパイルの手順を学ぶ。 ・C プログラムのスケルトン (雛形) を元に文字列を出力するだけの簡単な C プログラムを開発する。
	英	Development of a simple C program	<ul style="list-style-type: none"> ・ To understand the essential rules and learn to use some tools such as Eclipse, moodle in Software Exercise I. ・ To learn C compilation procedures. ・ To develop a simple C program that outputs a string.
2	日	変数と四則演算	<ul style="list-style-type: none"> ・変数の宣言、代入、参照の基本を学ぶ。 ・数値を入力して変数に代入し、それらを加工して出力するプログラムを開発する。
	英	Variables and four arithmetic operations	<ul style="list-style-type: none"> ・ To learn the basic syntaxes of declarations, assignments, and references in variables. ・ To develop a program that assigns a numerical value to a variable, transforms it, and displays the result.
3	日	制御(I)条件分岐	<ul style="list-style-type: none"> ・関係演算子、if 文による判断・判定制御、複合条件の判定、switch-case 文による分岐制御、強制飛び越しについて学ぶ。 ・数値や文字を入力して条件判定を行い、それらを整形出力するプログラムを開発する。
	英	Control (I) Condition	<ul style="list-style-type: none"> ・ To learn relational operator, conditional evolution, and control using if statement, compound statements for branching control, and forced loop exits by using switch-case statements. ・ To develop a program where a numerical value or string is inputted
4	日	制御(II)繰り返し	<ul style="list-style-type: none"> ・ for 文、while 文、do-while 文による繰り返し制御について学ぶ。 ・数値や文字を繰り返し入力し、その値によって処理結果を整形出力したり、実行を終了させるプログラムを開発する。 ・デバッガの利用法も実習する。
	英	Control (II) Loop	<ul style="list-style-type: none"> ・ To learn iteration control using for, while and do-while statements. ・ To develop a program where a numerical value or string is inputted by iteration, the results are outputted by different inputs and finish the execution. ・ To practice using debugg
5	日	配列、文字列	<ul style="list-style-type: none"> ・配列、文字配列としての文字列の構造と使い方について学ぶ。 ・数値や文字列を入力し、配列を用いた処理を行い、結果を表示するプログラムを開発する。
	英	Array and string	Array and string
6	日	関数	<ul style="list-style-type: none"> ・関数の宣言、定義、呼び出しの方法について学ぶ。 ・数値を入力し、関数呼び出しによって処理を行い、結果を表示するプログラムを開発

