

先導的ソフトウェア科目

— 平成19年度 履修案内 —

情報システム工学コース

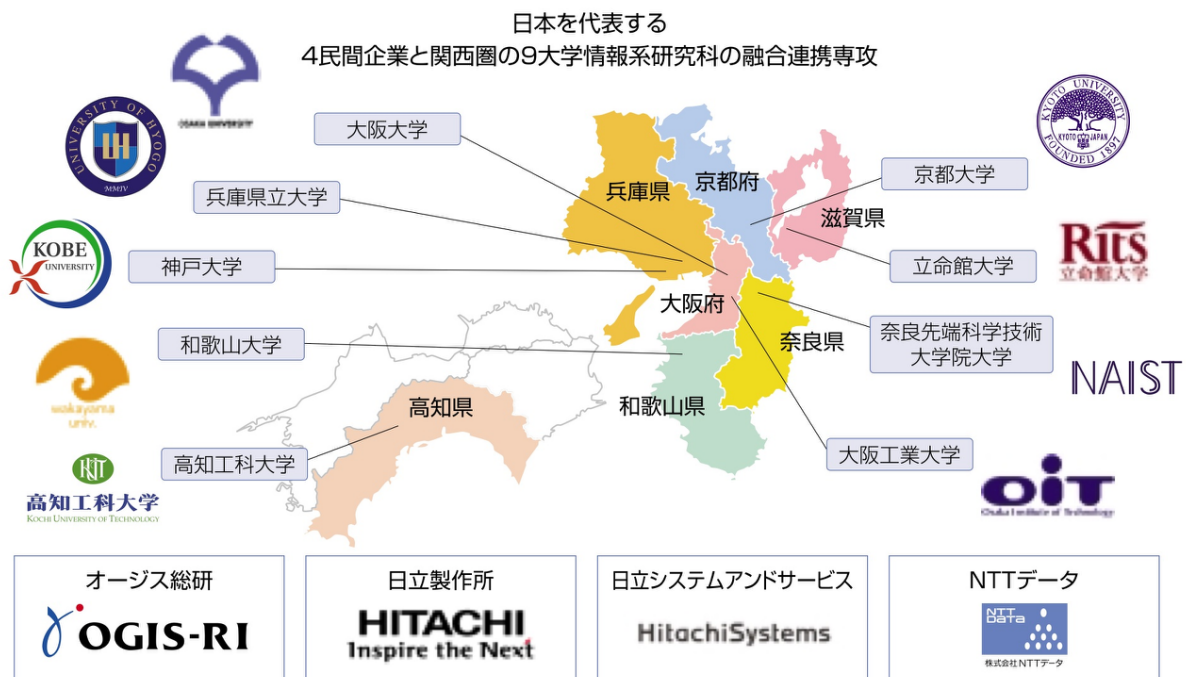
概要

情報システム工学コースは、関西を中心とする9大学が連携して行う教育プログラムである IT Spiral に参加しています。IT Spiralは文部科学省の主導する「先導的ITスペシャリスト育成推進プログラム」の一環として平成19年度から3年間行われ、ソフトウェア工学分野の先進的かつ実地的な知識を幅広く習得した技術者の輩出を目指します。

本教育プログラムの対象となるのは修士課程1年生です。平成19年度は情報システムコースから2名の参加を予定しています。参加した学生は、本学での通常の授業のほかに、ビデオ教材を利用して他大学の教員の授業を受講し、さらに2週間に一度、大阪大学の中之島センターに出向いて講義と演習を受講してもらいます。

IT Spiral対応科目

情報システム工学コースでは、今年度より IT Spiralに対応した履修科目として新たに「先導的ソフトウェア科目群」を設定しています。修士課程の修了のためにはセミナー4単位、特別研究8単位に加え、講義で18単位以上を修得する必要がありますが、先導的ソフトウェア科目群も修了のための単位数に含めることができます。



先導的ソフトウェア科目群は以下の3つのグループに分かれます。IT Spiralに参加する学生はこれらの科目を履修することになります。一方、IT Spiralに参加しない学生は以下の(B)、(C)の科目を履修することはできません。

(A) 基礎ソフトウェア工学科目群 情報システム工学コースの既存の次の科目です。

| | |
|----------------|-----|
| コンピュータアーキテクチャ論 | 2単位 |
| オペレーティングシステム論 | 2単位 |
| アルゴリズム論 | 2単位 |
| 応用ソフトウェア工学論 | 2単位 |

(B) 先端ソフトウェア工学科目群 ビデオ教材により、他大学の教員の講義を受けます。

| | |
|--------------|-----|
| 先導的ソフトウェア設計論 | 2単位 |
| 先導的ソフトウェア構成論 | 2単位 |

(C) 実践ソフトウェア開発科目群 大阪大学で開講される科目です。高知工科大学の単位として認定されます。

| | |
|--------------|-----|
| 実践プロジェクト管理 | 2単位 |
| 実践ソフトウェア開発論 | 2単位 |
| 実践ソフトウェア開発演習 | 2単位 |

