

科目	情報処理 (Information Processing)		
担当教員	加藤 真嗣		
対象学年等	電気工学科・2年・通年・必修・2単位 (学修単位I)		
学習・教育目標	工学複合プログラム	-	JABEE基準1(1) -
授業の概要と方針	1年で学んだ情報処理に続き、C言語プログラミングの講義を行う。本講義では、繰り返しで用いられるforとwhileの使い方、関数・配列・構造体・ポインタの作成方法および使い方、ファイルの操作方法を学ぶ。講義では、前半に説明を行い、後半に説明した内容に則した例題を演習する。筆記試験に加えてプログラミングの試験を行うことにより、C言語を用いてプログラミングできる実践力を効率よく身につける。		
	到達目標	達成度	到達目標毎の評価方法と基準
1	forを使った繰り返しのプログラムが作成できる。		forを使った繰り返しのプログラムを理解しているか、また二重のfor文を使った繰り返しのプログラムを理解しているか、プログラミング試験と中間試験で評価する。
2	whileおよびdo whileを使った繰り返しのプログラムが作成できる。		whileおよびdo whileを使った繰り返しのプログラムを理解しているか、forとの違いを理解しているか、プログラミング試験と中間試験で評価する。
3	関数の構造が説明でき、関数を使ったプログラムが作成できる。		関数の構造や作成方法を理解し、関数を使ったプログラムを作成できるか、プログラミング試験と定期試験で評価する。
4	配列の構造が説明でき、配列を使ったプログラムが作成できる。		配列の作り方や使い方を理解し、配列を使ったプログラムを作成できるか、プログラミング試験と定期試験で評価する。
5	構造体の構成が説明でき、構造体を使ったプログラムが作成できる。		構造体の構成を理解し、構造体を使ったプログラムを作成できるか、プログラミング試験と中間試験で評価する。
6	ポインタの働きが説明でき、ポインタを使ったプログラムが作成できる。		ポインタの働きを理解し、ポインタを使ったプログラムが作成できるか、プログラミング試験と中間試験で評価する。
7	ファイル操作を扱う関数を用いて、ファイルからのデータの読み込みや書き込みするためのプログラムが作成できる。		ファイルを扱う関数を理解し、実際にファイルを扱うことができるか、プログラミング試験と定期試験で評価する。
8	1年からこれまで学んできたC言語プログラミングの知識を用いて、要求されている処理が行えるプログラムが作成できる。		あるテーマを与え、それに応じたプログラムを作成することができるか、プログラミング試験で評価する。
9			
10			
総合評価	成績は、試験70%、プログラミング試験30%として評価する。		
テキスト	「C言語プログラミングレッスン」：結城浩著（ソフトバンク） 配布プリント		
参考書	「独習C」：シルト・ハーバード著，柏原正三監修（翔泳社）		
関連科目	情報処理(1年)，情報基礎		
履修上の注意事項	1年で学んだ情報処理が基礎となっているので、しっかり復習しておくこと。1年の情報基礎で学んだLinuxコマンドを頻繁に使用するので、しっかり復習しておくこと。必ず教科書，配布プリントは毎回持参すること。		

授業計画 1 (情報処理)		
週	テーマ	内容(目標, 準備など)
1	C言語プログラミングの復習	1年で学んだC言語プログラミング(四則演算, 変数, if文, switch文)を復習する。
2	forの使い方の説明とプログラミング演習	繰り返しに用いるforは決められた回数の繰り返しを行う構文である。forの使い方について説明した後, 簡単な例題を演習することにより, forの使い方を習得する。
3	二重のforの使い方の説明とプログラミング演習	forを二重にして使うことにより, 簡単により複雑な繰り返しが行えるようになる。二重のforの使い方について説明した後, 簡単な例題を演習することにより, 二重のforの使い方を習得する。
4	whileの使い方の説明とプログラミング演習	繰り返しに用いるwhileは決められた条件の下で繰り返しを行う構文である。whileの使い方について説明した後, 簡単な例題を演習することにより, whileの使い方を習得する。
5	do-whileの使い方の説明とプログラミング演習	whileは繰り返し処理を行う前に条件を調べるが, do-whileを用いると繰り返し処理を行った後に条件を調べる。whileとdo-whileの違いについて説明した後, 簡単な例題を演習することにより, whileとdo-whileの違いを理解する。