	英	Function	<p>する。</p> <ul style="list-style-type: none"> ・ To learn the declaration, definition and call of function. ・ To develop a program that takes a numerical value as input, processes it using a function, and displays the result.
7	日	再帰	<ul style="list-style-type: none"> ・ 再帰関数呼び出し、変数の有効範囲と記憶域期間について学ぶ。 ・ 数値を入力し、再帰関数呼び出しによって処理を行い、結果を表示するプログラムを開発する。
	英	Recursion	<ul style="list-style-type: none"> ・ To learn recursive function calls, variable scope, and storage duration. ・ To develop a program that takes a numerical value as input, processes it using a recursive function call, and displays the result.
8	日	ポインタ(I)	<ul style="list-style-type: none"> ・ ポインタの構造と基本的な使い方について学ぶ。 ・ ポインタ変数を関数の引数としてとる場合や、ポインタと配列の関係について学ぶ。
	英	Pointer (I)	<ul style="list-style-type: none"> ・ To learn the structure and usage of pointer. ・ To learn the relation between pointers and arrays when a pointer is an argument of the function.
9	日	ポインタ(II)	<ul style="list-style-type: none"> ・ ポインタを介した文字列の操作について学ぶ。
	英	Pointer (II)	<ul style="list-style-type: none"> ・ To learn string operations using pointers.
10	日	ポインタ(III)	<ul style="list-style-type: none"> ・ 「ポインタの配列」とその使い方を学ぶ。 ・ コマンドライン引数を解析して表示するプログラムを開発する。
	英	Pointer (III)	<ul style="list-style-type: none"> ・ To learn pointer arrays and their usage. ・ To develop a program that analyzes and displays command-line arguments.
11	日	プログラム開発(I)	<ul style="list-style-type: none"> ・ プログラムの開発の手順(設計、コーディング、テスト、デバッグ、修正)について学ぶ。
	英	Program development (I)	<ul style="list-style-type: none"> ・ To learn the process of program development (design, coding, testing, debugging and modification).
12	日	構造体	<ul style="list-style-type: none"> ・ 構造体の定義、参照、代入の方法について学ぶ。 ・ 構造体を用いてデータの処理を行うプログラムを開発する。
	英	Structure	<ul style="list-style-type: none"> ・ To learn definition, reference, and assignment of structures. ・ To develop a program that processes data using structures.
13	日	プログラム開発(II)	<ul style="list-style-type: none"> ・ プログラムの開発の手順(設計、コーディング、テスト、デバッグ、修正)について学ぶ。
	英	Program development (II)	<ul style="list-style-type: none"> ・ To learn the process of program development (design, coding, testing, debugging and modification).
14	日	ファイル操作	<ul style="list-style-type: none"> ・ ファイルへのデータの入出力について学ぶ。 ・ ファイルからデータを入力して処理結果をファイルに出力するプログラムを開発する。
	英	File operation	<ul style="list-style-type: none"> ・ To learn file input and output operations. ・ To develop a program that reads data from files and displays the results.
15	日	プリプロセッサ	<ul style="list-style-type: none"> ・ プリプロセッサとその働きについて学ぶ。 ・ プリプロセッサ指令を用いて、単一ソースファイルから複数の実行ファイルを作成する。
	英	Preprocessor	<ul style="list-style-type: none"> ・ To learn preprocessor and its manipulation. ・ To generate multiple executable files from a single source file using preprocessor instruction.

履修条件 Prerequisite(s)	
日	<ul style="list-style-type: none"> ・情報リテラシー概論、情報工学概論を履修済みであること。 ・プログラミングIを並行して履修すること。
英	<ul style="list-style-type: none"> ・ To complete Introduction to Computer Literacy and Introductory Laboratory in Information Science. ・ To register for 'Programming I' simultaneously.

授業時間外学習（予習・復習等） Required study time, Preparation and review	
日	<p>本演習ではCコンパイラを用いてプログラミングを行う。情報工学概論を受講し、基本的なコンピューターの操作法を十分習得しておくこと。</p> <p>本科目はプログラミングの演習科目であり、プログラミングに関する予習（2時間）及び次週の小テスト実施に備えた復習（1時間）を要する。プログラミングは新しい概念であるため、積極的に質問するなど授業への熱心な取り組みが期待される。</p>
英	<p>This exercise focuses on writing programs using a C compiler. Basic computer operations should be well understood after completing Introductory Laboratory in Information Science.</p> <p>This course is an exercise in programming. It requires two hours to prepare by acquiring foundational knowledge of programming and one hour to complete a short test next week. Students are expected to make a serious effort to learn, including actively asking questions, as programming is a new concept.</p>

教科書／参考書 Textbooks/Reference Books	
日	教科書:なし(必要な情報は Moodle で提供する。プログラミングIの教科書は持参すること。)/参考書: 演習中に適宜紹介する。
英	Textbook: None (Materials will be provided on Moodle. Bring the Programming I textbook.) Reference books will be introduced during Software Exercise I.

成績評価の方法及び基準 Grading Policy	
日	出席およびレポートの提出が成績評価の前提となる。その上で、小テストならびに演習問題・自習問題レポートの評点(50%)、開発課題レポートの評点(30%)、レポートの提出状況(20%)により評価する。
英	Attendance and report submission are required for performance evaluation. The final score is composed of the scores from minute tests, exercises in Software Exercise I and at home (50%), the program development report (30%), and the score based on whether reports are submitted on time (20%).

留意事項等 Point to consider	
日	
英	