

OSSモデルカリキュラムコースウェア概要

	スキル名	概要	受講対象者、受講前提	1	2	3	4	5	6	7
基礎	1 OSSの概要に関する知識	OSSの歴史と理念、代表的なOSS、標準化動向、利用分野と市場動向、OSSプロジェクトの発達と運営、OSSコミュニティへの参加、主なコミュニティ	入門カリキュラムのため、特に規定しない。基礎的なコンピュータ科学(ITSSレベル1程度)を習得、経験しているレベルの知識を有すること。	＜オープンソースの理念＞ オープンソースという言葉の定義と概念を理解する。あわせてオープンソースの基本的なライセンスを理解する。	＜オープンソースOSの歴史＞ オープンソースのOSの発展の歴史をUNIX、BSDとLinuxを軸に理解する。	＜代表的なオープンソース＞ 代表的なオープンソース製品の特徴を理解する。OS(オペレーティングシステム)とサーバ製品を中心に解説する。	＜代表的なオープンソース開発言語＞ 代表的なオープンソースの開発言語の特徴を理解する。開発言語とフレームワーク、統合開発環境を中心に解説する。	＜代表的なオープンソースアプリケーション＞ 代表的なオープンソースアプリケーションをデスクトップ用、サーバサイド用のカテゴリ毎に特徴を理解する。	＜オープンソースの市場動向＞ オープンソースの標準化動向とビジネスでの利用状況を理解する。	＜オープンソースソフトウェアを用いたシステム事例＞ Webシステムのインフラ構築を例に、実際のシステム構築におけるオープンソースの採用時のメリットや注意点を理解する。
	2 法務分野に関する基礎知識	OSS関連のライセンス等、法務面での基礎知識	本カリキュラムの「OSSの概要に関する知識」を受講済みであること。または、同等の知識を有すること。	＜オープンソース・ライセンスの概要＞ オープンソースのライセンスの法的な位置づけを理解する。また、代表的なライセンスについて分類を行い、それぞれの特徴を理解する。	＜代表的なオープンソース・ライセンスの解説[GPL類型]＞ オープンソースの登場とともに発展してきたライセンスで、世界のオープンソースプロジェクトで最も多く採用されていると言われているGPLのポイントを理解する。	＜代表的なオープンソース・ライセンスの解説[GPL類型]＞ オープンソースの登場とともに発展してきたライセンスで、世界のオープンソースプロジェクトで最も多く採用されていると言われているGPLを理解する。	＜代表的なオープンソース・ライセンスの解説[BSD類型]＞ GPLやLGPLと比較して制限事項が緩いとされているBSD Licenseと、類型でApache Software Foundationを中心に利用されているApache Software Licenseを理解する。	＜知的財産の概要(1)＞ 知的財産権のうち、著作権と特許権についてソフトウェア産業との関わりについて理解する。	＜知的財産の概要(2)＞ 知的財産権のうち、商標権と実用新案権、意匠権についてソフトウェア産業との関わりについて理解する。	＜OSS利用上の知的財産面での考慮点＞ OSS利用上の注意点を、知的財産面から理解する。
	3 コンピュータシステムやアーキテクチャに関するスキル	CPU、バス、DMA、I/O、PO SIX、スレッド等	入門カリキュラムのため、特に規定しない。基礎的なコンピュータ科学(ITSSレベル1程度)を習得、経験しているレベルの知識を有すること。	＜コンピュータアーキテクチャの基本＞ コンピュータアーキテクチャの意味と発展の歴史、必要性を理解する。ソフトウェアやハードウェアのアーキテクチャとは何を規定するものか、その特徴と役割を理解する。	＜コンピュータハードウェアの基本＞ コンピュータハードウェアの基本構成と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。	＜CPUアーキテクチャの基本＞ CPUの基本構成と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。	＜ディスクと周辺機器の基本＞ ディスクと周辺機器の基本構成と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。	＜インタフェース技術の基本＞ インタフェースの基本構成と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。	＜ソフトウェアアーキテクチャ＞ ソフトウェアの種類と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。	＜OSのアーキテクチャ＞ OSの基本構成と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。
	4 分散アーキテクチャに関するスキル	統計確率論、トラフィック理論、RIP/OSPF、CORBA等	【レベル1】 Webやメールなどの基本的なインターネットアプリケーションの利用経験があること ファイル、ディレクトリ、プロセスなどのOSの基本概念を理解していること 確率に関する初歩的な知識があること 【レベル2】 複数のプログラミング言語を使った経験があること	＜分散アーキテクチャとは＞ コンピュータとネットワークの発展史を概観し、分散情報システムが登場した理由を理解する。そして、分散アーキテクチャのメリット、分散情報システムに求められるもの、実際の分散アプリケーションで採用されているアーキテクチャの事例を学ぶ。安価な機器が大量にネットワーク接続される時代に適した情報システムの構成法としての分散アーキテクチャの意義を理解する。	＜名前管理＞ 分散情報システムでは、資源にアクセスするために論理的な名前を用いることが多い。この名前という考え方の重要性、名前を介したアクセスが一般のインターネットアプリケーションでも広く使われていること、名前と密接に関連した透過性という概念と分散アーキテクチャの関係について理解する。最後に、LDAP、DNSなどの実際に使われている分散名前管理システムの大まかな仕組みを理解し、コマンドを使ってこれらにアクセスする方法を学ぶ。	＜複製管理＞ 分散情報システムが管理するデータの可用性とアクセス性能を向上させる技術としての複製管理の重要性を理解し、その概要を学ぶ。さらに、複製の存在がシステムの性能に与える影響を推定するために重要となるキャッシュのヒット率を概算する方法を学ぶ。	＜耐故障性＞ 情報システムが社会基盤化し、組織の基幹部門での利用も進んだ結果、システム障害が大きな被害を招く事例が増えしてきた。一方で、ハードウェアの故障やソフトウェアのバグを完全になくすことは困難である。不安定な機器やソフトウェアから構成される情報システムを安定して稼働させることの重要性を理解し、そのための基礎となる技法を学ぶ。そして、分散情報システムでは、冗長性を持たせた機器構成が一般的であり、うまく設計、運用すれば耐故障性や可用性を高めやすいことを理解する。最後に故障確率を数学的に議論する方法の概要を学ぶ。	＜セキュアチャネル＞ 分散アプリケーションの中には、インターネットのようなセキュアとは限らないネットワークを利用するものが少なくない。このようなネットワーク上にセキュアな通信路を論理的に構成する方式の重要性を学び、暗号技術を利用することでそれが実現可能であることを理解する。また、セキュアな通信路の事例としてSSL/TLSの基本的な仕組みを学ぶ。	＜CORBA入門(1)＞ 分散アプリケーション構築基盤としてのCORBAの概要を理解し、CORBAによるアプリケーション開発の手順を学ぶ。同時に、遠隔呼び出しに必要なコードの大部分が自動生成可能なことを学ぶ。	＜CORBA入門(2)＞ CORBAが提供するさまざまな機能を概観するとともに、非同期メソッド呼び出し等を含むアプリケーション開発の手順を学ぶ。
5 Linuxの概念や基本操作に関するスキル	概念整理、基本操作	コンピュータシステムやアーキテクチャに関するスキル、OSSの概要に関する基礎知識を受講済みであることが望ましい。また、基礎的なコンピュータ科学、セキュリティ工学基礎(ITSSレベル1程度)を習得、経験を持つレベルの知識を有すること。	＜Linux概要＞ OSの役割や歴史、その中でUNIXの基本的な概念、Linuxとオープンソースの関係などを理解する	＜ファイル操作＞ UNIX/Linuxの基本操作およびファイル操作について学ぶ。なお、Linuxでの多くのディストリビューションにおいて、Bシェル系がデフォルトとなっているため、キー操作等ではBシェル(実際にはbashだが)を中心に教えるものとする(別の回でも同様)。	＜ユーザの権限と管理＞ UNIX/Linuxにおけるユーザ権限についての概念を理解し、操作方法を学ぶ。	＜システム管理＞ UNIX/Linuxにおけるシステムの管理技法について学習する。このカテゴリで1つのものとするのが望ましい。				
6 Linuxのカーネルに関するスキル	カーネル構造、プロセス、スレッド、スケジューラ等	基礎的なコンピュータ科学、セキュリティ工学基礎(ITSSレベル1程度)を習得、経験を持つレベルの知識を有すること。	＜Linuxカーネル概論＞ OSがコンピュータに必要とされるに至った経緯を説明する。また、商用OSとは別の流れの中でUNIXが登場し、やがてLinuxが開発され、今日に至っている背景とともに、Linuxの構造と基本的な機能を学ぶ。	＜スケジューリング＞ 「プロセス」の概念とプロセスが切り替わるメカニズムの概要を理解する。併せて、スケジューリングのアルゴリズムを学ぶ。	＜割り込みと遅延＞ さまざまな割り込みについて学ぶ。OSそのものは、アプリケーションプログラムやハードウェアに対して能動的には振る舞わない。それらの位置づけを理解すると「呼び出される」。そのきっかけが「割り込み」である。これらについて学習する。	＜システムコール＞ OSの機能を、一般のアプリケーションプログラムが利用する際に使うのがシステムコールである。それらの位置づけを理解するとともに、どのように実装されているかを学ぶ。	＜プロセス管理＞ プロセスが生成されてから終了して消滅するまでのライフサイクルについて理解する。また、そのサイクルにカーネルがどのように関わっているかを学ぶ。	＜メモリ管理(1)＞ 仮想メモリと実メモリの管理方法を3回にわたって学ぶ。第1回目は、ハードウェアのメモリアドレスと仮想アドレスの変換機構について学ぶ。	＜メモリ管理(2)＞ 物理的に装備されている実メモリの管理方法を学ぶ。また、仮想記憶方式の基本である「ページ」の管理方法を学ぶ。	
7 Linuxのシステム管理に関するスキル	インストール、カーネル設定、ポート設定、ネットワーク設定、パッケージ管理、ユーザ管理、ファイル管理、サービス管理、デバイス管理、ログ管理、バックアップ等	本カリキュラムの「Linuxの概念や基本操作」を受講済みであること。または、同等の知識を有すること。	＜Linuxシステム管理の作業概要＞ OSSの基本概念とカテゴリ、その発展の歴史、必然性、メリット、最新動向などを概説する。	＜Linuxシステム管理・サーバ管理＞ Linuxのインストール、管理者権限の設定、アプリケーションパッケージの導入などの方法とその作業内容を実習で理解する。	＜Linuxシステム管理・ファイル/ディスク管理＞ Linuxのインストール、管理者権限の設定、アプリケーションパッケージの導入などの方法とその作業内容を実習で理解する。	＜Linuxシステム管理・ユーザ管理＞ Linuxにおけるユーザ登録・管理運用の設定の方法とその作業内容を実習で理解する。	＜Linuxシステム管理・バックアップとログ運用管理＞ Linuxにおけるデータやアプリケーションリソースのバックアップ運用管理の方法と作業内容を実習することにより理解する。	＜Linuxシステム管理・リソース管理＞ Linuxのハードウェアリソースの管理や利用制限などの方法とその作業内容を実習で理解する。	＜Linuxシステム管理・カーネルの管理＞ Linuxのカーネルをシステム管理の観点から学習し、アップデートなどの運用管理の方法とその作業内容を実習で理解する。	

