

徳山工業高等専門学校		開講年度	平成30年度 (2018年度)		授業科目	コンピュータ工学
科目基礎情報						
科目番号	0064		科目区分	専門 / 必修		
授業形態	講義		単位の種別と単位数	履修単位: 2		
開設学科	情報電子工学科		対象学年	2		
開設期	通年		週時間数	2		
教科書/教材	テキスト：「TeC教科書」徳山高専情報電子工学科（必要に応じて参考資料を配布する）					
担当教員	古賀 崇了,浦上 美佐子					
到達目標						
クロス開発したプログラムを実際にTeCマイコンで動作させることによってコンピュータの仕組みや内部動作が理解できるようになることを目標とする。						
ルーブリック						
	理想的な到達レベルの目安		標準的な到達レベルの目安		未到達レベルの目安	
クロス環境における開発の実践	クロスアセンブラによるプログラム開発の学習を通してコンピュータの仕組みや内部動作を深く理解している。様々な応用課題についてクロス環境における開発とTeCマイコンによる実際の動作確認を通じて、プログラム開発を実践することができる。		クロスアセンブラによるプログラム開発の学習を通してコンピュータの仕組みや内部動作を深く理解している。様々な基本課題についてクロス環境における開発とTeCマイコンによる実際の動作確認を通じて、プログラム開発を実践することができる。		コンピュータの仕組みや内部動作を理解していない。様々な基本課題についてクロス環境における開発とTeCマイコンによる実際の動作確認を通じたプログラム開発を実践することができない。	
学科の到達目標項目との関係						
到達目標 C 1						
教育方法等						
概要	1年次に履修した基礎コンピュータ工学の内容のさらなる応用について講義と演習を通して修得する。まずコンピュータの入出力手法を学び、入出力プログラムの動作を理解する。また、クロスアセンブラによるプログラム開発の学習を通してコンピュータの仕組みや内部動作の理解を深める。その後、様々な応用課題についてクロス環境における開発とTeCマイコンによる実際の動作確認を通じて、プログラム開発の実践力を身に付ける。					
授業の進め方・方法	各回の授業では、前半に講義による課題の解説を行い、後半にプログラミング演習を行う。翌週の講義の最初に演習の解説を行う。理解度を確保するために項目毎に演習課題の一部を提出し、学習シートの役割とする。定期試験は主に例題、演習課題から出題する。					
注意点	【関連科目】基礎コンピュータ工学（1年）、コンピュータシステム概論（3年）、コンピュータアーキテクチャ（4年）					
授業計画						
		週	授業内容	週ごとの到達目標		
前期	1stQ	1週	入出力の仕組み	シリアル入出力の原理とマイコン上でのプログラミング方法を学ぶ。パソコンと通信できることを確認する。〔演習〕		
		2週	入出力プログラム	入出力する文字コードについて学び、入出力プログラムを作成し、動作を確認する。〔演習〕		
		3週	クロス環境によるシステム操作	クロスアセンブラによるアセンブルの手順を説明し、クロスア環境によるシステム操作を学ぶ。〔演習〕		
		4週	クロスアセンブラによるプログラミング	クロスアセンブラによる基本プログラミングを理解し、プログラムを動作させる方法を体得する。〔演習〕		
		5週	掛算・割算プログラム	シフトによる掛算、繰返しによる掛算・割算プログラムを作る。〔演習〕		
		6週	スタック操作	スタックの概念を解説し、スタック操作及びスタックを用いたレジスタ操作を学ぶ。〔演習〕		
		7週	総合演習	入出力プログラム、基本プログラムに関する総合演習問題のプログラミングを行う。〔演習〕		
		8週	サブルーチン	サブルーチン化について学び、サブルーチンを用いた入出力プログラムを作る。〔演習〕		
	2ndQ	9週	中間試験	入出力プログラム、基本プログラム、スタック操作が理解できているか確認する。		
		10週	2進数表示	レジスタの内容を2進数に変換する方法を学び、2進数で表示するプログラムを作る。〔演習〕		
		11週	16進数表示	レジスタの内容を16進数に変換する方法を学び、16進数で表示するプログラムを作る。〔演習〕		
		12週	文字列操作	メモリ上の文字列を操作するプログラムを作る。〔演習〕		
		13週	メモリダンプ	メモリの内容を16進数表示するプログラムを作る。〔演習〕		
		14週	総合演習	サブルーチンを使った2進数、16進数表示に関する総合演習問題のプログラミングを行う。〔演習〕		
		15週	期末試験	サブルーチンを使ったプログラム、2進数、16進数変換、メモリ上の文字列の操作ができるかを確認する。		
		16週	答案返却など	試験の解答と解説を行う。		
後期	3rdQ	1週	メモリ操作	連続したメモリ操作の手法について学び、プログラムの実行結果を確認する。〔演習〕		
		2週	メモリクリアとメモリコピー	メモリクリアとメモリコピーするプログラムを作る。〔演習〕		
		3週	文字列入力	キーボード入力した文字列を操作するプログラムを作る。〔演習〕		
		4週	文字列編集	メモリ上の文字列を編集するプログラムを作る。〔演習〕		

