東京工業高等専門学校	開講年度	令和03年度 (2021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	0132	科目区分	専門 / 必修
授業形態	授業	単位の種別と単位数	学修単位: 2
開設学科	情報工学科	対象学年	4
開設期	後期	週時間数	2
教科書/教材	システムプログラミング入門 -UNIXシステムコール，演習による理解-		
担当教員	鈴木 雅人		

到達目標

言語処理とオートマトンの基礎的な概念を理解する
Linux上でのプログラミング開発やソースコードの管理について理解する
IoT開発の基礎技術について演習を通して理解する

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	最低限の到達レベルの目安(可)	未到達レベルの目安
言語処理とオートマトンの基礎的な概念を理解している	言語処理とオートマトンの概念を理解し，設計や理論的な考えに役立てることができる.	言語処理とオートマトンの概念を理解し，自分の言葉で説明できる	言語処理の意味やオートマトンの概念についてある程度説明できる	言語処理やオートマトンの概念を理解していない
Linux上でのプログラミング開発やソースコードの管理について理解している	Linux上でのプログラミング開発やソースコードの管理を理解し積極的に開発に役立てることができる.	Linux上でのプログラミング開発やソースコードの管理方法について説明できる.	Linux上でのプログラミング開発やソースコードの管理について理解している.	Linux上でのプログラミング開発やソースコードの管理について理解できていない.
IoT開発の基礎技術について理解している.	IoT開発の基礎技術について理解し，ソフトウェアとハードウェアの両面から設計開発を推進できる.	IoT開発の基礎技術について理解し，一部の開発を推進できる.	IoT開発の基礎技術について理解している.	IoT開発の基礎技術について理解していない.

学科の到達目標項目との関係

教育方法等

概要	計算機システムを用いたソフトウェア開発に関する基礎知識を習得する．Linux上でのプログラミング環境構築や，gitを用いたソースコードの管理が行えるようになることを目標とする．特に，ハードウェアとの連携したプログラミング開発技術について学ぶことで，IoT技術に関する基礎を習得する．この科目は，企業でシステム開発等を担当していた教員が，その経験を活かしてLinux上でのシステム開発の基礎や，ソースコードの管理，ハードウェアとの連携について講義・演習形式で授業を行うものである.
授業の進め方・方法	独自の教材を用いて説明を行い，学んだ内容を使った演習を行う．座学講義と演習を交互に実施することでシステムプログラミングに関する深い理解を得る.
注意点	C言語の基礎，計算機工学で学んだコンピュータの仕組みに関する知識がベースとなる. この科目は学習単位科目のため，事前・事後学習を良く行い，課題に取り組むこと.

授業の属性・履修上の区分

□ アクティブラーニング	□ ICT 利用	□ 遠隔授業対応	□ 実務経験のある教員による授業

授業計画

		週	授業内容	週ごとの到達目標
後期	3rdQ	1週	授業ガイダンス　Gitとソースコードの管理	授業の目的を理解し，Gitの使い方について説明できる.
		2週	RaspberryPiを用いた環境の構築	RaspberryPiを用いた環境が構築できる
		3週	Pythonプログラミングの基礎	Pythonプログラミングの基礎が身についている
		4週	RaspberryPiを用いたLEDの制御	RaspberryPiを用いてLEDを制御できる
		5週	RaspberryPiを用いたサーボモータの制御	RaspberryPiを用いてサーボモータを制御できる
		6週	RaspberryPiを用いたWebアプリケーションの開発１	RaspberryPiを用いた基本的なWebアプリケーション開発ができる
		7週	RaspberryPiを用いたWebアプリケーションの開発２	RaspberryPiを用いた基本的なWebアプリケーション開発ができる
		8週	IoTシステム開発演習	Raspberry Piを用いて基本的なIoT開発が推進できる.
	4thQ	9週	IoTシステム開発演習	Raspberry Piを用いて基本的なIoT開発が推進できる.
		10週	オートマトンと言語理論１	オートマトンと形式言語の概念について説明できる.
		11週	オートマトンと言語処理２	決定性有限オートマトン，非決定性有限オートマトン，ε遷移付き非決定性有限オートマトンについて説明できる.
		12週	正規表現と有限オートマトン	正規表現と有限オートマトンの関係性について説明できる.
		13週	コンピュータとオペレーティングシステム１	OSの位置づけ，プロセス管理，スケジューリングについて説明できる.
		14週	コンピュータとオペレーティングシステム２	共有資源と排他制御，記憶管理の基本的な考え方を説明できる.
		15週	まとめとテスト対策	授業全体の内容を振り返り各単元で学んだ内容が説明できる.
		16週	学年末試験	60点以上で合格

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

専門的能力	分野別の専門工学	情報系分野	ソフトウェア	ソフトウェアを中心としたシステム開発のプロセスを説明できる。	4	
			システムプログラム	コンピュータシステムにおけるオペレーティングシステムの位置づけを説明できる。	4	後13
				プロセス管理やスケジューリングなどCPUの仮想化について説明できる。	4	後13
				排他制御の基本的な考え方について説明できる。	4	後14
				記憶管理の基本的な考え方について説明できる。	4	後14
				形式言語の概念について説明できる。	4	後10,後11
				オートマトンの概念について説明できる。	4	後10,後11
				コンパイラの役割と仕組みについて説明できる。	4	後10,後11
				形式言語が制限の多さにしたがって分類されることを説明できる。	4	後10,後11
				正規表現と有限オートマトンの関係を説明できる。	4	後12
	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】	標準的な開発ツールを用いてプログラミングするための開発環境構築ができる。	4	後1,後2
				要求仕様にあったソフトウェア(アプリケーション)を構築するために必要なツールや開発環境を構築することができる。	4	後1,後2

評価割合

	試験	プログラミング課題	演習プリント	態度	ポートフォリオ	その他	合計
総合評価割合	50	25	25	0	0	0	100
基礎的能力	0	0	0	0	0	0	0
専門的能力	50	25	25	0	0	0	100
分野横断的能力	0	0	0	0	0	0	0