8	9	10	11	12	13	14	15	スキル名	
<p><オープンソースソフトウェアコミュニティ> オープンソースコミュニティの成り立ちと運営方法を理解する。オープンソースコミュニティへの貢献や参加方法を理解する。</p>	<p><オープンソースソフトウェアビジネス> オープンソースソフトウェアに関連したサポートサービスやコンサルティングなどのビジネスモデルとそのトレンドを理解する。</p>	<p><オープンソースの技術情報獲得方法> オープンソースの技術情報の調査方法を理解する。代表的なオープンソースコミュニティや企業が提供するオープンソース情報および各種の技術情報Webサイトを紹介します。実際に技術情報の調査の演習を行う。</p>	<p><オープンソースのOSの導入と動作確認> 複数のオープンソースOSを演習用PCに導入して、導入の手順やOSによる基本的な違いなどを理解する。</p>	<p><オープンソースのサーバ製品の導入と動作確認> 代表的なオープンソースのサーバ製品を演習用PCに導入して、導入の手順や製品による基本的な違いなどを理解する。</p>	<p><オープンソースのデスクトップ用アプリケーションの導入と動作確認> 代表的なオープンソースのデスクトップ用アプリケーションを演習用PCに導入して、導入の手順や製品による基本的な違いなどを理解する。</p>	<p><オープンソースのサーバサイドアプリケーションの導入と動作確認> 代表的なオープンソースのサーバサイドアプリケーションを演習用PCに導入して、導入の手順や製品による基本的な違いなどを理解する。</p>	<p><オープンソースの仮想化ツールの導入と動作確認> 代表的なオープンソースの仮想化ツールと、その上で動作するOSを演習用PCに導入して、導入の手順や製品による基本的な違いなどを理解する。</p>	OSSの概要に関する知識	1
講義90分	講義90分	講義+ワークショップ90分	ワークショップ90分	ワークショップ90分	ワークショップ90分	ワークショップ90分	ワークショップ90分		
<p><OSS利用上の知的財産以外での法務面での考慮点> OSS利用上の注意点を、知的財産以外の法務面から理解する。</p>	<p><企業/団体等における法的リスク管理> 企業や団体がどのようにOSSに対する法的リスクを管理し、OSS開発者や自社のビジネス、ユーザを保護しているか理解する。</p>	<p><法的リスクに対応したビジネス> OSSの法的リスクを新たなビジネスチャンスとして活動している企業のビジネスモデルを理解する。</p>	<p><OSS開発コミュニティが検討すべき法的リスク低減策> OSSの法的リスクを低減するため、OSS開発コミュニティが実施を検討すべき方策を理解する。</p>	<p><OSSビジネス関連企業が検討すべき法的リスク低減策> OSSの法的リスクを低減するため、OSSビジネス関連企業が実施を検討すべき方策を理解する。</p>	<p><OSSの知的財産問題に関する訴訟/トラブル事例> OSSの知的財産侵害に関して発生した訴訟やトラブルの事例と経緯から、OSS利用上の法的リスクを理解する。</p>	<p><ソフトウェア特許論争> OSSのみではなく、ソフトウェア産業全体の問題として取り上げられるソフトウェア特許の論争を、各国のソフトウェア特許に対する方針と併せて理解する。</p>	<p><知的財産関連の出願のガイドライン> 各種知的財産について、ソフトウェアを対象とした場合の例として出願方法の概要を理解する。演習として架空の新発明/新商品を出願するシミュレーションを実施する。</p>	法務分野に関する基礎知識	2
講義90分	講義90分	講義90分	講義90分	講義90分	講義90分	講義90分	講義+ワークショップ90分		
<p><ミドルウェアの種類と特徴> ミドルウェアの基本構成と役割、特徴、発展の歴史、オープンソースとのかかわりの概要を理解する。</p>	<p><コンピュータシステムの構成> オープンソースを用いたコンピュータシステムの基本構成と役割、特徴、発展の歴史とオープンソースとのかかわりの概要を理解する。</p>	<p><システムアーキテクチャの活用事例> OSSの活用のかたを、Webシステムを中心に事例を使って理解する。</p>	<p><Webシステムのアーキテクチャ> 事例をもとに、OSSだけで構成されたWeb 3層アプリケーションを構築し、それぞれの導入方法や機能を理解する。</p>	<p><OSSを活用した基盤設計ケースワーク> OSS、ハードウェア、ミドルウェアなどを組み合わせてシステム基盤を構築するケースワークを実施する。</p>	<p><オープンソースシステムアーキテクチャ構築> 第12回で設計したシステム基盤を構築する。ビジネスロジックや非機能要件を実際にコンピュータに導入し、構築する。</p>	<p><OSSの動作環境としてのハードウェア> OSSの基盤となるべきハードウェアの発展の歴史、必然性、メリット、最新動向などを理解する。</p>	<p><これからのオープンソースアーキテクチャの動向> これからのオープンソースアーキテクチャの種類とその内容、特徴、発展の動向を理解する。</p>	コンピュータシステムやアーキテクチャに関するスキル	3
講義90分	講義90分	講義90分	(ワークショップ 90分)	(ワークショップ 90分)	(ワークショップ 180分)	(講義 90分)	(講義 90分)		基礎
<p><Webサービス入門(1)> サービスとしてのアプリケーションという考え方やWebサービスが登場した背景を学ぶ。また、既存のWebサービスを利用するクライアント側プログラムの開発手順を学ぶ。</p>	<p><Webサービス入門(2)> Webサービスに関するさまざまな規格を概観するとともに、サーバ側アプリケーション開発の手順を学ぶ。</p>	<p><マッシュアップ入門> 大手ポータルサイトをはじめとして、WebアプリケーションのAPIを公開する事例が増えている。これらAPIを活用することで、比較的簡単に高機能なアプリケーションを実現できることを学ぶ。ワークショップでは、JavaScriptを用いてWeb APIを操作する演習を行う。</p>	<p><分散トランザクション> 機器の故障から重要なデータを守り、さらに並行制御の難しさを緩和するために利用されるトランザクション技術の概要を理解し、その分散版である分散トランザクションの要素技術を学ぶ。また、長時間トランザクションの必要性とそれを実現するための妥協策を理解する。</p>	<p><ピアツーピアアーキテクチャ> ピアツーピア(P2P)は、情報漏洩と関連づけて論じられることが多く、悪いイメージが先行した感がある。一方、ピアツーピア型の分散アーキテクチャをうまく利用することで、スケーラビリティと耐故障性を高いレベルで実現したシステムも少なくない。ピアツーピア型アーキテクチャの特徴を理解し、実際の活用事例について学ぶ。</p>	<p><モビリティ> 今日の情報システムは、携帯電話、PDA、ノートPCなどの移動する機器を構成要素として含むことが多くなった。一方、Javaアプレット、JavaScript、Active Xコントロールのようにソフトウェアの移動も頻繁に発生している。機器やソフトウェアの移動により享受できる利便性と解決すべき問題について理解するとともに、円滑な移動を支援するために開発された要素技術について学ぶ。</p>	<p><一貫性> 大規模な分散アプリケーションでは、共有データへの書き込みが性能上のボトルネックになりやすい。スケーラビリティを向上させるためには緩やかな一貫性という考え方が必要であることを理解し、その適用事例を学ぶ。そして、データの一致性と深い関係にある分散システムの問題モデルを学び、メモリの一貫性を表現するモデルを学ぶ。</p>	<p><大規模な分散システムの事例> 今日の分散情報システムには、小さなLAN内に限定されるものから全世界規模のものまでがあり、システムのスケールによって設計方針やアーキテクチャの妥当性に対する考え方も異なってくる。代表的な大規模システムの事例について学び、設計方針の妥当性に関連した議論を行う。</p>	分散アーキテクチャに関するスキル	4
講義+ワークショップ 120分	講義+ワークショップ 120分	講義+ワークショップ 180分	講義 90分	講義 90分	講義 90分	講義 90分	講義+ワークショップ 120分		
<p>ゴリはボリュームがあるため、3</p>	<p><ファイルシステム> UNIX/Linuxにおけるファイルシステムとパーティショニングの考え方を学ぶ。</p>		<p><データ保全とバックアップ> サーバとしての利用を考慮する上では必須ともなる、データのバックアップの方法について考えさせる。</p>		<p><シェルスクリプトと開発環境> UNIXの制御に必要になるシェルとシェルスクリプトの概要を学び、実際にスクリプトを構築してみる。また、C言語の開発についても最低限の部分を学ぶ。</p>		<p><ネットワークの基本> UNIX/Linuxにおけるネットワークの基本を再確認し、Linuxにおける設定について学習する。ネットワークはUNIX/Linuxにおいては必須ではあるが、別の科目にて、より専門的に習得することが望ましいため、ここではあくまで基本の部分として「利用できる環境にする」ということを主眼とする。</p>	Linuxの概念や基本操作に関するスキル	5
講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義90分		
<p><メモリ管理(3)> 仮想アドレス空間(プロセス空間)と実ページの対応方法を通してプロセス空間の管理方法を学ぶ。また、実ページが不足してきた場合の処理方法を学ぶ。</p>	<p><ファイル管理(1):仮想ファイルシステム> 仮想ファイルシステムについて学ぶ。</p>	<p><ファイル管理(2):ファイルの操作> ここでは一般的なファイル操作を学ぶ。はじめに全体像の概略を示した後、個々の処理方式について解説する。</p>	<p><ファイル管理(3):特殊ファイル> Linuxが持つ特殊ファイルについて学ぶ。</p>	<p><ネットワーク(1):ソケットインタフェース> ネットワークについて4回にわたって学ぶ。第1回目は、プロトコルスタックの階層構造について学んだあとに、Linuxのソケットについて学ぶ。</p>	<p><ネットワーク(2):IPとUDP> ネットワークの2回目ではインターネットプロトコル(IP)について学ぶ。</p>	<p><ネットワーク(3):UDPとTCP> UDPと、より信頼できる通信の仕組みのためのTCPについて学ぶ。</p>	<p><ネットワーク(4):TCPフロー制御と輻輳制御> TCPのフロー制御と輻輳制御について学ぶ。</p>	Linuxのカーネルに関するスキル	6
講義90分	講義90分	講義90分	講義90分	講義90分	講義90分	講義90分	講義90分		
<p><Linuxシステム管理・ネットワーク管理> Linuxにおけるネットワークのインストール、設定などの方法とその作業内容を実習することにより理解する。</p>	<p><Linuxシステム管理・ルーティング管理> Linuxにおけるネットワークルーティングの設定、検証などの方法とその作業内容を実習することにより理解する。</p>	<p><Linuxシステム管理・DHCPの構築と運用> DHCPのインストール、設定、運用などの方法とその作業内容を実習で理解する。</p>	<p><Linuxシステム管理・FTPの構築と運用> FTPのインストール、設定、運用などの方法とその作業内容を実習で理解する。</p>	<p><Linuxシステム管理・NFSの構築と運用> LinuxにおけるNFSの設定、運用などの方法とその作業内容を実習で理解する。</p>	<p><Linuxシステム管理・Sambaの構築と運用> LinuxとWindowsを共用するためのSambaのインストール、権限の設定、導入などの方法とその作業内容を実習で理解する。</p>	<p><Linuxシステム管理・基本運用作業のトラブルシューティング> Linuxのトラブルのうち、サーバ本体の問題について、その種類と対処方法を実習で理解する。</p>	<p><Linuxシステム管理・ネットワークのトラブルシューティング> Linuxのトラブルのうち、ネットワーク関係の問題についてその種類と対処方法を実習で理解する。</p>	Linuxのシステム管理に関するスキル	7
講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分		システム

