

科目名	システムプログラミング System Programming			担当教員	篠山 学		
学 年	4 年	学 期	通年	履修条件	選択	単位数	2
分 野	専門	授業形式	講義・演習	科目番号	10I04_31010	単位区別	履修
学習目標	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。UNIX が実装しているシステムコールは授業で扱うもの以外にも存在する。必要に応じてオンラインマニュアルや Web を参照して、必要なシステムコールを探し、その使用方法を理解しプログラムに組み等、自ら問題解決する能力を身につける。						
進め方	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。						
学習内容	学習項目（時間数）			学習到達目標			
	1. システムプログラミングのための C 言語（2） 2. システムプログラミングのための C 言語（2） 3. システムプログラミングのための C 言語（2） 4. 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（2） 5. .fork, wait, exec, exit（基本概念）（2） 6. fork, wait, exec, exit（プログラミング）（2） 7. fork, wait, exec, exit（プログラミング）（4）			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。 D2:1, 2 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。 D2:2, 3			
	[前期中間試験]（2）						
	8. 試験問題の解答（2） 9. ファイルシステム（基本概念）（2） 10. ファイルシステム（プログラミング）（2） 11. ファイルシステム（プログラミング）（2） 12. ファイルシステム（プログラミング）（2） 13. 簡易シェルの作成（2） 14. 簡易シェルの作成（2）			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。 D2:2, 3 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。 D2:1-3			
	前期末試験						
	15. 試験問題の解答（2） 16. pipe 機能（2） 17. pipe 機能（2） 18. パイプ機能を持つシェルの作成（2） 19. パイプ機能を持つシェルの作成（2） 20. パイプラインを用いたプロセス間通信（2） 21. パイプラインを用いたプロセス間通信（2） 22. パイプラインを用いたプロセス間通信（2）			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。 D2:1-3			
	[後期中間試験]（2）						
	23. 試験問題の解答（2） 24. シグナルの原理（基本概念）（2） 25. シグナルの原理（基本概念）（2） 26. シグナルの原理（プログラミング）（2） 27. シグナルの原理（プログラミング）（2） 28. 共有メモリによる IPC(基本概念)（2） 29. 共有メモリによる IPC(基本概念)（2）			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。 D2:2, E2:2 共有メモリ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。 D2:1, 2, E2:2			
	後期末試験						
	30. 試験問題の解答（2）						
評価方法	定期試験 80%，レポート 20%の比率で総合評価する。						
履修要件	情報処理Ⅱ（2年），ソフトウェア設計論（3年）						
関連科目	情報処理Ⅱ（2年），基礎情報工学（3年），計算機システム（4年）						
教 材	教科書：羽山博 著 「Linux システムプログラミング」						
備 考	特になし。						