

香川高等専門学校		開講年度	平成28年度 (2016年度)	授業科目	システムプログラミング
科目基礎情報					
科目番号	0012		科目区分	専門 / 選択	
授業形態	授業		単位の種別と単位数	履修単位: 2	
開設学科	情報工学科 (2018年度以前入学者)		対象学年	4	
開設期	通年		週時間数	2	
教科書/教材	教科書: 羽山博 著 「Linuxシステムプログラミング」				
担当教員	篠山 学				
到達目標					
オペレーティングシステムの役割を理解し、それらの資源を利用する手段としてシステムコールを使用したプログラミングが行える。計算機内でのプロセスの状態遷移を把握でき、それを自由にコントロールできるようになる。					
ループリック					
		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	
評価項目1					
評価項目2					
評価項目3					
学科の到達目標項目との関係					
教育方法等					
概要	オペレーティングシステムの役割を理解し、それらの資源を利用する手段としてシステムコールを使用したプログラミングが行える。計算機内でのプロセスの状態遷移を把握でき、それを自由にコントロールできるようになる。				
授業の進め方・方法	各学習項目ごとに内容の解説を行い、関連する例題を説明した後、実際に実行結果を確認し理解させる。その後課題プログラムを作成し、レポートとして提出する。必要な関数の用法等はオンラインマニュアル等の参照によって自ら解決できるよう指導する。				
注意点					
授業計画					
		週	授業内容	週ごとの到達目標	
前期	1stQ	1週	システムプログラミングのためのC言語知識	低学年で学んだC言語の知識に加えて、構造体、ポインタ、リスト処理について復習し理解を深める。	
		2週	システムプログラミングのためのC言語知識	低学年で学んだC言語の知識に加えて、構造体、ポインタ、リスト処理について復習し理解を深める。	
		3週	システムプログラミングのためのC言語知識	低学年で学んだC言語の知識に加えて、構造体、ポインタ、リスト処理について復習し理解を深める。	
		4週	UNIXにおけるマルチプログラミングとプロセスの状態遷移	どのようにマルチプログラミングが実現されるかを理解する。プロセスの生成、プログラムの実行、他のプロセスとの同期を行うシステムコールを使った20行程度のプログラムはマニュアルを参照しながら作成できる。	
		5週	fork, wait, exec, exit (基本概念)	forkシステムコールにより子プロセスを生成できることを理解する。	
		6週	fork, wait, exec, exit (プログラミング)	execファミリによって外部プログラムを実行できることを理解する	
		7週	fork, wait, exec, exit (プログラミング)	waitシステムコールにより親プロセスと子プロセスを同期させられることを学ぶ。	
		8週	fork, wait, exec, exit (プログラミング)	プロセスの状態が主に4種類あることを理解し、ゾンビプロセスを作成するプログラムについて学ぶ。プロセステーブルが読み取れるようになる。	
	2ndQ	9週	[前期中間試験]		
		10週	試験問題の解答と授業評価アンケート ファイルシステム (基本概念)	UNIXファイルシステムのディスクの領域管理、ファイル管理の仕組みが理解できる。ファイルの管理情報を参照、変更を行うプログラムを作成できる。	
		11週	ファイルシステム (パーミッション)	ファイルのパーミッションについて理解し、パーミッションの表示、変更ができるようになる。Linuxのマスク値について理解する。	
		12週	ファイルシステム (プログラミング)	UNIXファイルシステムのディスクの領域管理、ファイル管理の仕組みが理解できる。ファイルの管理情報を参照、変更を行うプログラムを作成できる。	
		13週	ファイルシステム (プログラミング)	UNIXファイルシステムのディスクの領域管理、ファイル管理の仕組みが理解できる。ファイルの管理情報を参照、変更を行うプログラムを作成できる。	
		14週	簡易シェルの作成	UNIXのシェルの位置付けを理解し、簡易なシェルを作成できる。	
		15週	簡易シェルの作成	UNIXのシェルの位置付けを理解し、簡易なシェルを作成できる。	
		16週			
後期	3rdQ	1週	pipe機能	標準入力とパイプラインが理解できる。	
		2週	pipe機能	パイプラインの付け替えができるようになる。	
		3週	パイプ機能を持つシェルの作成	標準入力とパイプラインが理解でき、プロセス間通信が行えるプログラムが作成できる。	

4thQ	4週	パイプ機能を持つシェルの作成	標準入力とパイプラインが理解でき、プロセス間通信が行えるプログラムが作成できる。
	5週	パイプ機能を持つシェルの作成	標準入力とパイプラインが理解でき、プロセス間通信が行えるプログラムが作成できる。
	6週	シグナルの原理（基本概念）	シグナルを使った、プロセス間の同期の原理を理解するとともに、プログラムが作成できる
	7週	シグナルの原理（基本概念）	Ctrl+CやCtrl+Zなど何気なく使っていたショートカットキーが実はSIGINTやSIGTSTPなどのシグナルに対応していたことを学ぶ。
	8週	シグナルの原理（プログラミング）	さまざまなシグナルを使ったプログラムが作成できる
	9週	シグナルの原理（プログラミング）	シグナルを受信するだけでなく、無視することも必要になるケースを理解する。
	10週	シグナルの原理（プログラミング）	アラームシグナルを理解するとともに、プログラムを作成できる。
	11週	[後期中間試験]	
	12週	試験問題の解答と共有メモリによるIPC(基本概念)	共有メモリおよびメッセージを用いた複数プロセス間の通信の仕組みを理解し、プログラムが作成できる。
	13週	試験問題の解答と共有メモリによるIPC(基本概念)	共有メモリおよびメッセージを用いた複数プロセス間の通信の仕組みを理解し、プログラムが作成できる。
	14週	試験問題の解答と共有メモリによるIPC(基本概念)	共有メモリおよびメッセージを用いた複数プロセス間の通信の仕組みを理解し、プログラムが作成できる。
	15週	メッセージによるIPC(基本概念)	共有メモリおよびメッセージを用いた複数プロセス間の通信の仕組みを理解し、プログラムが作成できる。
	16週	メッセージによるIPC(基本概念)	共有メモリおよびメッセージを用いた複数プロセス間の通信の仕組みを理解し、プログラムが作成できる。

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
評価割合					
		試験	課題提出	合計	
総合評価割合		80	20	100	
基礎的能力		40	10	50	
専門的能力		40	10	50	