OSSモデルカリキュラムコースウェア概要

黄色はレベル1(基本レベル)に、青色はレベル2(応用レベル)に対応しています。

	スキル名	概要	受講対象者、受講前提	1	2	3	4	5	6	7	
システム	8	Linuxのシステムプログラミングに関するスキル	Shellプログラミング、スレッド、ファイル入出力プログラミング、ネットワークプログラミング、共有メモリ、セマフォ、キュー、問題判別等	本カリキュラムの「Linuxのシステム管理」を受講済みであること。または、同等の知識を有すること。	<p><ログイン手順とコンパイル手順></p> <p>ユーザ名とパスワードを認証後にUNIXシステムにログインできる。パスワード、PAM、LDAP認証について解説する。ログインして簡単なプログラムを作成し、コンパイルして実行する。よく利用するエディタの操作やコマンド入力を練習する。</p>	<p><shellプログラミング></p> <p>エディタを使って#で始まるコマンドを記述したスクリプトファイルを作成する。ファイルに実行権を与えて、/ファイル名で実行して確認する。スクリプトファイルを作成することで、定型的な作業を簡単に実行できることを体験する。簡単なサーバ起動スクリプト等を例にあげて処理手順を理解する。</p>	<p><ファイル入出力プログラミング></p> <p>標準入出力ライブラリだけでなく、低水準ファイルアクセスをするシステムコールを使って、ファイルを作成し、ファイルからデータを入力し、ファイルを開けるプログラムを作成する。オーナー、グループ、その他に区分したファイルアクセス権限を理解する。</p>	<p><ファイルシステム></p> <p>大容量のディスクを複数のパーティションに分割して独立したディスクとして扱うことができる。パーティションは1つのブートブロックと複数のブロックにわけられる。分割したブロックをmountすることで1つのルートディレクトリのツリーに付加できることを学習する。</p>	<p><UNIX環境></p> <p>コマンドラインからmain()関数に渡す引数の処理手順を学習する。また環境変数の取得・設定方法を学習する。さらに共有ライブラリを使用する・しないによって、実行形式ファイルサイズの大きさや利点・欠点を理解する。</p>	<p><ライブラリの利用方法と作成手順></p> <p>開発された有益なオブジェクトモジュールをライブラリに登録しておくことで、同じ開発を重複する必要がなくなる。staticとsharedの2種類のライブラリの利用方法と登録方法について理解する。また、SQLデータベースや暗号化ライブラリを利用したシステムプログラムのコンパイル手順を理解する。</p>	<p><データの管理></p> <p>メモリは他のプロセスや不正プログラムによって書き換えられないようにOSが管理している。ロックを掛けることで、複数のプロセスが同じファイルを同時に利用しないようにする。また、代表的なBerkeleyデータベースを処理するプログラム例を理解する。</p>
	9	ネットワークサーバ管理に関するスキル	WEBアプリケーションサーバ、ネットワーク基盤、ファイルサーバ、トラブルシューティング等	本カリキュラムの「Linuxの概念や基本操作」を受講済みであること。または、同等の知識を有すること。	<p><ネットワークサーバの機能と特徴></p> <p>ネットワークサーバの役割、機能、特徴などを理解する。</p>	<p><サーバシステムの導入></p> <p>ネットワークサーバ導入の作業概要とその手順、注意点を理解する。</p>	<p><ネームサーバの導入></p> <p>DNSサーバ導入の作業手順、設定内容を理解する。</p>	<p><Webサーバの導入></p> <p>Webサーバ導入の作業手順、設定内容を理解する。</p>	<p><メールサーバ導入の内容と作業手順></p> <p>メールサーバ導入の作業手順、設定内容を理解する。</p>	<p><スーパーサーバの導入></p> <p>スーパーサーバの役割、メリット、構成及び導入の作業手順、設定内容を理解する。</p>	<p><プロキシサーバの導入></p> <p>プロキシサーバの導入と設定の内容、手順を理解する。</p>
システム	10	クラスタシステム構築に関するスキル	HPCやEnterprise SystemのFailSafe用等	HAクラスタ、HPCクラスタの構築、設計を行うSE。Linuxのインストール、設定が行える。C、Fortranでのプログラミング経験者。	<p><クラスタシステム概論、HAクラスタ(1)></p> <p>HAクラスタ、HPCクラスタの概要を理解する。HAクラスタの概念およびその基本技術に関する理解を深める。</p>	<p><HAクラスタ(2)></p> <p>HAクラスタ構築を実際に行い、その構築手順を理解する。</p>	<p><HAクラスタ(3)></p> <p>HAクラスタ構築を実際に行い、その構築手順を理解する。</p>	<p><コンピュータシミュレーション></p> <p>コンピュータシミュレーションの必要性および現在のHPCの動向に関して理解する。</p>	<p><並列プログラミング概論></p> <p>並列プログラミングの概要を理解する。</p>	<p><並列プログラミング 実践(1) マルチスレッドプログラミング></p> <p>マルチスレッドプログラミングの基礎を習得する。</p>	
	11	ネットワーク・アーキテクチャに関するスキル	TCP/IP概論	基礎的なコンピュータ科学、セキュリティ工学基礎(ITSSレベル1程度)を習得、経験しているレベルの知識を有すること。	<p><オープンネットワークの概念と仕組み></p> <p>基本的なネットワークの構成及び伝送技術について大規模化、マルチプロトコル化を中心に技術の発展と企業インフラへの適用を理解する。その基本となっているOSI 7階層モデルについて理解する。</p>	<p><通信の形態とプロトコル></p> <p>通信の形態と通信プロトコルの種類、その位置づけと特徴、ネットワークシステムにおける意義について理解する。</p>	<p><インターネット通信の仕組み></p> <p>通信の形態と通信プロトコルの種類、その位置づけと特徴、ネットワークシステムにおける意義について理解する。</p>	<p><LANネットワークの仕組み></p> <p>ネットワーク通信におけるデータリンク層のプロトコルとしてのLAN通信の仕組みと特徴について理解する。</p>	<p><無線ネットワークの種類と通信の仕組み></p> <p>無線通信の仕組みと特徴について理解する。IEEE802.11、802.15のプロトコル仕様を理解する。</p>	<p><オープンネットワークの通信仕様></p> <p>インターネットの通信はどのように連携し、動作するのか、その機能分担と連携の仕様、ネットワークシステムの特長について理解する。</p>	<p><IPネットワークの仕組み></p> <p>IP通信の形態と通信プロトコルの種類、その位置づけと特徴、ネットワークシステムにおける意義について理解する。</p>
ネットワーク	12	ネットワーク管理に関するスキル	コンピュータネットワーク構築・運用	業務担当としてネットワーク運営に関わった経験があること、または担当システムの設計要件が理解できていること。	<p><ネットワークシステム運用の概要></p> <p>ネットワーク運用管理の全体像と各運用管理作業の概要を理解する。ネットワーク管理作業の重要性や注意点を理解する。</p>	<p><ネットワーク管理の個別項目とその内容></p> <p>ネットワーク管理の個別作業の目的、内容や注意点を理解する。</p>	<p><ネットワークのキャパシティ管理の個別項目とその内容></p> <p>ネットワーク管理の個別作業の中でもっとも複雑で重要なキャパシティ管理の目的、内容や注意点を理解する。</p>	<p><ネットワークの性能管理の個別項目とその内容></p> <p>ネットワーク管理の個別作業の中で重要な性能管理の目的、内容や注意点を理解する。</p>	<p><TCP/IPの管理></p> <p>ネットワーク管理対象の中でもっとも重要なTCP/IPの管理について、その目的、内容や注意点を理解する。</p>	<p><ネットワークサーバの運用管理実践></p> <p>ネットワークサーバ運用管理の目的、内容、仕様と特徴を理解する。</p>	<p><ネットワークハードウェアの運用管理></p> <p>ネットワークのハードウェア(ルータ、スイッチ等)の運用管理の項目や情報収集の方法、注意点を理解する。</p>
	13	Javaに関するスキル	Applet, Servlet, JSP, EJB等	入門カリキュラムのため、特に規定しないが、以下の受講条件を満たすものとする。 ・プログラミング経験があること	<p><Javaの基本></p> <p>Javaのオープンソースとしての基本概念と特長な機能、その発展の歴史、必然性、メリット、最新動向などを理解する。</p>	<p><Java言語の基本構造></p> <p>Java言語の基本的な仕組み、特徴、プログラミングの内容を理解する。サンプルプログラムをもとに、実際のプログラミングを行い、内容を確認する。</p>	<p><オブジェクト指向プログラミングのメリット></p> <p>Javaにおけるオブジェクト指向プログラミングの内容とそのメリット、具体的なプログラミング方法を理解する。</p>	<p><Javaによるアプリケーション開発手順></p> <p>Javaによるアプリケーション開発の手順と注意点、品質作りこみの方法、標準化などを理解する。また、統合開発環境としてのOSSであるEclipseについてその使用方法、特徴と利点を理解する。</p>	<p><Javaによるネットワークプログラミング></p> <p>Javaによるネットワークプログラミングの内容、特徴、ソケットによるセッション管理の仕様について理解する。</p>	<p><Servlet/JSP/JDBCによるWebアプリケーション開発の概要></p> <p>Java/Servletによるアプリケーションの開発の手順とその作業内容を理解する。Webアプリケーションの実現形態とそのメリットを理解する。</p>	<p><JDBCによるデータベースアクセス></p> <p>JavaアプリケーションとデータベースのJDBCによる接続の方法とその具体的な設定、プログラミングを理解する。</p>
プログラミング	14	C、C++に関するスキル	POSIX termio, curses, gtk++, Qt等を含む	入門カリキュラムのため、特に規定しないが、以下の受講条件を満たすものとする。 ・プログラミング経験があること	<p><Cの基本></p> <p>Cプログラミングの概要として、歴史や特徴、適用範囲について学習する。また、プログラムを開発する流れについて習得する。</p>	<p><Cの基本構造></p> <p>Cの基本的な仕組みと構成、基本構文を学習し、プログラミングの内容を理解する。また、サンプルプログラムをもとに、Cで作成したプログラムの流れを確認する。</p>	<p><文字列操作></p> <p>配列の扱いを学習し、Cによる文字列に対する処理方法について習得する。また、文字列操作の関数を使用し、より簡潔なプログラム作成法について学習する。</p>	<p><関数></p> <p>ユーザ定義関数の扱いを学習し、Cプログラムの部品化の方法を習得する。また、前節で学習した文字列操作関数以外の、標準関数を利用したプログラミングについて学習する。</p>	<p><ポインタ></p> <p>ポインタの扱いを学習し、アドレスによる値の参照方法について習得する。また、ポインタを利用した文字列、配列、関数の扱いについて学習する。</p>	<p><構造体></p> <p>構造体の扱いを学習し、変数や配列をまとめて扱う方法について習得する。また、構造体を使用したプログラムについて学習する。</p>	<p><コンソール入出力></p> <p>標準入出力や、対話型入出力について学習し、キーボードで入力した情報をプログラムに渡し、処理した後に表示する方法について学習する。</p>
	15	Light Weight Languageに関するスキル	PHP, Perl, Python, Ruby等	入門カリキュラムのため、特に規定しないが、以下の受講条件を満たすものとする。 ・プログラミング経験があること	<p><LightWeight Languageの基本></p> <p>LightWeight Languageプログラミングの概要として、歴史や特徴、適用範囲について学習する。また、プログラムを開発する流れについて習得する。</p>	<p><Perlの基本構造></p> <p>Perlの基本的な仕組みと構成、基本構文を学習し、プログラミングの内容を理解する。また、サンプルプログラムをもとに、簡単なプログラムの流れを確認する。</p>	<p><PHPの基本構造></p> <p>PHPの基本的な仕組みと構成、基本構文を学習し、プログラミングの内容を理解する。また、サンプルプログラムをもとに、簡単なプログラムの流れを確認する。</p>	<p><Pythonの基本構造></p> <p>Pythonの基本的な仕組みと構成、基本構文を学習し、プログラミングの内容を理解する。また、サンプルプログラムをもとに、簡単なプログラムの流れを確認する。</p>	<p><Rubyの基本構造></p> <p>Rubyの基本的な仕組みと構成、基本構文を学習し、プログラミングの内容を理解する。また、サンプルプログラムをもとに、簡単なプログラムの流れを確認する。</p>	<p><オブジェクト指向プログラミング></p> <p>Rubyのクラスの使用法を学習することで、オブジェクト指向の考え方を習得する。また、サンプルプログラムを作成する事で、オブジェクト指向プログラミングの利点を学習する。</p>	<p><組み込みクラス[データ構造]></p> <p>Rubyに組み込まれている、配列やハッシュといったデータ構造に特化したクラスについて学習する。また、サンプルプログラムを通して、データ構造に関するクラスの使用法を習得する。</p>