先端ソフトウェア工学科目

ビデオ教材を使って他大学の教員の講義を受けます。授業は高知工科大学で開講され、ビデオ教材の視聴と関連する課題に対する演習を行います。それぞれ5コマからなる3つのまとまり（チャプター）から構成されます。

平成19年度のチャプター構成は次の通りです。

先導的ソフトウェア設計論

- (1) モデル中心ソフトウェア開発（京都大学）
- (2) アジャイルソフトウェア工学（奈良先端大）
- (3) ソフトウェア保守（大阪大学）

先導的ソフトウェア構成論

- (1) 組み込みシステム構成論（高知工科大学）
- (2) 組み込みソフトウェア設計論（兵庫県立大学）
- (3) コンポーネント/パターン指向ソフトウェア開発（立命館大学）

実践ソフトウェア開発科目

ソフトウェア開発における高い技術力と実績を有する民間企業4社のノウハウを活かした、実践的な講義・演習です。この講義・演習は、大阪大学中之島センターに、受講生が一堂に集まって行います。なお、演習では複数の大学院の学生と一緒に小規模グループでソフトウェア開発プロジェクトを進めて行きます。この科目は一ヶ月に2回、金曜日に行います（平成19年度の場合）。

高知工科大学の学生には、交通費と宿泊費が支給されます。

| | |
|--------------|---|
| 実践プロジェクト管理 | ヒューマンスキル技術（コミュニケーション技術、プレゼンテーション技術等）、マネジメント技術（プロジェクト管理技術、検証と品質保証技術等）、関連知識（コンプライアンス、内部統制等）など |
| 実践ソフトウェア開発論 | UML、ERモデル、Struts、エンピリカル技法、バグ管理など |
| 実践ソフトウェア開発演習 | 実践ソフトウェア開発論の演習 |

平成19年度の開講日

| | | |
|-----------|------------|------------|
| 4月 20日（金） | 6月 29日（金） | 11月 9日（金） |
| 4月 27日（金） | 7月 13日（金） | 11月 16日（金） |
| 5月 11日（金） | 7月 27日（金） | 11月 30日（金） |
| 5月 18日（金） | 10月 5日（金） | 12月 14日（金） |
| 6月 1日（金） | 10月 12日（金） | 12月 21日（金） |
| 6月 15日（金） | 10月 26日（金） | |

開講科目の内容

先端ソフトウェア工学科目

先導的ソフトウェア設計論

チャプター1：モデル中心ソフトウェア開発（京都大学）

第1回 沢田篤史 モデルを用いたソフトウェア開発

ソフトウェア開発におけるモデルの活用について概説する。モデルの意義、良いモデルの満たすべき性質（妥当性、完全性、無矛盾性など）、モデル間の追跡性（水平、垂直）について解説し、ソフトウェア開発を支援するために必要なモデル管理の要件について説明する。

第2回 沢田篤史 メタモデリングとソフトウェア開発支援

ソフトウェア開発支援に必要となるメタモデルについて解説する。メタモデルをスキーマとしたソフトウェアリポジトリの構成法と、それを利用した追跡性の管理について、例題・演習を含めた解説を行う。

第3回 沢田篤史 モデル駆動アーキテクチャとモデル 駆動開発（1）

OMGの提唱するモデル駆動アーキテクチャ（MDA）について概説する。MDAの技術要素である、CIM、PIM、PSM、モデル変換、マッピング、MOFなどについて解説する。

第4回 沢田篤史 モデル駆動アーキテクチャとモデル 駆動開発（2）

MDAの考え方にしななれたモデル駆動ソフトウェア開発（MDD）について例題・演習を含めた解説を行う。第2回で解説したメタモデルを利用したモデル変換やコード生成について解説し、MDA/MDDの自動化技術に対する理解を深める。

第5回 沢田篤史 ソフトウェアテスト・検証へのモデルの活用

ソフトウェアテストおよび検証におけるモデルの利活用について概説する。モデルレベルでのテスト技法、モデル変換によるテストモデルの生成などについて解説するとともに、モデル検査検証についても概説する。

チャプター2：アジャイルソフトウェア工学（奈良先端大）

第1回 Michael Barker 飯田 元 アジャイルの概念と方法論の概要

- ・アジャイル（俊敏）な開発と呼ばれる方法論発生の経緯と基本概念
- ・代表的なアジャイル開発方法論（XP, Scrum その他）やアジャイルモデリングについて紹介する
- ・従来の開発管理法との対比について議論する
- ・アジャイル開発における要求分析とテストについて議論する

