

科目名	プログラミング基礎Ⅱ	英文表記	Programming II	2017/3/13			
科目コード	2204						
教員名:神里 志穂子 技術職員名:佐竹 卓彦				作成			
対象学科/専攻コース	学年	必・選	履修・学修	単位数	授業形態	授業期間	
情報通信システム工学科	2年	必	履修	2単位	講義	通年	
科目目標 【MCC目標】	<p>プログラミングの構造を理解し、構造体、ポインタなどを利用したプログラムを作成できる。 【V-D-1】①プログラミング言語Cの制御構造の概念を理解し、これらを含むプログラムを記述できる。 【V-D-1】②変数型と配列・ポインタ・構造体を理解し、これらを含むプログラムを記述できる。 【V-D-1】③スタック・メモリマップの概念を理解し、スタック・メモリマップの概念を理解し、与えられた課題に対して、ソフトウェア生成に必要なツール類を用いてソースプログラムを生成し実行できる。</p>						
総合評価	<p>前期・後期評価:定期試験(中間・期末)の平均の70%+課題演習30%とする。 学年末評価は前期評価と後期評価の平均で行い、60%以上を合格とする。</p>						
科目達成度目標	目標割合	科目達成度目標	達成度目標の評価方法	ルーブリック			
				理想的な到達レベル	標準的な到達レベル	最低限必要な到達レベル	セルフチェック
	40%	① プログラミング言語Cの制御構造の概念を理解し、これらを含むプログラムを記述できる。	正しく説明できるか、正しく記述できるかを、定期試験および講義での課題に対する回答状況により評価する。定期試験70%+課題演習30%とする。	授業で学習した内容と関連付けながら、作成したプログラムの要点を説明できる。	講義資料や参考図書の内容に従い、プログラミング言語Cを用いて作成したプログラムの要点を説明できる。	講義資料や参考図書を参照しながら、プログラミング言語Cを用いて簡単なプログラムを作成できる。	
	30%	② 変数型と配列・ポインタ・構造体を理解し、これらを含むプログラムを記述できる。	正しく説明できるか、正しく記述できるかを、定期試験および講義での課題に対する回答状況により評価する。定期試験70%+課題演習30%とする。	授業で学習した内容と関連付けながら、変数型と配列・ポインタ・構造体を用いたプログラムの要点を説明できる。	講義資料や参考図書の内容に従い、変数型と配列・ポインタ・構造体を用いたプログラムの要点を説明できる。	講義資料や参考図書を参照しながら、変数型と配列・ポインタ・構造体を用いたプログラムを作成できる。	
30%	③ スタック・メモリマップの概念を理解し、与えられた課題に対して、ソフトウェア生成ツールを用いて解決するソースプログラムを記述し実行できる。	正しく説明できるか、正しく記述できるかを、定期試験および講義での課題に対する回答状況により評価する。課題演習30%とする。	授業で学習した内容と関連付けながら、スタック・メモリマップの概念を理解し、プログラミング言語Cを用いて作成したプログラムの要点を説明できる。	講義資料や参考図書の内容に従い、スタック・メモリマップの概念を理解し、プログラミング言語Cを用いて作成したプログラムの要点を説明できる。	講義資料や参考図書を参照しながら、スタック・メモリマップの概念を理解し、プログラミング言語Cを用いて簡単なプログラムを作成できる。		
本科・専攻科教育目標	1	2	3	4	<本科教育目標> (1)技術者に必要な基礎知識を備え、実践力のある人材を育成する (3)専門的基礎知識を理解し、自ら学ぶことのできる人材を育成する		
評価方法と評価項目および関連目標に対する評価割合							
	目標との関連	定期試験	小テスト	レポート	その他(演習課題・発表・実技・成果物等)	総合評価	セルフチェック
評価項目		70	0	0	30	100	
基礎的理解	①②③	40			10	50	
応用力(実践・専門・融合)	②③	15			5	20	
社会性(プレゼン・コミュニケーション・PBL)						0	
主体的・継続的学修意欲	①②③	15			15	30	
授業概要、方針、履修上の注意	講義の前半で、パワーポイントによる講義資料に基づいて文法規則や要点について説明を行う。例題を通じて注意点について解説し、各自のPCでのプログラム作成・コンパイル・実行・デバッグ等の演習を通じ、基本的な構文の復習から画像処理応用例までを学ぶ。毎回講義形式と演習問題への取り組みの形式で進め、項目修了時には確認の為の課題を課し、実行結果とソースコードを提出させることによりプログラミング能力を修得させる。演習時間中に個別指導や対応を通じ、不明な点や理解できない点が残らないように指導する。演習結果はサーバの指定フォルダに格納し、個別理解度の把握に利用する。						
教科書・教材	教員自作のテキスト、パワーポイント・プレゼン資料。 「プログラミング言語C第2版」(共立出版株式会社) 「C実践プログラミング第3版」(オライリー) (他にも参考図書を探す場合のキーワード:C言語 プログラミング)						

授 業 計 画					
週	授 業 項 目	時間	授 業 内 容	自学自習 (予習・復習)内容	セルフ チェク