8	9	10	11	12	13	14	15	スキル名	
<p><ソフトウェアの開発環境> コンパイル対象となるソースファイルの数が多くなった場合は、gccでなくmakeを利用する。簡単なMakefileの内容を読むことで、どのように定義された処理が行われるかを理解する。CVSを利用してソフトウェア開発のバージョン管理を体験する。また配布する圧縮ファイルの作成と解凍・展開手順、パッチファイルの作成方法を学習する。</p>	<p><デバッグ> ソースプログラムを作成するとき、最初からデバッグを目的としたコードを挿入しておき、プリプロセッサでデバッグモードにコンパイルすることを体験する。また、gdbデバッグツールを利用する手順を習得する。ただし、究極的なデバッグ作業はソースプログラムにprintfを追加して確認する。</p>	<p><プロセスとスレッド> forkシステムコールを利用して子プロセスを生成し、複数の処理を並行に実行する。forkの戻り値によって、親プロセス、子プロセス、エラーを区別できる。6種類のexecファミリーを使用して、新しいプロセスを実行するプログラム例を作成する。スレッドは1つのプロセス内に複数作成することができ、forkよりも処理が軽いことを理解する。</p>	<p><シグナル> シグナルは割り込みの一種で、発生したことを非同期にプロセスに伝えることができる。signalで指定した事象が発生すると、指定したシグナルハンドラへ制御を渡す。具体的によく利用されるシグナルプログラム例を作成する。</p>	<p><プロセス間通信とパイプ> forkによって起動したプロセス間で情報交換を行うプロセス間通信(IPC: InterProcess Communication)を学習し、pipeによってプロセス間で情報を通信する手順を理解する。またpopenを利用したプログラム例を作成して実行する。</p>	<p><端末機器の入出力> すべてのデバイス(キーボード、ディスプレイ、プリンタ、ディスク、シリアルポート)をファイルとして扱えるUNIXの特徴を理解する。シリアル端子に接続した機器(例えば蛍光表示管:VFD)にメッセージを表示するプログラムやシリアル端子に接続したLEDやスイッチの情報を処理するプログラム例をあげて、動作させることで端末入出力に関する知識を習得する。</p>	<p><セマフォ、共有メモリ、メッセージキュー> 一般にSystemVのプロセス間通信と呼ばれているセマフォ、共有メモリ、メッセージキューについて理解する。セマフォは排他制御に利用される。パイプは親子関係にあるプロセス間でしか通信できないが、共有メモリはどのようなプロセス間でも利用できる。共有メモリはプロセス間で共通のメモリ領域を利用するが、メッセージキューはプロセス間でデータを送受信する機能であることを理解する。</p>	<p><ネットワークプログラミング> ネットワーク回線を介してクライアントとサーバ間でデータ通信を行うことを理解する。TCP、UDP、UNIXソケットのサーバに複数のクライアントを接続するシステムも可能である。スーパーサーバxinetd配下で動作するサーバプログラム例を作成する。</p>	Linuxのシステムプログラミングに関するスキル	8
<p><その他のネットワークサーバ導入の作業内容と手順> ネットワークサーバのさまざまな機能とその導入方法を理解する。</p>	<p><ネットワークサーバによるルーティング処理、フィルタリング処理の実装> ネットワークサーバによるルーティング処理、フィルタリング処理の実装方法を理解する。</p>	<p><ネットワークサーバによるインターネット接続> インターネットにアクセスするサーバの特徴、構築内容と作業手順を理解する。</p>	<p><サーバの運用管理業務> ネットワークサーバの運用管理作業の内容と管理項目の内容、重要性を理解する。 実際にネットワークサーバを運用管理する項目ごとにその運用内容や手順を理解する。</p>	<p><ログ管理の内容と手順> ネットワークサーバ管理の必須作業である「ログ管理」について作業内容と手順を理解する。</p>	<p><Linuxサーバセキュリティ> Linuxのサーバセキュリティの構築と運用の方法と手順を理解する。</p>	<p><Linuxのサービスセキュリティ> Linuxのサービスセキュリティの実装方法とその内容を理解する。</p>	<p><セキュアOSの機能と実装> セキュアOSとは何か、どんな要件を満たす必要があるのか、セキュアOSの種類と特徴について理解する。</p>	ネットワークサーバ管理に関するスキル	9
<p><並列プログラミング実践(2)> HPF(High Performance Fortran)とOpenMP HPFおよびOpenMPの基礎技術を習得する。</p>	<p><並列プログラミング 実践(3)> MPI(Message Passing Interface) HPCクラスタでのプログラミングのトレンドであるMPIの基礎技術を習得する。</p>			<p><Beowulf PCクラスタの構築> 一般的なPCクラスタの構築手順を理解する。</p>	<p><SCoreクラスタ> 国産のHPCクラスタ専用ミドルウェアSCoreの構築手順を理解する。</p>	<p><PCクラスタの周辺技術> HPCクラスタで使用すると便利なツールの紹介。</p>	<p><グリッド・コンピューティング> グリッド・コンピューティングに関する概念および最新の動向に関して理解する。</p>	クラスタシステム構築に関するスキル	10
<p><ルーティングの仕組み> ルーティングの重要性、仕様、問題点について、理解する。</p>	<p><ルーティングプロトコルの仕様> 具体的なルーティングプロトコルの分類とその内容について理解する。</p>	<p><TCPの仕組み> TCPプロトコルの特徴と業務への適用方法を理解する。</p>	<p><通信プロトコルの動作確認> 実際のLAN、TCP/IPプロトコルの動作を、プロトコルアナライザを用いて検証する。</p>	<p><TCPアプリケーションの仕組み・Web> Webの通信プロトコルHTTPの内容、動作仕様を理解する。実際の通信の内容をプロトコルアナライザで確認し、そのシーケンスを理解する。</p>	<p><TCPアプリケーションの仕組み・FTP> FTPの通信プロトコルの内容、動作仕様を理解する。実際の通信の内容をプロトコルアナライザで確認し、そのシーケンスを理解する。</p>	<p><TCPアプリケーションの仕組み・Telnet> Telnetの通信プロトコルの内容、動作仕様を理解する。実際の通信の内容をプロトコルアナライザで確認し、そのシーケンスを理解する。</p>	<p><新しいネットワークアーキテクチャ> 新しい情報通信端末と次世代の通信アーキテクチャに関して、その概要と特徴、構成、既存通信アーキテクチャとのかわりを学ぶ。</p>	ネットワーク・アーキテクチャに関するスキル	11
