

科目名 (Eng)		情報工学応用 (Applied Information Engineering)						
担当教員		大槻 正伸						
対象学年等	学科・学年		授業期間・区分・単位数・時間数			分野	形態	学修単位科目
	電気工学科	3	通年	必修	2	60	専門	B
目標基準との対応	福島高専の教育目標との対応：(B-3)．(B-4)．(E-2)． 卒業時に身に付けるべき学力や資質・能力との対応：2)．5)． JABEE基準1(1)との対応：							
授業の概要と方針	より高度な、効率よく問題を解くためのプログラミング技術を学ぶ。また構造化プログラミング、数値解析の初歩についても学ぶ。							
到達目標	①PADにより基本的なアルゴリズムの表現ができ、それをプログラム化できる。 ②再帰呼び出しを用いてプログラムが設計できる。 ③ソートングのプログラムが設計でき、効率の良し悪しが判断できる。 ④台形公式による数値積分、Newton法等による方程式解法プログラムが設計できる。 ⑤微分方程式を解く数値計算プログラムが設計できる。							
授業計画								
	週	授業項目	理解すべき内容				事前学習	
前期	1	PADによる表現	PADとは何か、PADによるアルゴリズムの表現				情報工学基礎の内容の復習	
	2	PADと判断、繰り返し	PADによる判断、繰り返しの表現					
	3	PADとC言語 1	C言語の基本的な文法、変数の型					
	4	PADとC言語 2	PADで表現されたアルゴリズムをプログラムに直す				PADによる表現、C言語基本文法の復習	
	5	プログラムの実行	簡単なプログラムの入力、実行、保存					
	6	プログラミング演習	プログラミング演習					
	7	前期中間試験						
	8	判断文	if文を用いたプログラム				C言語の文法の復習	
	9	繰り返し 1	for文による繰り返し					
	10	繰り返し 2	配列とfor文による繰り返し					
	11	繰り返し 3	while文による繰り返し				if文、for文の使い方の復習	
	12	様々な応用プログラム	プログラムによる様々な数列計算、2分探索法				VBの関数、Subプロシージャ	
	13	プログラミング演習	データの合計、平均、最大値、標準偏差					
	14	関数と手続き	関数、手続きとは何か、Local変数とGlobal変数					
	15	関数、手続きの具体例	関数や手続きを用いた様々なプログラム作成					
後期	16	再帰呼び出し 1	再帰呼び出しとは、再帰呼び出しを用いたプログラム				C言語の関数の理解	
	17	再帰呼び出し 2	再帰呼び出しと数列、漸化式、ハノイの塔の解法					
	18	ソートング問題	ソートング問題とソートングアルゴリズム					
	19	ソートングと計算量	ソートングアルゴリズムとデータ比較回数、計算量				これまでの授業内容理解、特にfor文によるループ、再帰呼び出しの復習	
	20	クイックソート	クイックソートと計算量					
	21	クイックソート設計	クイックソートプログラムの設計と入力実行、デバッグ					
	22	後期中間試験						
	23	クイックソート実行、解析	クイックソートプログラム入力と実行				y=f(x)のグラフと方程式の 微分方程式の意味の理解	
	24	クイックソート実行、解析	クイックソートプログラムデバッグ、計算時間解析					
	25	数値積分	台形公式による数値積分プログラム					
	26	方程式の解法	代数方程式とNewton法、2分探索法					
	27	微分方程式の解法 1	微分方程式を数値的に解くとはどういうことか					
	28	微分方程式の解法 2	グラフィック命令、Euler法とRunge-Kutta法					
	29	微分方程式の解法 3	Euler法によるプログラミング					
	30	微分方程式の解法 4	Runge-Kutta法によるプログラミング					
試験について	前期試験は実施する。後期試験は実施する。							
評価方法	定期試験の成績を50%、小テストや課題の総点を50%として総合的に評価する。							
教科書	・エンジニアのためのプログラミング入門 ーVB.NETによるプログラミングの基礎ー、大槻他編著、電気書院、 ・配布プリント							
参考書	Visual C++2005 ビギナー編、林 晴比古著、ソフトバンク クリエイティブ (株)							
関連科目	情報工学基礎							
履修上の注意	単に正しく動くプログラムを作成できるというだけでなく、見やすく、効率的に様々な問題を解くプログラムを設計できるようになることが重要である。							