第2回 Michael Barker プロジェクト管理・リスク管理とアジャイル

- ・プロジェクト管理知識体系PMBOKの概説
- ・PMBOKとアジャイル開発プロジェクトとの関連について議論する
- ・リスク管理の概念について概説し、アジャイル開発に特有のリスク要因とその管理手法について説明する

第3回 飯田 元 プロセスのモデリングとアジャイル

- ・ソフトウェアプロセスモデリングの概念
- ・プロセスモデリング手法PRePの紹介
- ・プロセスモデリングの観点からアジャイル開発について議論する

第4回 Michael Barker 英語プレゼンテーション法ほか（特別講義）

- ・英語による論文執筆方法
- ・ソフトウェア工学分野における注意点
- ・主要ジャーナル、国際会議の紹介

第5回 Michael Barker 英語プレゼンテーション法ほか（特別講義）

- ・英語によるスライド作成とプレゼンテーション方法
- ・ソフトウェア工学分野における注意点
- ・主要ジャーナル、国際会議の紹介

チャプター3：ソフトウェア保守（大阪大学）

第1回 楠本真二 ソフトウェア保守の概要

ソフトウェア保守の概要、ソフトウェア保守における諸問題等について説明する。

第2回 楠本真二 保守の見積

新規開発と同様にソフトウェア保守作業についても、保守規模や保守工数の見積は保守計画を作成する上で必要な作業である。本講義では、ソフトウェア保守の際に用いられる見積手法や見積モデル等について紹介する。

第3回 松下 誠 ソフトウェア理解支援

ソフトウェア保守作業では、まず、保守対象のソフトウェアの全体的な理解や保守範囲を把握する必要がある。本講義では、プログラム理解や影響波及解析手法について紹介する。

第4回 井上克郎 コードクローン検出と分析

コードクローンとは、ソースコード中に存在する、他のコード片と一致または類似しているコード片を意味する。一般にコードクローンはプログラムの保守性に悪影響を与える要因と言われている。本講義では、コードクローンの検出方法、分析方法について紹介する。

第5回 松下 誠 ソフトウェア修正支援

保守作業において、ソフトウェアを修正する時には、過去に実施された類似の修正作業に関する情報を再利用することが有効であることが知られている。本講義では、バージョン履歴情報から、保守を行う上で有益な情報を効率よく検出するための手法について紹介する。

先導的ソフトウェア構成論

チャプター1：組み込みシステム構成論（高知工科大学）

第1回 鶴保証城、渡辺 登 組み込みソフトウェアの現状と今後

組み込みソフトウェアには、ハードウェア依存性が高いこと、非決定性部分を含むために検証にかかるコストが大きいことといった特徴がある。高い信頼性が求められる一方で、開発の多くの場面で熟練者のノウハウに依存するといった問題もある。さまざまな組み込みシステムとその開発の現状について、事例を示しながら説明する。また、生産性向上に向けた取り組みについて紹介する。

第2回 岩田 誠 ハードウェアの基礎

組み込みソフトウェアの開発対象は極めて多岐にわたり、しかも技術革新による変化のスピードが速い。汎用マイコン、DSP、リコンフィギュラブルLSI、FPGAなどのハードウェアの基礎について説明し、組み込みシステムへの応用について述べる。低消費電力、高速化、低コストを実現するための取り組みについても触れる。

第3回 荻原剛志 アセンブラプログラミング

実行速度やメモリ使用量が重要視される場合や、ハードウェア固有の性能を引き出したい場合などには、高水準言語ではなくアセンブラによるプログラミングが必要とされる。また、高水準言語でプログラムする場合でも、生成される機械語と対象機器のアーキテクチャの関係を意識することは重要である。ここではアセンブラとしてGNU asを取り上げて説明し、次にサブルーチンとして利用するための書き方についても学ぶ。

第4回 酒居敬一 MCUのアーキテクチャとクロス開発

シンプルなマイクロコントローラユニットであるRenesus H8/3292を例として、アーキテクチャを説明する。C言語とアセンブラを用いてプログラムをクロス開発する方法について述べ、外部インタフェースへのアクセスと割り込み処理についても説明する。また、ROMモニタを用いたデバッグの方法などについても触れる。

第5回 平山雅之 組み込みシステムのコーディング規約

多人数でのソフトウェア開発でソースコードの品質を保つためには、一定のコーディング規約を守ることが重要となる。組み込みソフトウェアの品質向上に向け、産学官が連携して策定したコーディング規約について説明する。同時に、ソフトウェアの品質特性について述べ、テストをしやすくし、バグを作り込まないようにする信頼性の高いコーディング方法について学ぶ。