<p><ネットワーク管理プロトコルの概要> ネットワーク運用管理のインターネット標準プロトコルであるSNMPの動作仕様と特徴を理解する。</p>	<p><MRTGによるネットワーク管理の実装> ネットワーク運用管理のインターネット標準プロトコルであるSNMPの動作仕様と特徴を理解する。</p>	<p><ネットワーク運用設計> ネットワーク運用設計の目的、作業プロセスとその内容、注意点について理解する。</p>	<p><ネットワーク運用設計> ネットワーク運用設計の目的、作業プロセスとその内容、注意点について、実際に基づくワークショップによって理解する。</p>	<p><運用管理の実際的手順と体制> ネットワーク管理の個別作業の中でもっとも複雑な重要なキャパシティ管理の目的、内容や注意点を理解する。</p>	<p><WANの運用管理> WANサービスの運用管理方法として、日常監視の方法、ネットワーク事業者を交えた体制、障害予防の考え方(未然防止、再発防止)、障害発生時の切り分け方法などを理解する。</p>	<p><ネットワーク障害管理> ネットワーク運用管理業務の最大の課題である障害管理について日常監視の方法、障害予防の考え方(未然防止、再発防止)、障害発生時の切り分け方法などを理解する。</p>	<p><ネットワークトラブルシューティング> ネットワーク運用に潜むリスク、発生するトラブルについて、その検知および回避の方法論を学びながら、トラブル対応をどのように行っていくべきか検討する。</p>	ネットワーク管理に関するスキル	12
<p><MVCモデル> Webアプリケーションの基本アーキテクチャとしてのMVCモデルの内容、メリットを理解する。それぞれを構成する要素技術の意義とその内容を理解する。</p>	<p><EJBによるアプリケーション開発> EJBの機能を活用したアプリケーション開発のノウハウを理解する。</p>	<p><JavaによるServer処理実装の特徴と設計方法> JavaアーキテクチャによるWebサーバ/Webサービス実装の内容を理解する。また、有能なオープンソースシステムアーキテクチャの基盤となるJavaによるサーバ処理のアーキテクチャについて、設計方法論と利点/欠点について理解する。</p>	<p><JavaによるWebアプリケーションの設計/実装> Javaに対して「実開発への適用」を実務手順レベルで、実際にシステムを作成しながら解説し、アプリケーション開発の手順、技術内容を具体的に習得する。Javaの性能/保守性の最適化を実践できるノウハウを習得する。</p>	<p><JavaによるServer Side Java/Servlet+オープンソースデータベースを用いたサンプルアプリケーションを実装し、その手順や作業内容を理解する。></p>	<p><オブジェクト指向システム分析/設計/実装の実践技術> オブジェクト指向設計による設計・開発手順を実務レベルの設計を演習しながら実践的に理解する。オブジェクト指向開発を体験しながらJavaアプリケーションへの変換を実務レベルで理解する。</p>	<p><デザインパターンによる開発手順> デザインパターンによる設計の標準化、オブジェクト指向の強化の方法、内容、メリット、著名なデザインパターンについて理解する。</p>	<p><Javaのパフォーマンスチューニング> Java(J2EE)によるWebアプリケーション開発において、パフォーマンスを意識した設計や実装ができるようになる。効果的なパフォーマンスチューニングの戦略と手法を習得する。</p>	Javaに関するスキル	13
<p><ファイル管理> 高水準入出力関数と低水準入出力関数について学習し、ファイルに対してアクセスして、データを入力する方法について習得する。また、ファイル操作やディレクトリ操作といった、ファイルを管理する上で必要となるプログラムについて学習する。</p>	<p><データ構造> データを格納する仕組みを定義したデータ構造について学習し、用途に応じたデータ構造の使用方法を習得する。また、データ構造を使用したプログラムについて学習する。</p>	<p><C++の基本> C++プログラミングの概要として、歴史や特徴について学習する。また、Cと比較した上で、オブジェクト指向プログラミングの利点について学習する。最後に、C++によるプログラムを開発する流れについて習得する。</p>	<p><C++の基本構造> C++の基本的な仕組みと構成、基本構文を学習し、プログラミングの内容を理解する。また、サンプルプログラムをもとに、C++で作成したプログラムの流れを確認する。</p>	<p><オブジェクト指向プログラミング> C++のクラスの使用方法を学習することで、オブジェクト指向プログラミングの考え方を習得する。また、サンプルプログラムを作成することで、オブジェクト指向プログラミングの有効性を学習する。</p>	<p><STL(Standard Template Library)> コンテナとアルゴリズムのライブラリであり、C++標準の機能であるSTL(Standard Template Library)の構成や種類について学習する。また、サンプルプログラムをもとに、STLの使用方法について学習する。</p>	<p><GUIアプリケーションの開発> ユーザインタフェースを兼ね備えたGUIアプリケーションを開発するために、GUIライブラリであるGTK+とQtを導入し、使用方法について学習する。</p>	<p><開発ライブラリの使用> 開発ライブラリを使用する方法を学習し、データベースやコマンドラインオプションの解析など、多機能を備えたアプリケーション開発について習得する。</p>	C、C++に関するスキル	14
<p><組み込みクラス[データ操作]> Rubyに組み込まれている、数値や文字列、日付といったデータ操作に特化したクラスについて学習する。また、サンプルプログラムを通じて、データ操作に関するクラスの使用方法を習得する。</p>	<p><組み込みクラス[ファイル管理]> Rubyに組み込まれている、ファイルに対する入出力やファイル操作といった、ファイル管理に特化したクラスについて学習する。また、サンプルプログラムを通じて、ファイル管理に関するクラスの使用方法を習得する。</p>	<p><GUIアプリケーション開発> GUIアプリケーションを開発するために、GUIライブラリであるRuby/TkとRuby-GNOME2を使用する方法について学習する。</p>	<p><Ruby on Rails> Rubyを使用したWebアプリケーションを開発するフレームワークRuby on Railsの概念と構成といった仕組みについて学習する。また、Ruby on Railsで採用されているソフトウェア設計アーキテクチャであるMVCアーキテクチャの仕組みについて習得する。</p>	<p><データベースアプリケーション開発> Railsフレームワークを使用したデータベースアプリケーション開発の流れについて学習する。本回では、RailsフレームワークのActiveRecordを使用して、データベースへの操作を実現する方法を学習する。</p>	<p><Webアプリケーション開発> Railsフレームワークを使用したWebアプリケーション開発の流れについて学習する。また、WebアプリケーションフレームワークとしてのRailsの構成要素(Model、View、Controller)の機能について学習する。</p>	<p><プラグイン導入と開発> Railsの機能を拡張する方法として、プラグインの導入について学習する。プラグインは、既存のものを使用する方法と、新規にプラグインを作成する方法について学習する。</p>	<p><オープンソースシステムのカスタマイズ> オープンソースシステムとして公開されているRubricsというCMS(コンテンツマネジメントシステム)を利用して、独自に作成したコンポーネントを導入する事によってカスタマイズする方法について学習する。</p>	Light Weight Languageに関するスキル	15