		5週	10進数表示	レジスタの内容を10進数に変換する方法を学び、10進数で表示するプログラムを作る。〔演習〕
		6週	10進数表示	レジスタの内容を符号付き10進数で表示するプログラムを作る。〔演習〕
		7週	総合演習	連続したメモリ操作、10進数表示に関する総合演習問題のプログラミングを行う。〔演習〕
		8週	中間試験	連続したメモリ操作、文字列操作、10進数変換に関するプログラミングが理解できているか確認する。
	4thQ	9週	ソフトタイマー	命令を繰返し実行して一定時間のソフトタイマーを作る。〔演習〕
		10週	音の出力	ソフトタイマーを使ってブザーとスピーカーから音を出力する方法を学ぶ。〔演習〕
		11週	電子オルゴール	音符データと電子オルゴールプログラムを作る。〔演習〕
		12週	コンソール割込み	割込みの概念を解説し、コンソール割込みによる割込み処理プログラムの動作を学ぶ。〔演習〕
		13週	タイマー割込み	タイマー割込みによる割込み処理プログラムを作る。〔演習〕
		14週	総合演習	タイマーサブルーチン、割込みに関する総合演習問題のプログラミングを行う。〔演習〕
		15週	期末試験	タイマー、割込み操作が理解できているかを確認する。
		16週	答案返却など	試験の解答と解説を行う。

モデルコアカリキュラムの学習内容と到達目標

分類		分野	学習内容	学習内容の到達目標	到達レベル	授業週
基礎的能力	工学基礎	情報リテラシー	情報リテラシー	論理演算と進数変換の仕組みを用いて基本的な演算ができる。	3	前10
				コンピュータのハードウェアに関する基礎的な知識を活用できる。	2	前1
				同一の問題に対し、それを解決できる複数のアルゴリズムが存在していることを知っている。	1	前5
				与えられた基本的な問題を解くための適切なアルゴリズムを構築することができる。	3	前5
				任意のプログラミング言語を用いて、構築したアルゴリズムを実装できる。	3	前5
専門的能力	分野別の専門工学	情報系分野	プログラミング	代入や演算子の概念を理解し、式を記述できる。	3	前4
				プロシージャ(または、関数、サブルーチンなど)の概念を理解し、これらを含むプログラムを記述できる。	3	前1
				変数の概念を説明できる。	3	前4
				データ型の概念を説明できる。	3	前8
				制御構造の概念を理解し、条件分岐を記述できる。	3	前2
				制御構造の概念を理解し、反復処理を記述できる。	3	前5
				与えられた問題に対して、それを解決するためのソースプログラムを記述できる。	3	前4
				ソフトウェア生成に必要なツールを使い、ソースプログラムをロードモジュールに変換して実行できる。	3	前4
				与えられたソースプログラムを解析し、プログラムの動作を予測することができる。	3	前2
				ソフトウェア開発に利用する標準的なツールの種類と機能を説明できる。	1	前4
				要求仕様に従って、標準的な手法により実行効率を考慮したプログラムを設計できる。	3	
				要求仕様に従って、いずれかの手法により動作するプログラムを設計することができる。	2	
				要求仕様に従って、いずれかの手法により動作するプログラムを実装することができる。	2	
				要求仕様に従って、標準的な手法により実行効率を考慮したプログラムを実装できる。	1	
	分野別の工学実験・実習能力	情報系分野【実験・実習能力】	情報系【実験・実習】	整数・小数をコンピュータのメモリ上でデジタル表現する方法を説明できる。	3	
				基数が異なる数の間で相互に変換できる。	3	
				整数を2進数、10進数、16進数で表現できる。	4	
				小数を2進数、10進数、16進数で表現できる。	4	
				基本的な論理演算を行うことができる。	2	
				基本的な論理演算を組合わせて、論理関数を論理式として表現できる。	2	
				与えられた問題に対してそれを解決するためのソースプログラムを、標準的な開発ツールや開発環境を利用して記述できる。	3	
				ソフトウェア生成に利用される標準的なツールや環境を使い、ソースプログラムをロードモジュールに変換して実行できる。	3	
				ソフトウェア開発の現場において標準的とされるツールを使い、生成したロードモジュールの動作を確認できる。	3	
				フローチャートなどを用いて、作成するプログラムの設計図を作成することができる。	3	

				問題を解決するために、与えられたアルゴリズムを用いてソースプログラムを記述し、得られた実行結果を確認できる。	3	
				要求仕様に従って標準的な手法によりプログラムを設計し、適切な実行結果を得ることができる。	2	
評価割合						
		試験		レポート		合計
総合評価割合		80		20		100
プログラミングの知識とプログラム設計		80		0		80
プログラミングの実践的能力		0		20		20