

## 1. 科目コード

1207

## 2. 科目名

ソフトウェア工学基礎論 (Fundamentals of Software Engineering)

## 3. 担当教員

伊藤 守 (Mamoru ITO)

## 4. 開講期

春2期 ((昼) 木曜 3-4 時限、(夜) 木曜 6-7 時限)

## 5. 科目の目的・概要

実用的なソフトウェア・システムやソフトウェア製品を開発するために必要となる開発プロセスや開発技術（分析、設計、検証）に関する基礎知識と実践的なプロジェクトマネジメント手法を学ぶ。また、ソフトウェアの特徴やソフトウェアを取り巻く社会環境を理解し、開発現場で直面する様々な問題に対する判断力や行動力を育成する。

## 6. 科目の学習目標

- (1) ソフトウェアの開発及びマネジメントのプロセスを理解し、活用することができる。
- (2) ソフトウェアの分析、設計手法を理解し、実際の開発に応用することができる。
- (3) ソフトウェアの特徴を理解し、様々な問題に対して適切に判断し、行動することができる。
- (4) ソフトウェア開発を取り巻く社会環境や今後の動向について説明することができる。

## 7. 本学の教育目標と科目の学習目標との対応

教育目標		学習目標	
高度 ICT スキルの修得	基礎的素養	(1) (2)	
	専門知識および業務応用力	(1) (2)	
人間力 (=探究力) の修得	自ら強みを磨き続ける力	(2)	
	自ら社会における課題を発見し、解決する力	課題設定	(3) (4)
		仮説立案	(3) (4)
		仮説検証	
		実行	
	社会人基礎力	前に踏出す力	(3)
		考え抜く力	(3) (4)
チームで働く力		(1)	
職業倫理の修得		(3) (4)	

## 8. 履修要件

『コンピュータシステム基礎論』を受講していることが望ましい  
『プログラミング特論①』の学習目標に到達していること

## 9. 教科書

なし

## 10. 参考書

書籍名 : ソフトウェア工学の基礎  
著者 : 玉井 哲雄  
出版社 : 岩波書店

## 11. 評価方法と配点

学習目標	達成度評価方法と配点					
	期末試験	小テスト	レポート	発表	成果物	その他
(1)	○	○	○			
(2)	○	○	○	○		
(3)	○	○	○	○		
(4)			○	○		
配点	30	30	20	20		

## 12. 備考

本授業では教科書は利用しませんが、授業資料を Moodle に公開する予定です。

## ■ 授業計画

(注)授業計画は、あくまでも予定であり、実施時に、適時、追加・変更・修正等が生じる場合があります。

---

### 第1回 コンピュータの動作原理 (講義 90分)

---

ソフトウェアはコンピュータの上で動作します。コンピュータがなぜ動くのかを正しく把握しておく、ソフトウェア開発の理解が深まります。本授業ではコンピュータの動作原理を学習します。

1. 本講義の進め方(学習目標、成績評価方法、評価基準など)
2. コンピュータの構成(CPU、メモリ、I/O、クロック)
3. フォン・ノイマン・アーキテクチャ
4. メモリの階層構造
5. プログラムの性能

---

### 第2回 データ構造とアルゴリズムの基礎 (講義 90分)

---

今回はデータ構造とアルゴリズムを取り上げます。プログラムを作成するには、処理の手順(アルゴリズム)と処理の対象となるデータの並べ方(データ構造)と一緒に考える必要があります。本授業では代表的なデータ構造とアルゴリズムを紹介し、データ構造とアルゴリズムの重要性を学びます。

1. アドレス空間と仮想記憶
2. 代表的なデータ構造(配列、リスト、スタックとキュー、木構造)
3. O 記法
4. 代表的なアルゴリズム(整列処理、探索処理)

---

### 第3回 ソフトウェアの特徴とソフトウェア工学 (講義 90分)

---

コンピュータの爆発的な普及に伴い、社会におけるソフトウェアの重要性はますます高まっていますが、ソフトウェア開発は容易ではありません。本授業では、ソフトウェアの特徴を理解した上で、ソフトウェア開発を取り巻く現状と課題を議論し、ソフトウェア工学の必要性を学びます。

1. ソフトウェアの特徴
2. ソフトウェアの重要性
3. ソフトウェア開発を取り巻く環境
4. ソフトウェア工学の役割

---

### 第4回 ソフトウェア開発プロセス (講義 90分)

---

プロセスはプロダクト(製品、成果物)を生み出す工程であり、プロダクトはプロセスの結果として生み出させるものです。従って、良いソフトウェアを開発するためには良いソフトウェアプロセスが必要です。本授業ではソフトウェアのライフサイクルにおけるプロセスモデルを概観し、プロセス改善の意義と狙いを学習します。

1. ソフトウェアプロセスの定義
2. ライフサイクルモデル
3. ソフトウェアプロセスの現状と課題
4. プロセス改善の意義と狙い

---

### 第5回 プロジェクトマネジメント・プロセス (講義 90分)

---

プロジェクトに与えられた使命を達成するために、プロジェクト・チームはさまざまなプロセスを組み合わせ、プロジェクトを推進することになります。本授業では、プロジェクトマネジメントに必要なプロセスを概観し、プロジェクトの立上げから終結まで、どのような流れでプロジェクトが進むのかを学習します。