OSSモデルカリキュラムコースウェア概要

黄色はレベル1(基本レベル)に、青色はレベル2(応用レベル)に対応しています。

	スキル名	概要	受講対象者、受講前提	1	2	3	4	5	6	7
開発体系	16 開発フレームワークに関するスキル	Struts, UML等	本カリキュラムの「Javaに関するスキル」を受講済みであること。または、同等の知識を有すること。システム開発経験及びプログラミング経験を有していること。	<開発フレームワークとは何か> <開発フレームワークの基本理念と目的、有効性、その発展の歴史、必然性、メリット、最新動向などを理解する。>	<開発フレームワークの種類と特徴> <開発フレームワークの基本的な仕組み、特徴、開発の実際のスタイル内容を理解する。サンプルプログラムをもとに、実際のプログラミングを行い、内容を確認する。>	<オープンソースによるWebアプリケーションのフレームワーク> <オープンソースによるWebアプリケーションの開発フレームワークの基本的な仕組み、特徴、プログラミングの内容を理解する。サンプルプログラムをもとに、実際のプログラミングを行い、内容を確認する。>	<フリーのWebコンテナ/J2EEコンテナの概要> <Java開発に関連するさまざまな無償のWebコンテナ/J2EEコンテナの種類と特徴を理解する。>	<オープンソースの開発ツール> <Java開発に関連するさまざまなオープンソースの種類と特徴を理解する。>	<開発フレームワークによる開発プロセスの手順> <開発フレームワークによるアプリケーションの開発の手順とその作業内容を理解する。Webアプリケーションの実現形態とそのメリットを理解する。>	<Ruby on Railsによるアプリケーション開発> <開発フレームワークRuby on Railsを用いたアプリケーション開発の概要を理解する。>
	17 開発ツールに関するスキル	バージョン管理システム、デバッグ、バグ追跡システム、システムプロファイラ、カーネルデバッグ等	入門カリキュラムのため、特に規定しない。基礎的なコンピュータ科学(ITSSレベル1程度)を習得、経験しているレベルの知識を有すること。	<開発の流れとツール> <ソフトウェア開発環境を構成するソフトウェア、ハードウェア、ツールの概要と開発やデバッグの進め方を学ぶ。>	<ソフトウェア開発環境の概要> <ソフトウェア開発環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。>	<Linux開発環境におけるソフトウェアアプリケーション開発の概要> <Linux開発環境におけるソフトウェアアプリケーション開発の作業内容、手順を実際に行い、理解する。サンプルを用いてプログラムを作成しながら、実習形式で開発ツールの特徴と役割を理解する。>	<バージョン管理ツールの活用> <バージョン管理ツールの機能と特徴、メリット・デメリット、バージョン管理ツールの開発手順を理解する。>	<デバッグによるプログラムデバッグの環境> <ソフトウェア開発環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。>	<カーネルデバッグを使用したデバッグ> <カーネルデバッグを使用したデバッグ環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。>	<バグ追跡システムを使用したデバッグ> <バグ追跡システムを使用したデバッグ環境の全体像とその構成、必要性を理解する。代表的なオープンソースのバグ追跡システム「bugzilla」を用いたデバッグ方法の特徴と役割を理解する。>
	18 統合開発環境に関するスキル	Eclipse, Net Beans, WideStudio等	ソフトウェア開発やプログラミングで用いられる基本的な用語を理解していること	<統合開発環境によるソフトウェア開発> <オープンソースソフトウェア開発で用いられている統合開発環境にどのようなものがあるかについて理解する。>	<さまざまな統合開発環境> <オープンソースソフトウェアとして開発されている「Eclipse」について、その背景や機能的特徴について理解する。>	<Eclipseとは何か> <オープンソースソフトウェアとして開発されている「Eclipse」について、その背景や機能的特徴について理解する。>	<NetBeans IDEとは何か> <オープンソースソフトウェアとして開発されている「NetBeans IDE」について、その背景や機能的特徴について理解する。>	<WideStudioとは何か> <オープンソースソフトウェアとして開発されている「WideStudio」について、その背景や機能的特徴について理解する。>	<Eclipse入門 - セットアップ> <Eclipseを利用するために必要な環境を適切に設定する方法を理解する。>	<Eclipse入門 - 基本操作> <Eclipseにおける、プロジェクトの作成、コーディング、ビルド、デバッグといった、基本的な操作を理解する。>
セキュリティ	19 暗号化に関するスキル	公開鍵基盤、電子署名、認証、ハッシュ関数	基礎的なコンピュータ科学、セキュリティ工学基礎(ITSSレベル1程度)を習得、経験を持つレベルの知識を有すること。	<セキュリティ機能と暗号化の位置づけ> <OSSにおけるセキュリティの基本概念とそこでの暗号化の役割、必然性、利点などを理解する。>	<暗号化の方式・共通鍵暗号方式> <暗号化の方式である「共通鍵暗号方式」の基本概念と仕組み、利点/欠点、動向を理解する。>	<暗号化の方式・公開鍵暗号方式> <暗号化の方式である「公開鍵暗号方式」の基本概念と仕組み、利点/欠点、動向を理解する。なぜ公開鍵方式がインターネット利用において重要なかを理解する。>	<情報システムにおける暗号化適用の方式> <実際の情報システムにおいて、暗号化がどのようなところで活用されているか、ソフトウェア、ハードウェア、ネットワークそれぞれについて、実装方法、目的、機能について理解する。>	<電子証明書の仕組み> <インターネットビジネスに必須のセキュリティ機能である電子証明書の仕様、仕組み、役割、必然性、利点などを理解する。>	<OSSの活用シーンと暗号化> <OSSにおけるセキュリティの基本概念とそこでの暗号化の役割、必然性、利点などを理解する。>	<無線LANの暗号化> <無線LANにおける暗号化の仕様、必然性、利点/欠点、運用仕様などを理解する。>
	20 ネットワークセキュリティに関するスキル	ファイアウォール設計・構築、ネットワーク侵入分析、ログ分析、セキュリティ攻撃に対する防御設計、TCP/IPにおける不正アクセス技術	基礎的なコンピュータ科学、セキュリティ工学基礎(ITSSレベル1程度)を習得、経験を持つレベルの知識を有すること。	<ネットワークセキュリティの概要> <ネットワークセキュリティの基本概念と必要な機能、その発展の歴史、必然性、利点、最新動向などを理解する。>	<ウィルスの特性と対策> <コンピュータウィルスの特性、発生する理由、対処方法、ウィルス対策ソフトウェアの特徴と運用方法などについて理解する。>	<ネットワーク攻撃方法の簡易的な分類> <ネットワークセキュリティの攻撃の種類、攻撃方法の概要とその影響について理解する。>	<TCPIにおける不正アクセス技術> <TCPの仕様に基づく不正攻撃とその内容について、デモンストレーションやワークショップを通して具体的に理解する。>	<Webにおける攻撃> <Webサービス/サーバへの不正アクセス/攻撃の内容とそのリスク、攻撃の手順とその対策を理解する。>	<IPにおける不正アクセス技術> <IPプロトコルを悪用した不正アクセスの内容とその方法について理解する。>	<TCP/IPネットワークセキュリティの設計方法> <TCP/IPネットワークおよびそこで動作するアプリケーションに対するネットワークセキュリティの設計・実装方法を理解する。>
	21 OSセキュリティに関するスキル	Linuxシステム・セキュリティ、セキュリティ強化OS	基礎的なコンピュータ科学、セキュリティ工学基礎(ITSSレベル1程度)を習得、経験を持つレベルの知識を有すること。	<OSのセキュリティ機能> <OSのセキュリティの基本概念と必要な機能、その発展の歴史、必然性、利点などを理解する。>	<Linuxサーバのローカルセキュリティ対策> <Linuxサーバのローカルセキュリティ対策の基本項目の内容と設定内容を実習で理解する。>	<Linuxのネットワークセキュリティ対策> <Linuxサーバのネットワークセキュリティ管理の基本設定、パッケージの導入などの方法とその作業内容を実習で理解する。>	<Linuxによるファイアウォール構築> <Linuxサーバによるファイアウォール設計、基本設定、導入などの方法とその作業内容を実習で理解する。>	<Linuxのサーバセキュリティ設定> <Linuxサーバセキュリティ管理の基本設定、アプリケーションパッケージの導入などの方法とその作業内容を実習で理解する。>	<安全なリモートアクセス> <Linuxの管理者権限の設定、アプリケーションパッケージの導入などの方法とその作業内容を実習で理解する。>	<SSLによるサーバVPNとCA> <LinuxサーバによるSSLによるサーバVPNとCAの導入、設定の方法とその作業内容を実習で理解する。>
RDB	22 RDBに関する基礎スキル	ERモデル、SQLプログラミング等	入門カリキュラムのため、特に規定しないが、以下の受講前提を満たすものとする。本カリキュラムの基本的なデータベース、コンピュータシステム基礎(ITSSレベル1程度)を習得、経験しているレベルの知識を有すること。	<データベースの基礎理論> <RDBMSの基本手順、データベースアクセスの方法論やデータベース適用方法・設計上の特性を学ぶ。>	<RDBMSの基本知識> <RDBMSの機能と特徴、その使用方法を理解する。アプリケーションとOS、RDBMSの関係を理解する。>	<トランザクションの基本概念> <データベースの特性およびデータベース設計に不可欠なトランザクションの概念と特徴について理解する。>	<データベースの構成要素> <関係データベースの構成要素である表、行、カラムなどの意味を理解し、それらの間の整合性を保持する方法や条件などを理解する。>	<DOAの内容概要> <データ中心分析の意義、目的、内容を理解する。従来のプロセス中心アプローチとの違い、メリットを理解する。>	<データベース設計の基本理論> <企業ビジネスデータをRDBで管理する場合の設計方法とその手順を理解する。その手法として特に正規化について、その方法やメリット・デメリットを理解する。>	<ERモデル> <関係データベース設計の重要な表現法である「ERモデル」について、その具体的な考え方・作成手順、記述方法を理解する。>
	23 RDBシステム管理に関するスキル	MySQL, FireBird, PostgreSQL等の導入、設定、チューニング	本カリキュラムの「RDBに関する基礎スキル」を受講済みであること。または、同等の知識を有すること。システム開発経験及びプログラミング経験を有していること。	<データベース運用管理の目的と項目> <データベース運用の必要性、方針、主要な問題点と内容とその理由、運用管理の基本的な項目、作業内容を理解する。>	<データベースの運用作業と障害回復> <一般的なデータベースの運用作業の種類とその内容、障害時の回復手順とその措置を理解する。>	<データベース運用設計> <データベースの運用設計作業の種類とその内容、検討項目、システム開発作業の中での実施手順と含まれているべき項目、内容を理解する。>	<データベースセキュリティ> <データベースセキュリティの内容、運用作業の中で実施しなければならないこと、実際のセキュリティ侵害発生時の対処方法などを理解する。>	<データベースリカバリ設計> <データベースリカバリ作業の内容、運用作業の中で実施しなければならないこと、実際の障害発生時の対処方法などを理解する。>	<データベースの最適化> <データベース性能向上施策、クライアントアプリケーションからの最適なアクセス方法、最適化、性能管理等のチューニングテクニックを学ぶ。>	<データベースのトラブル> <データベーストラブルの種類と原因、特徴、対策の内容と手順、結果評価方法などの内容を理解する。>