6	総合演習	繰り返し処理に用いるfor, while, do-whileを用いて, より複雑な繰り返し処理を行うプログラムを作成する。
7	プログラミング試験	繰り返しに用いるfor, while, do-whileに重点をおいたプログラミング試験を実施する。
8	中間試験	中間試験を実施する。
9	中間試験とプログラミング試験の解答	中間試験およびプログラミング試験の解答を行う。
10	引数および戻り値をもたない関数の作り方の説明とプログラミング演習	関数とはある決まった処理を行うプログラムである。これまでに用いた関数を例に挙げながら関数について説明した後, オリジナルな関数を作る簡単な例題を演習することにより, 関数の作り方を習得する。
11	引数および戻り値をもつ関数の作り方の説明とプログラミング演習	引数や戻り値をもつ関数の使い方を説明した後, 引数と戻り値をもつオリジナルな関数を作る簡単な例題を演習する。これにより, 引数や戻り値を関数にもたせることにより, 関数がより汎用性をもつことを理解する。
12	一次元配列の作り方の説明とプログラミング演習	C言語で変数を使うときには, 個々の変数に対して個別に定義を必要がある。しかし, 変数が多くなると定義をすることが大変になる。そこで, 配列が使われる。配列は変数に対して番号を付けたものである。簡単な一次元配列の作り方について説明した後, 簡単な例題を演習することにより, 一次元配列の作り方を習得する。
13	二次元配列の作り方の説明とプログラミング演習	二次元配列とは変数に対して縦方向と横方向に番号をつけた配列である。したがって, 一次元配列に比べてより多くの番号が配列につけられるため, 多くの値を保存することができる。そこで, 簡単な二次元配列の作り方について説明した後, 簡単な例題を演習することにより, 二次元配列の作り方を習得する。
14	総合演習	関数と配列を組み合わせたプログラムを作成する。
15	プログラミング試験	関数と配列に重点をおいたプログラミング試験を実施する。
16	定期試験とプログラミング試験の解答	定期試験およびプログラミング試験の解答を行う。
17	構造体の説明とプログラミング演習	構造体とは, 複数の異なる型の変数を1つにまとめたものである。構造体は配列と異なるため, 初めにこの点について説明する。次に, 構造体の宣言・定義・代入・参照の方法について説明する。その後, 簡単な例題を演習することにより, 構造体の作り方を習得する。
18	メンバに配列をもつ構造体の説明とプログラミング演習	構造体のメンバとして配列をもつ構造体について説明した後, 簡単な例題を演習することにより, メンバに配列をもつ構造体の使い方を習得する。
19	ポインタの説明とプログラミング演習	ポインタとは, 変数のアドレスをもつ変数である。初めに, 変数とポインタの違いについて説明し, ポインタの定義・代入・参照について説明する。そして, 簡単な例題を演習することにより, ポインタの使い方を習得する。
20	配列・構造体とポインタの関係の説明とプログラミング演習	ポインタは, 配列や構造体にも適用できる。配列・構造体とポインタの関係について説明した後, 簡単な例題を演習することにより, 配列・構造体とポインタの関係について理解する。
21	総合演習	構造体とポインタを組み合わせたプログラムを作成する。
22	プログラミング試験	構造体とポインタに重点をおいたプログラミング試験を実施する。
23	中間試験	中間試験を実施する。
24	中間試験とプログラミング試験の解答	中間試験およびプログラミング試験の解答を行う。
25	ファイル操作の説明とプログラミング演習	既存のファイルを開いたり閉じたり, 開いたファイルから文字を読み込んだり, 開いたファイルに文字を書き込んだりする関数について説明する。そして, 簡単な例題を演習することにより, ファイル操作の関数の使い方を習得する。
26	書式つき入出力関数の説明とプログラミング演習	書式つき入出力関数について説明した後, 簡単な例題を演習することにより, 決められた書式でファイルからデータを読み込んだり, ファイルにデータを書き込む方法を習得する。
27	総合演習	ファイル操作の関数を使ったプログラムを作成する。
28	総復習	これまで学んできたC言語について, 基本的な構文や演算子の優先順位などに重点をおいて復習する。
29	総仕上げのプログラミング演習	これまで学んできたC言語について復習するために, あるテーマを与えて, 総合的なプログラムを作成する。
30	プログラミング試験	ファイル操作の関数に重点をおいたプログラミング試験を実施する。
備考	中間試験および定期試験を実施する。プログラミング試験を実施する。	