チャプター2：組み込みソフトウェア設計論（兵庫県立大学）

第1回 中本幸一 組み込みソフトウェア概論

組み込みシステムは、リアルタイム性能の重視、並行プロセス、省メモリ容量、省電力、クロスソフトウェア開発環境など、企業向けソフトウェアとは異なる要求事項を有する。本講義ではこれらの要求事項とその解、特に資源管理手法を述べる。

第2回 中本幸一 組み込みソフトウェア設計概論

組み込みシステムは外界でのイベントを入力し適切な応答を適切な時間以内で出力するという形態をとる。このためのソフトウェア設計手法として、状態遷移、時間駆動型設計プロダクトライン、設計モデリング技術などがある。本講義ではこれらの手法の背景と概要を述べる。

- 第3回 中本幸一 リアルタイムスケジューリング概論
組込みシステムは外界を制御する用途が多い。外界を制御するためにある一定時間以内に処理を行う必要がある。これをリアルタイムスケジューリングといい、各種の技法が研究されてきている。本講義では Rate Monotonic, Earlier Deadling Firstの各スケジューリング技法、優先度逆転問題とそれを解決する優先度継承プロトコルなどを述べる。
- 第4回 中本幸一 組込みシステム向けJava概要
携帯電話をはじめとして組込みシステムのエンドユーザー向けアプリケーションとしてJava実行環境が利用されることが多い。本講義では情報家電で利用されるConnected Device Configuration、携帯電話で使用されるConnected Limited Device Configuration、ICカードで利用されるJavaCard、関連応用ライブラリを概説する。
- 第5回 中本幸一 組込みシステム向けミドルウェア
組込みソフトウェアも要求される機能の増加、開発規模の増大により、ミドルウェア導入による、生産性、信頼性の効用が進められている。応用領域に対して共通なミドルウェアとして組込みシステム向けCORBA、応用領域専用のミドルウェアとして情報家電向けミドルウェアのUPnP、OSGi、携帯電話のミドルウェアのMOAPなどを紹介する。
- チャプター3：コンポーネント/パターン指向ソフトウェア開発（立命館大学）
- 第1回 丸山勝久 ソフトウェア開発におけるコンポーネントとパターン
コンポーネント指向ソフトウェア開発を紹介し、従来のソフトウェア開発手法との違い、利点をまとめる。また、パターンを取り入れたソフトウェア開発に関して、その概要を紹介する。
- 第2回 丸山勝久 コンポーネント指向開発方法論
開発方法論として、UMLコンポーネント設計、KobrAを取り上げ、それぞれのプロセスや特徴をまとめる。また、プロダクトライン開発についても解説する。
- 第3回 丸山勝久 コンポーネントアーキテクチャと実装技術
代表的なコンポーネントアーキテクチャとその実装技術に関して歴史的背景を紹介する。また、コンポーネント実装技術の具体例として、EJB, Webサービス技術(SOAP, WSDL, REST)などを解説する。
- 第4回 丸山勝久 ソフトウェアパターン
ソフトウェア開発工程におけるソフトウェアパターンを紹介する。さらに、分析パターン、アーキテクチャパターン、デザインパターンを解説する。
- 第5回 丸山勝久 リファクタリング
オブジェクト指向の設計原則と設計を改善するためのリファクタリングについて解説する。

実践ソフトウェア開発科目

実践プロジェクト管理

情報産業の現状や課題について概説した上で、ソフトウェアプロジェクト管理に関する技術の詳細について学ぶ。特に、現在標準的に開発現場で用いられているコミュニケーション技術、ソフトウェアのテストやレビューに代表される品質保証技術、要求分析を行なう上で必須となる要求獲得・定義手法、技術について実例を用いて解説する。また、近年のウェブアプリケーション開発における最新の話題についても紹介する。

- プロジェクト管理、ヒューマンスキル、要求分析、テスト技術

実践ソフトウェア開発論

実践ソフトウェア開発演習

業務アプリケーションの開発プロセスを例題を通じて体験する。具体的には、実用規模のウェブアプリケーションソフトウェアの仕様書をUML (Unified Modeling Language) を用いてモデル化し、ファンクションポイント等を用いた見積もりを行う。次に、複数人のチームに分かれて実装する。実装には、Javaと現在標準的に用いられているフレームワークである Struts を使用する。実装したプログラムに対する品質保証活動（テスト、レビュー）も実施する。プログラム開発時には、データ収集・分析ツールを用いて、プログラムの構成管理情報、バグ情報、メールを通じたチーム内でのコミュニケーション情報の収集を行なう。収集したデータを基に、各チームの進捗管理やバグ管理を行う。最後に、開発したプログラムの複雑さや保守性を様々な解析ツールを用いて評価し、改善点や改良法について議論する。以上のような開発プロセスを通じて、実践的なソフトウェア開発・管理技術を体得する。

- UML、Struts、Java、Webアプリケーション、進捗・品質管理、プログラム解析