8	9	10	11	12	13	14	15	スキル名		
<p><Strutsとは> 開発フレームワークにおけるStrutsの内容とそのメリット、具体的なプログラミング方法を理解する。</p>	<p><MyFace(JSF)の開発モデルとは> 開発フレームワークにおけるMyFaceの内容とそのメリット、具体的なプログラミング方法を理解する。</p>	<p><データベース接続・アクセスのフレームワーク> データベース接続やアクセスを行うフレームワークと、ORMマッピングの仕様、フレームワークを理解する。</p>	<p><DlxAOPコンテナの概要> 新しいWebアプリケーションの基盤DlxAOPコンテナの基本的な仕組み、特徴、開発の内容を理解する。</p>	<p><Springフレームワーク> DlxAOPコンテナにもとづくSpringフレームワークの基本的な仕組み、特徴、開発の内容を理解する。</p>	<p><Seasar2> DlxAOPコンテナにもとづくSeasar2の基本的な仕組み、特徴、開発の内容を理解する。</p>	<p><Tapestry> Webアプリケーション開発におけるデザインとプログラマの作業をシームレスに結合することを目標に置いたフレームワークTapestryの基本的な仕組み、特徴、開発の内容を理解する。</p>	<p><Strutsによるアプリケーション開発> 開発フレームワークにおけるStrutsを用いたWebアプリケーション開発をワークショップによって実施し、その内容と実際の手順、具体的な内容を理解する。</p>	開発フレームワークに関するスキル	16	開発体系
講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	ワークショップ90分			
<p><オープンソース開発ツールの種類と機能> オープンソースの開発ツールの種類とそれぞれの機能、使用方法などを理解する。サンプルを用いて実習形式で開発ツールの特徴と役割を理解する。</p>	<p><統合開発環境を用いた開発手順> 統合開発環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p>	<p><オープンソース統合開発環境の種類と特徴> オープンソース統合開発環境の種類とその特徴、各ツールの構成、機能を実際に開発を行い、理解する。サンプルを用いて実習形式で開発ツールの特徴と役割を理解する。</p>	<p><Linux開発環境におけるソフトウェア開発ワークショップ> Linux開発環境におけるソフトウェアアプリケーション開発の作業内容、手順を実際に開発を行い、理解する。サンプルを用いて実習形式で開発ツールの特徴と役割を理解する。</p>	<p><Linux開発環境におけるソフトウェア開発支援ツール概要> Linux開発環境におけるソフトウェア開発を支援するツールの概要、機能、操作手順を実際に開発を行い、理解する。実習形式で開発ツールの特徴と役割を理解する。</p>	<p><ソフトウェア開発ツールの評価> ソフトウェア開発ツール評価の観点とその内容、必要性を理解する。開発環境を用いた開発評価方法の特徴と役割を理解する。</p>	<p><Eclipseを用いたソフトウェア開発> Eclipseを用いたソフトウェア開発環境におけるプログラム開発および環境構築について実際の事象ごとの手順と留意点を実習を行いながら理解する。</p>	<p><Eclipseを用いたソフトウェア開発ワークショップ> Eclipseを用いたソフトウェア開発環境におけるプログラム開発管理/デバッグについて実際の事象ごとの手順と留意点を実習を行いながら理解する。</p>	開発ツールに関するスキル	17	
講義+ワークショップ 90分	講義 90分	ワークショップ 90分	ワークショップ 90分	講義+ワークショップ 90分	講義 90分	ワークショップ 90分	ワークショップ 90分			
<p><NetBeans入門 - セットアップ> NetBeans IDEを利用するために必要な環境を適切に設定する方法を理解する。</p>	<p><NetBeans入門 - 基本操作> NetBeans IDEにおける、プロジェクトの作成、コーディング、ビルド、デバッグといった、基本的な操作を理解する。</p>	<p><WideStudio入門 - セットアップ> WideStudioを利用するために必要な環境を適切に設定する方法を理解する。</p>	<p><WideStudio入門 - 基本操作> WideStudioにおける、プロジェクトの作成、コーディング、ビルド、デバッグといった、基本的な操作を理解する。</p>	<p><EclipseによるJavaプログラミング> Eclipseにおける、Java Development Tools (JDT)を用いたJavaプログラミングについて、その機能概要や基本操作を理解する。</p>	<p><EclipseによるWebアプリケーション開発> Eclipseにおける、Web Tools Platform (WTP)を用いたWebアプリケーション開発について、その機能概要や基本操作を理解する。</p>	<p><NetBeansによるWebアプリケーション開発> NetBeans IDEによるWebアプリケーション開発について、その機能概要や基本操作を理解する。</p>	<p><WideStudioによるアプリケーション開発> WideStudioによる、マルチプラットフォームのアプリケーション開発について、その機能概要や基本操作を理解する。</p>	統合開発環境に関するスキル	18	
ワークショップ90分	ワークショップ90分	ワークショップ90分	ワークショップ90分	講義+ワークショップ120分	講義+ワークショップ120分	講義+ワークショップ120分	講義+ワークショップ120分			
<p><認証と暗号化> ネットワークにおける認証の実現方式と暗号化の役割、必然性、利点などを理解する。</p>	<p><IPsecによる暗号化通信> 新しいオープンソースネットワーク基盤におけるVPNのもととなるIPsecプロトコルについて、仕様、課題、役割、必然性、利点などを理解する。</p>	<p><SSHによるトンネリング> SSHを用いたクライアント/サーバ間の暗号化通信の仕組み、特徴、実装手順について理解する。</p>	<p><SSLプロトコルの仕組み> インターネット暗号化の仕組みの中核をなすプロトコルであるSSLについて、その仕様、課題、役割、必然性、利点などを理解する。</p>	<p><VPN通信の構築> IPsecを用いたVPNを設定構築し、その手順、機能効果、暗号化の状況などを理解する。</p>	<p><PKI(公開鍵暗号化基盤)の仕組み> 暗号化を運用する重要な基盤であるPKIについて、仕様、課題、役割、必然性、利点などを理解する。また、実際の運用仕様、代表的なPKIについても理解する。</p>	<p><認証基盤構築実習> PKIの仕組みをSSLによって実際に構築し、その動作仕様や暗号化の様子を検証する。</p>	<p><暗号化・これからの活用シーンと課題> 新しいオープンソースネットワーク基盤における暗号化のニーズとその仕様、課題、役割、必然性、利点などを概説する。</p>	暗号化に関するスキル	19	
講義90分	講義90分	講義90分	講義90分	講義+ワークショップ90分	講義90分	講義+ワークショップ90分	講義90分			
<p><アクセス制御の仕組みとファイアウォールの機能> ネットワークセキュリティの重要技術であるアクセス制御/フィルタリングの設定方法とその内容について理解する。</p>	<p><Linuxのネットワークセキュリティ対策> Linuxのネットワークセキュリティの機能とその設定方法・内容について理解する。</p>	<p><ネットワーク脆弱性調査> Webネットワークを題材に、ネットワーク脆弱性調査の仕様、方法と結果の評価方法を理解し、結果のネットワーク設計やセキュリティ対策へいかに反映するかを理解する。</p>	<p><セキュアなネットワークの構築> ネットワーク、ハニーポットなどの新しいネットワークセキュリティ実装の方法論についてその目的、内容、手順、特徴を理解する。</p>	<p><侵入検知システムの仕様と導入> 侵入検知システムの機能、仕様、利点とリスクを理解する。</p>	<p><IDSによる侵入検知> 実際にIDSを導入設定し、侵入検知の仕組みを構築する。実際に侵入を実施し、IDSの動作仕様を検証する。</p>	<p><ネットワークセキュリティ構築> ネットワークのセキュリティ要件を分析し、対策を実装するワークショップでセキュリティ構築の手順を理解する。</p>	<p><モバイルコンピューティングとリモートアクセスのセキュリティ> モバイルコンピューティングとリモートアクセスのセキュリティ実装方法とその内容について理解する。</p>	ネットワークセキュリティに関するスキル	20	セキュリティ
講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義90分	講義+ワークショップ90分	ワークショップ90分	ワークショップ90分	講義90分			
<p><ドメインネームサービスのセキュリティ対策> LinuxのDNSに関してセキュリティ設定、導入などの方法とその作業内容を実習で理解する。</p>	<p><電子メールのセキュリティ対策> Linuxのメールサーバ機能に関してセキュリティ設定、導入などの方法とその作業内容を実習で理解する。</p>	<p><Webのセキュリティ対策(1)> LinuxによるWebサーバの設定、導入方法とその作業内容を実習で理解する。</p>	<p><Webのセキュリティ対策(2)> Linuxの管理者権限の設定、アプリケーションパッケージの導入などの方法とその作業内容を実習で理解する。</p>	<p><ファイルサービスのセキュリティ対策> Linuxのファイルサービスのセキュリティ設定、導入などの方法とその作業内容を実習で理解する。</p>	<p><システムログの管理> Linuxのシステムログの管理の方法、手順、設定、作業内容を実習で理解する。</p>	<p><Linuxによる侵入検知の手法> Linuxサーバを用いて、サーバによる侵入検知の仕組み、方法、実際の作業内容を実習で理解する。</p>	<p><サーバのセキュリティ監査と設定の自動化> Linuxの管理者権限の設定、アプリケーションパッケージの導入などの方法とその作業内容を実習で理解する。</p>	OSセキュリティに関するスキル	21	
講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分			
<p><正規化の手順と方法> 関係データベース設計の必須技術である「正規化」について、その具体的な考え方・手順を理解する。</p>	<p><データベースインデックス> 関係データベースアクセス高速化のための必須技術であるインデックスの物理構造やその意味、性能やリカバリ特性への影響について、その具体的な考え方・手順を理解する。</p>	<p><データベースの物理構造> 関係データベース設計を実装するために必要なデータベースの物理構造やその意味、性能やリカバリ特性への影響について、その具体的な考え方・手順を理解する。</p>	<p><SQLによるデータベースアクセス> 関係データベースアクセス技術であるSQLの構造、原理、動作仕様、種類と特徴を理解する。オープンソースデータベースのMySQLを用いてSQLを使用し演習を行う。</p>	<p><SQL実践演習ワークショップ> 関係データベースアクセス技術であるSQLを実践するため、オープンソースデータベースのMySQLを用いてSQLの演習を行う。第11回で学んだSQL文を実際のアプリケーションテーブルを用いた実践的なワークショップで習得する。</p>	<p><代表的なオープンソースRDBMS製品> 代表的なオープンソースRDBMS製品について、その開発の歴史、機能、特徴、ライセンスなどについて理解する。</p>	<p><データベース設計構築の実践> RDBを用いた企業データベース設計ワークショップを行い、RDB設計と実装を理解する。</p>	<p><データベース構築> 第14回で設計したRDBを用いた企業データベースをSQLのみで実装し、アクセスし、データベースアプリケーションの設計と実装を理解する。</p>	RDBIに関する基礎スキル	22	RDB
講義 90分	講義 90分	講義 90分	講義+ワークショップ 90分	ワークショップ 90分	講義 90分	ワークショップ 90分	ワークショップ 90分			
<p><データベースチューニング> データベース性能向上のためのチューニングの方針、検討項目、実施手順、評価方法などの内容を理解する。</p>	<p><データベース構築> データベースの構築からアプリケーション実装までをオープンソースデータベースMySQLを用いて実習でその手順や内容、注意点を理解する。</p>	<p><データベースインデックスを用いた性能改善> 関係データベースアクセス高速化のための必須技術であるインデックスの物理構造やその意味、性能やリカバリ特性への影響について、その具体的な考え方、手順を理解する。</p>	<p><MySQLの導入と運用> 代表的なオープンソースデータベースであるMySQLの運用/管理上必要な知識について、サンプルデータベースを用いて実習で理解する。</p>	<p><データベーストラブルシューティング> データベースの障害に対して調査、問題の解明、対処を行い、トラブルシューティングの方法をマスターする。PostgreSQLを用いたワークショップ形式で実施する。</p>	<p><データベース運用環境構築> データベースの運用環境の構築・設定について、オープンソースデータベースMySQLを用いて実習でその手順や内容、注意点を理解する。</p>	<p><データベース運用> データベースの運用仕様の決定、運用機能の実装やバックアップリカバリ機能の実装方法についてオープンソースデータベースMySQLを用いて実習でその手順や内容、注意点を理解する。</p>	<p><データベースチューニング> データベースチューニング作業の実践として、オープンソースデータベースMySQLを用いて実習でその手順や内容、注意点を理解する。</p>	RDBシステム管理に関するスキル	23	
講義 90分	ワークショップ 90分	ワークショップ 90分	ワークショップ 90分	ワークショップ 90分	ワークショップ 90分	ワークショップ 90分	ワークショップ 90分			