1. 立上げプロセス・グループ
2. 計画プロセス・グループ
3. 実行プロセス・グループ
4. コントロール・プロセス・グループ
5. 終結プロセス・グループ
6. プロセス・フロー

---

**第6回 要求分析****(講義 90分)**

ソフトウェア技術者の仕事は顧客や利用者の要求をソフトウェアによって実現することですが、顧客や利用者の要求は曖昧で一貫性に欠けることが多く、このままではソフトウェアが開発できません。関係者から要求を獲得し、分析し、明確にする必要があります。本授業では、要求に関する概念を整理し、要求分析の流れを解説します。

1. ニーズ、要求、要件の違い
2. 機能要求、非機能要求
3. 要求分析の流れ
4. 要求分析モデリングの概要

---

**第7回 ソフトウェア設計****(講義 90分)**

良いソフトウェアを作るには良い分析と設計が必要です。どんなに優れたプログラムであってもユーザのニーズが実現できていなければ役に立ちません。また、再利用性や保守性が悪いとビジネスには利用できません。本授業では代表的な設計手法を取り上げ、比較検討を行います。

1. アーキテクチャ設計
2. データフロー・ダイアグラム
3. 構造化設計
4. オブジェクト指向設計

---

**第8回 ソフトウェアテスト****(講義 90分)**

ソフトウェアの不具合による社会への影響が大きくなっているため、ソフトウェアテストの重要性は急速に高まっています。バグがあっても実行されなければ異常は生じないので、完成後に動作試験を行うだけでは不十分です。本授業ではソフトウェア開発におけるテストの位置付け、テストの種類、テスト技法について学びます。

1. テストの必要性と限界
2. ソフトウェア開発の流れとテスト工程
3. ホワイトボックス・テストとブラックボックス・テスト
4. おもなテスト技法

---

**第9回 ソフトウェア品質****(講義 90分)**

ソフトウェア工学を学ぶ目的の一つが高品質なソフトウェアを開発することです。ソフトウェアは目に見えないモノですので、ハードウェア以上に品質に対する活動が重要になります。本授業ではソフトウェアの品質の全体像を理解し、要求される品質を満たすための活動について説明します。

1. 品質とグレードの違い
2. ソフトウェア品質モデル
3. 定量的品質管理

## 4. ソフトウェア・レビュー

**第10回 UML (Unified Modeling Language)****(講義 90分)**

オブジェクト指向技術の普及に伴い、UMLがよく利用されるようになりました。UMLとは Unified Modeling Language (統一モデリング言語)の略で、グラフィカルな記法の一つです。複雑なソフトウェア・システムの記述や設計に便利な道具です。UMLには多くのダイアグラム(図)が用意されていますが、授業ではよく使用される代表的なダイアグラムの書き方を学びます。

1. ユースケース図
2. クラス図
3. 相互作用図
4. ステートチャート図
5. シーケンス図

**第11回 オブジェクト指向技術****(講義 90分)**

ソフトウェアの大規模化、複雑化に伴って、オブジェクト指向技術が普及してきました。オブジェクト指向にはさまざまな特徴がありますが、本授業ではその再利用性、保守性に注目して、オブジェクト指向技術の理解を深めます。

1. 再利用の必要性和難しさ
2. ソフトウェア・コンポーネント
3. フレームワーク
4. デザイン・パターン

**第12回 分析・設計モデリング演習****(演習 90分)**

実際のソフトウェア開発の上流工程がどのように進むのかをイメージするため、グループに分かれて、演習を実施します。例題として与えられたシステムを3つの視点(機能、構造、振る舞い)で整理し、UMLの図を使って表現します。

1. 要求(機能)のモデル化
2. 構造のモデル化
3. 振る舞いのモデル化

**第13回 分析・設計モデリング演習****(演習 90分)**

実際のソフトウェア開発の上流工程がどのように進むのかをイメージするため、グループに分かれて、演習を実施します。例題として与えられたシステムを3つの視点(機能、構造、振る舞い)で整理し、UMLの図を使って表現します。

1. 要求(機能)のモデル化
2. 構造のモデル化
3. 振る舞いのモデル化

**第14回 分析・設計モデリング演習(発表)****(演習 90分)**

前回の授業で作成した分析モデリングの結果をグループ毎に発表し、みんなで意見交換しましょう。自分たちが描いたモデルを客観的に見ることによって多くの気づきが得られると思います。

1. 発表資料の作成
2. 演習結果の発表

---

**第15回 最近の動向****(講義 90分)**

情報通信技術はめざましく進歩しており、それを支えるソフトウェア工学も進化を続けています。本授業ではソフトウェア工学に関連する最近の動向を紹介すると共に、このような高度情報化社会におけるソフトウェア技術者としての倫理観についても議論します。

1. 機能安全と本質安全
2. モデルベース開発と形式手法
3. ソフトウェア技術者の倫理

---

**第16回 期末試験****(試験 90分)**

学習目標の到達度を評価するため、期末試験を実施する予定です。

1. 筆記形式による試験