

科目名	ソフトウェア演習		英文表記	Software Practice		2016/3/20	
科目コード	1205						
教員名:	神里 志穂子					作成	
技術職員名:	佐竹 卓彦						
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	1年	必	履修	1単位	実習	後期	
科目目標 【MCC目標】	Unixシステムの基本的な使い方とUnixを用いたプログラミングの演習を行う。 【V-D-1】①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。 【V-D-1】②主要な言語処理プロセッサの種類と特徴を説明できる。 【V-D-1】③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。						
総合評価	課題(100%)で評価する						
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(優)	標準的な到達レベル(良)	最低限必要な到達レベル(可)	セルフチェック
	40%	① 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	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か課題で評価する。課題(100%)で評価する。	授業で学習した内容と関連付けながら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するまでの流れを説明できる。	教科書や資料に従って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するまでの要点を説明できる。	教科書や資料を見ながら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する方法を説明できる。	
	30%	② 主要な言語処理プロセッサの種類と特徴を説明できる	主要な言語処理プロセッサの種類と特徴を説明できるか課題で評価する。課題(100%)で評価する。	授業で学習した内容と関連付けながら主要な言語処理プロセッサの種類と特徴を説明できる。	教科書や資料に従って主要な言語処理プロセッサの種類と特徴を説明できる。	教科書や資料を見ながら主要な言語処理プロセッサの種類と特徴を説明できる。	
30%	③ 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	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か課題で評価する。課題(100%)で評価する。	授業で学習した内容と関連付けながら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。	教科書や資料に従って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。	教科書や資料を見ながら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。		
本科・専攻科教育目標	1	2	3	4			
	○		◎				
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			0	0	100	100	
基礎的理解	①②				40	40	
応用力(実践・専門・融合)	③				10	10	
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				50	50	
授業概要、方針、履修上の注意	Unixの基本操作やプログラミングの実習を通して、サーバなどに利用されているUnixの基本利用技術をマスターし、マルチユーザの仕組みを体験することで、サーバとユーザの関係の基礎を学ぶ。アプリケーションを作成する環境構築、アプリケーションを作成しながらプログラムの基本を学ぶ。						
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、UNIX入門関連の図書。						

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 チェッ ク
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
期末	期末試験	[2]			
16	ガイダンス	2	講義の進め方や課題の提出方法を説明する。Unixの特徴や概念について学習。演習のための環境設定。	演習に沿った課題を課す	
17	ディレクトリ構造	2	ディレクトリ構造の理解。ファイルとディレクトリの管理コマンドの学習 (Unixコマンドのレポート課題)		
18	ファイル操作の確認	2	Vi, emacsの利用について学習する		
19	データ処理1	2	データの抽出、流れ処理に関して学習する		
20	データ処理2	2	データの抽出、組み合わせ処理に関して学習する		
21	Unixを用いたプログラミング	2	算術演算子の演習		
22	Unixを用いたプログラミング	2	制御文の演習		
23	Unixを用いたプログラミング	2	制御文の演習		
24	Unixを用いたプログラミング	2	配列の演習		
25	Unixを用いたプログラミング	2	多次元ループの演習		
26	Unixを用いたプログラミング	2	多次元配列の演習		
27	Unixを用いたプログラミング	2	関数の演習		
28	Unixを用いたプログラミング	2	関数の演習		
29	Unixを用いたプログラミング	2	関数と引数に関する演習		
30	Unixを用いたプログラミング	2	関数と引数に関する演習		
期末	期末試験	[2]			
学習時間合計		30	実時間	22.5	
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×14回	
②					
③					
<b>備考欄</b>					
<p>(JABEE関連共通記述)</p> <ul style="list-style-type: none"> <li>・この科目はJABEE非対応科目である。その他必要事項は各コースで定める。</li> </ul> <p>(各科目個別記述)</p> <ul style="list-style-type: none"> <li>・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)、プログラミング基礎II(2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。</li> </ul> <p>(モデルコアカリキュラム)</p> <ul style="list-style-type: none"> <li>・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。</li> </ul> <p>(航空技術者プログラム)</p> <ul style="list-style-type: none"> <li>・【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。</li> </ul>					