1	年間のガイダンス	2	シラバスの説明(授業の概要、進め方)。		
2	WSプログラム開発環境の	2	ソースファイル作成・コンパイル・実行環境の確認。	先週の講義内容・演習課題の復習	
3	プログラムの作成手順の理解	2	ソースファイル作成・コンパイル・実行手順の修得。	先週の講義内容・演習課題の復習	
4	プログラム開発環境の構築	2	各自のPCへの開発環境を実装する。	先週の講義内容・演習課題の復習	
5	プログラムの作成手順の理解	2	PC環境でのコンパイル・実行手順の修得。	先週の講義内容・演習課題の復習	
6	変数について復習する	2	変数について復習し、演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
7	アドレスの理解	2	アドレスとアクセス法について学ぶ。	先週の講義内容・演習課題の復習	
8	前期中間試験	2	プログラミングの基礎知識に関する確認試験。	前期中間試験の講義内容・課題復習	
9	スタックの理解	2	スタックについて学ぶ。	先週の講義内容・演習課題の復習	
10	メモリ・マップの理解	2	メモリ・マップについて学ぶ。	先週の講義内容・演習課題の復習	
11	ローカル変数の理解と演習	2	ローカル変数について復習し、演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
12	ライブラリとリンクの理解	2	ライブラリとリンクについて学び、演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
13	型に関する理解	2	型の役割とキャストについて学ぶ。	先週の講義内容・演習課題の復習	
14	型とポインタに関する理解	2	typedefによる型の宣言を学ぶ。	先週の講義内容・演習課題の復習	
15	配列とポインタに関する理	2	ポインタと配列の関係について学ぶ。	先週の講義内容・演習課題の復習	
期末	期末試験	[2]			
16	文字列と配列に関する復習	2	文字列と配列について復習する。	先週の講義内容・演習課題の復習	
17	文字列とポインタに関する理解	2	文字列とポインタとの関係について学ぶ。	先週の講義内容・演習課題の復習	
18	文字列とポインタに関する演習	2	文字列とポインタについて演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
19	配列とポインタに関する理	2	配列とポインタとの関係について学ぶ。	先週の講義内容・演習課題の復習	
20	配列とポインタに関する演	2	配列とポインタについて演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
21	文字列定数に関する理解	2	文字列定数とNULLポインタについて学ぶ。	先週の講義内容・演習課題の復習	
22	メモリ領域破壊に関する理解	2	メモリ破壊について学ぶ。	先週の講義内容・演習課題の復習	
23	後期中間試験	2	ポインタの基礎知識に関する確認試験。	後期中間試験の講義内容・課題復習	
24	mallocとfreeに関する理解	2	mallocとfreeについて学び、演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
25	メモリ・リークに関する理解	2	メモリ・リークについて学ぶ。	先週の講義内容・演習課題の復習	
26	関数とポインタに関する理	2	関数と戻り値、引数へのポインタについて学ぶ。	先週の講義内容・演習課題の復習	
27	構造体メンバとポインタに関する理解	2	構造体メンバとポインタについて学び、演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
28	構造体引数に関する理解	2	構造体を引数について学ぶ。	先週の講義内容・演習課題の復習	
29	構造体とポインタに関する理解	2	構造体へのポインタ配列について学ぶ。	先週の講義内容・演習課題の復習	
30	構造体とポインタに関する演習	2	構造体へのポインタ配列について演習課題を通じ理解を深める。	先週の講義内容・演習課題の復習	
期末	期末試験	[2]			
学習時間合計		60	実時間	45	
自学自習(予習・復習)内容(学修単位における自学自習時間の保証)				標準的所用時間(試行)	
①	課題演習(その週の講義内容に沿った内容について演習課題を課す。)			各2時間×30回	
②	前期・後期中間テスト後に、各10問ずつ演習課題を課す。			各5時間×2回	
③					
備考欄					
(各科目個別記述) ・この科目の主たる関連科目は、プログラミング基礎I(1年)、アルゴリズムとデータ構造(3年)、オペレーティングシステム(3年)、応用プログラミングI(4年)、応用プログラミングII(5年)である。 (モデルコアカリキュラム) ・対応するモデルコアカリキュラム(MCC)の学習到達目標、学習内容およびその到達目標を【】内の記号・番号で示す。 (航空技術者プログラム) ・【航】は航空技術者プログラムの対応項目であることを意味する。					