OSSモデルカリキュラムコースウェア概要

黄色はレベル1(基本レベル)に、青色はレベル2(応用レベル)に対応しています。

	スキル名	概要	受講対象者、受講前提	1	2	3	4	5	6	7	
組み込みSW	24	組み込みシステムに関するスキル	システム構造、開発方法論、RTOS、センサープログラミング、組み込みプロセッサ・アーキテクチャ(ARM9、XScale、MIPS、SH、VR、MP、68k等)	本カリキュラムの基本となる組み込みコンピュータ科学基礎、ソフトウェア、ハードウェア基礎(ITSSレベル1程度)を習得、経験しているレベルの知識を有すること。	<p><組み込みコンピュータシステムとは何か> 組み込みコンピュータアーキテクチャの意味と発展の歴史、必要性を理解する。ソフトウェアやハードウェアのアーキテクチャとは何を規定するものか、その特徴と役割を理解する。</p> <p>講義90分</p>	<p><組み込みコンピュータのアーキテクチャ> 組み込みコンピュータアーキテクチャの基本構成と役割、特徴を理解する。その発展の歴史とオープンソースとのかかわりの概要を理解する。</p> <p>講義90分</p>	<p><組み込みシステムの基本構成> 組み込みシステムの応用分野、実現システムの種類とその内容、特徴、発展の動向と現状を理解する。</p> <p>講義90分</p>	<p><組み込みコンピュータハードウェアの基本> 組み込みコンピュータハードウェアの基本構成と役割、特徴を理解する。その発展の歴史とオープンソースとのかかわりの概要を理解する。</p> <p>講義90分</p>	<p><CPUアーキテクチャの基本> 組み込みコンピュータハードウェアの基本であるCPUの構成と役割、特徴を理解する。その発展の歴史とオープンソースとのかかわりの概要を理解する。</p> <p>講義90分</p>	<p><組み込みソフトウェアの概要> 組み込みソフトウェアの種類と役割、特徴を理解する。その発展の歴史とオープンソースとのかかわりの概要を理解する。</p> <p>講義90分</p>	<p><カーネル処理の概要> 組み込みOSのカーネルの機能と役割、特徴を理解する。その制御構造を理解する。</p> <p>講義90分</p>
	25	組み込み開発環境に関するスキル	クロスプラットフォーム、ツールチェーン(make、adb.minicom、Jflash、boothroad、tftp、tinybox)、GUIプログラミング(GTK+、QT、Qtopia)	本カリキュラムの基本となる組み込みコンピュータ科学基礎、ソフトウェア、ハードウェア基礎(ITSSレベル1程度)を習得・経験しているレベルの知識を有すること。	<p><組み込み開発の流れと環境> 組み込み開発環境を構成するソフトウェア、ハードウェア、ツールの概要と開発やデバッグの進め方を学ぶ。</p> <p>講義90分</p>	<p><組み込み開発環境の概要> 組み込み開発環境の全体像とその構成、必要性を理解する。</p> <p>講義90分</p>	<p><組み込み開発環境を用いた開発手順> 開発環境を用いた開発方法の特徴と役割を理解する。</p> <p>講義90分</p>	<p><プログラムデバッグの環境> 組み込み開発の基本的なプログラムデバッグのためのツールや方法の特徴を理解する。</p> <p>講義90分</p>	<p><デバッグソフトを使用したデバッグ環境> デバッグソフトを使用したデバッグ環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p> <p>講義90分</p>	<p><ICEを使用したデバッグ環境> ICEを使用したデバッグ環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p> <p>講義90分</p>	<p><ツールチェーンによるデバッグ> ツールチェーンの機能と特徴、利点/欠点、ツールチェーンを用いたデバッグの方法と手順を理解する。</p> <p>講義90分</p>
	26	組み込みアプリケーション開発に関するスキル	VM、J2ME、UPnP、SMS、WAP規格等	本カリキュラムの基本となる組み込みコンピュータ科学基礎、ソフトウェア、ハードウェア基礎(ITSSレベル1程度)を習得・経験しているレベルの知識を有すること。	<p><タスクとコンテキスト> オープンソースアプリケーション開発のためのタスク処理とコンテキストについて、開発の理由と特徴、設計内容と手順を学ぶ。</p> <p>講義90分</p>	<p><非同期と同期の設計仕様> 組み込みアプリケーション仕様として必須の非同期と同期の設計仕様について、実装仕様のパターンと設計内容、特徴、設計手順を学ぶ。</p> <p>講義90分</p>	<p><タスクの優先度とその制御仕様> 組み込みアプリケーション仕様として必須のタスクの優先度とその制御仕様について実装仕様のパターンと設計内容、特徴、設計手順を学ぶ。</p> <p>講義90分</p>	<p><組み込みアプリケーション間の資源配分技術> 組み込みアプリケーション同士で限定されたシステムリソースをいかに共有するかについて、実装仕様のパターンと設計内容、特徴、設計手順を学ぶ。</p> <p>講義90分</p>	<p><組み込みアプリケーション間のリソースの共有技術> 組み込みアプリケーション同士でシステムリソースを共有する具体的な実現方法について学ぶ。</p> <p>講義90分</p>	<p><リソース有効活用のアーキテクチャ> リソース有効活用の方式、実装方法を理解する。具体的な実装内容を理解する。</p> <p>講義90分</p>	<p><プログラムリソースの有効活用技術> プログラムリソース有効活用の方式、実装方法を理解する。具体的な実装内容を理解する。</p> <p>講義90分</p>
27	組み込みシステム最適化に関するスキル	低電力化、デバイスプログラミング、並列化	本カリキュラム体系の中の「組み込みシステムに関するスキル」を受講、もしくは同等のスキルレベルの知識を有すること。	<p><マルチプロセッサシステム> マルチプロセッサの種類、活用方法について理解する。</p> <p>講義90分</p>	<p><ハードウェアによる最適化> マルチプロセッサを使用したシステムの最適化について理解する。</p> <p>講義90分</p>	<p><リアルタイムシステムの設計> リアルタイムシステムの設計に必要な要素について理解する。</p> <p>講義90分</p>	<p><リアルタイムソフトウェアの条件と最適化> リアルタイムソフトウェアを利用したシステムの性能向上、最適化を行うためのソフトウェア作成方法、設計の方針について理解する。</p> <p>講義90分</p>	<p><性能最適化の評価項目> リアルタイム処理の設計を導入したシステムの性能評価について理解する。</p> <p>講義90分</p>	<p><ソフトウェアの最適化> ソフトウェアによるシステムの性能向上、最適化の基本概念、設計方針について理解する。</p> <p>講義90分</p>	<p><MPUの性能最適化設計> MPUを実装するシステムの性能向上を目指した設計の方針、最適化の基本概念を理解する。</p> <p>講義90分</p>	

8	9	10	11	12	13	14	15	スキル名		
<p><リアルタイムシステムの構成と仕組み> リアルタイムシステムの機能と制御仕様、設計の特徴を理解する。</p>	<p><組み込みシステムの開発方法> 組み込みシステムの開発方法について、基本的な考え方、技法、特徴、注意点を理解する。</p>	<p><オブジェクト指向分析による組み込みシステムの分析とモデリング> システムの分析・設計をオブジェクト指向で実施し、モデリングする。</p>	<p><組み込みコンピュータシステムの開発管理> 組み込みコンピュータシステムの開発管理の方法、必要なプロセスとその特徴を理解する。</p>	<p><組み込みシステムのアーキテクチャ> 組み込みコンピュータシステムの代表的な開発管理の方法、必要なプロセスとその特徴を理解する。</p>	<p><組み込みシステムのネットワーク機能> 組み込みコンピュータシステムのネットワーク機能の概要、特徴、利用方法、注意点を理解する。</p>	<p><組み込みシステムの設計> 実際の組み込みシステムの事例をもとに設計演習を行う。ハードウェア、ソフトウェア、ネットワークなどの要件をまとめてシステム設計を行う。</p>	<p><これからの組み込みシステムの動向> これからの組み込みシステムの応用分野、実現システムの種類とその内容、特徴、発展動向と現状を解説する。</p>	組み込みシステムに関するスキル	24	
講義90分	講義90分	ワークショップ90分	講義90分	講義90分	講義90分	ワークショップ90分	講義90分			
<p><組み込みアプリケーションデバッグの手順> 組み込み開発時のデバッグ方法やトラブル解消の手法を理解する。</p>	<p><組み込みアプリケーションの事例と開発環境> 組み込みアプリケーションの開発事例と、それに対応する開発環境の構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p>	<p><組み込みクロス開発環境の構築> 組み込みクロス開発環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p>	<p><GNU開発環境の特徴> GNU開発環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p>	<p><GNU開発環境における組み込みアプリケーション開発> GNU開発環境における組み込みアプリケーション開発を実際に行いながら理解する。GNU開発環境を用いた開発方法の特徴、役割、手順を理解する。</p>	<p><組み込みLinux開発最新デバッグ環境> 組み込みLinux開発最新デバッグ環境の全体像とその構成、必要性を理解する。開発環境を用いた開発方法の特徴と役割を理解する。</p>	<p><組み込み開発環境の評価> 組み込み開発環境評価の観点とその内容、必要性を理解する。開発環境を用いた開発評価方法の特徴と役割を理解する。</p>	<p><組み込み開発環境におけるデバッグのパターン演習> 組み込み開発環境におけるデバッグについて実際の事象ごとの手順と留意点を実習を行いながら理解する。</p>	組み込み開発環境に関するスキル	25	
講義90分	ワークショップ90分	講義90分	講義90分	ワークショップ90分	講義90分	講義90分	ケースショップ90分			
<p><入出力待ちリソースの有効利用> 入出力待ち時間の有効利用方法と制約、特徴を理解する。</p>	<p><入出力資源管理> 組み込みアプリケーション設計における最適な入出力資源管理プログラムの構造と設計について、その内容と設計の方法を理解する。</p>	<p><J2MEの仕様> 組み込み開発のオープンソース開発技術であるJ2ME (Java 2 Micro Edition) の仕組みと機能、動作仕様などについて理解する。</p>	<p><VMの概要とその活用> 組み込みアプリケーション設計開発におけるVMの特徴、活用方法を理解する。</p>	<p><高信頼性の実装> 組み込みアプリケーションにおける高信頼性の実装方法、活用事例を中心に紹介し、その実際を理解する。</p>	<p><組み込みアプリケーションの実装事例研究> 多様な組み込みアプリケーションの実装仕様の事例研究を行い、設計方針や機能分割、実装方法を学ぶ。</p>	<p><クリティカルなマイコンアプリケーションを活用した設計ケースワーク> マイコンアプリケーションの設計ワークショップを行う。ロボットのアプリケーションを設計する。ハードウェア、動作プラットフォーム前提で制約を理解し、組み合わせでアプリケーションを設計するケースワークを実施する。ハードウェア設計をもとに設計を行う。</p>	<p><組み込みアプリケーションソフトウェアの実装> 第14回で設計したシステム基盤を構築する。J2MEアーキテクチャを用いて、第14回の設計の中の制御仕様の一部の設計を行う。第13回の設計内容の一部モジュールを実装する。</p>	組み込みアプリケーション開発に関するスキル	26	組み込みSW
講義90分	講義90分	講義90分	講義90分	講義90分	ワークショップ90分	ワークショップ90分	ワークショップ90分			
<p><システムの性能要件と評価項目> システム性能の要件、考え方、評価方法を理解する。</p>	<p><システム性能の評価方法> システム性能の考え方、計測、量産時の制約事項とのトレードオフを理解する。</p>	<p><性能評価手法の分類> 性能評価の要件、評価方法、評価手順について理解する。</p>	<p><拡張性の評価> 拡張性評価の要件の実現方法・確認方法について理解する。</p>	<p><システム資源のトレードオフ> システム資源のトレードオフの概念と方法論及びそのポイントについて理解する。</p>	<p><基本ソフトウェアと応用ソフトウェアのトレードオフ> 組み込みシステム最適化でもっとも重要な設計方法である基本ソフトウェアとアプリケーションとの間の拡張性、性能向上のためのシステム構成要素や実現機能、資源のトレードオフの概念と方法及びそのポイントについて方法、手順とその内容を理解する。</p>	<p><組み込みシステム最適化のための方式設計> 組み込みシステムの性能向上のためのシステム方式設計の基本概念と設計方針などを理解する。</p>	<p><最適化のための検討項目> 組み込みシステムを最適化するためのハードウェア、ソフトウェア、ネットワーク、設計方法など、各要素ごとの検討項目などを理解する。</p>	組み込みシステム最適化に関するスキル	27	
講義90分	講義90分	講義90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分	講義+ワークショップ90分			