

# ソフトウェア設計法及び演習 ソフトウェア工学概論及び演習

関澤 俊弦 日本大学 工学部



- ■講義に関して
- ■情報システム
- "ソフトウェア"と"プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)
- ■演習課題



- ■講義に関して
- ■情報システム
- "ソフトウェア"と"プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)
- ■演習課題

### 講義時間と教室



- ■講義時間
  - □月曜3限,5限
- ■教室
  - ロAクラス: 3限 ⋅ 5限 7066教室
  - □Bクラス: 3限 5限 7067教室

### シラバス



#### ■ 教育目標

- □ <u>体系的なソフトウェア開発</u>(開発プロセス, 要件定義, 構造化分析など)の理解
- **ロ ソフトウェア設計過程で用いるモデルの記法と使い** <u>方(DFDやUMLなど)の理解</u>
- ロソフトウェア設計の体得(実際の問題に対してソフトウェア設計手法の適用を行えるようになること)
- ■授業の概要
  - □講義ではソフトウェア設計法と関連領域の解説
  - ロ演習ではノートPCを使用した設計モデルの作成
  - ロ設計演習(レポート)2回と期末試験

## 教科書•参考書



- ■教科書
  - □國友義久著: 効果的プログラム開発技法 第5版, 近代科学社
- ■参考書
  - ロ必要に応じて、講義中に指示する



### 授業計画



- 1. ガイダンス,情報システムの開発
- 2. ソフトウェア開発プロセス
- 3. 構造化分析
- 4. データフローダイアグラム(DFD)
- 5. エンティティリレーションシップダイアグラム(ER図)
- 6. 設計演習1
- 7. オブジェクト指向(1)
- 8. オブジェクト指向(2)
- 9. オブジェクト指向(3)
- 10. 構造化設計
- 11. 設計演習2
- 12. 構造化プログラミング
- 13. テスト技法(1)
- 14. テスト技法(2)
- 15. 期末試験および解説





#### ■留意事項

- □原則として、4回以上欠席した場合、期末試験の 受験を認めない
- □許可なく長時間離席した場合,欠席扱いとする
- □私語が多い場合、退席を求めることがある (欠席扱いとなる)

### 成績評価



- 成績評価 設計演習2回と期末試験で評価する
  - □設計演習(2回) ••• 60%
  - □期末試験 ••• 40%
- ■判定
  - ロ総計60点以上で単位認定



- ■講義に関して
- ■情報システム
- "ソフトウェア"と "プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)

## 情報システムと社会



- ■「情報」を保存・管理するための仕組み
  - □ 通常は、コンピュータ・ネットワーク・制御用ソフト ウェア・運用を含む
- 多くの製品にS/Wが組込まれている
  - ロ生活用品:電子レンジ,炊飯器,...
  - ロ電子機器: MP3プレーヤー, 携帯電話, ...
  - □車載機器

S/Wの重要性・役割の増加



## ソフトウェア工学とは

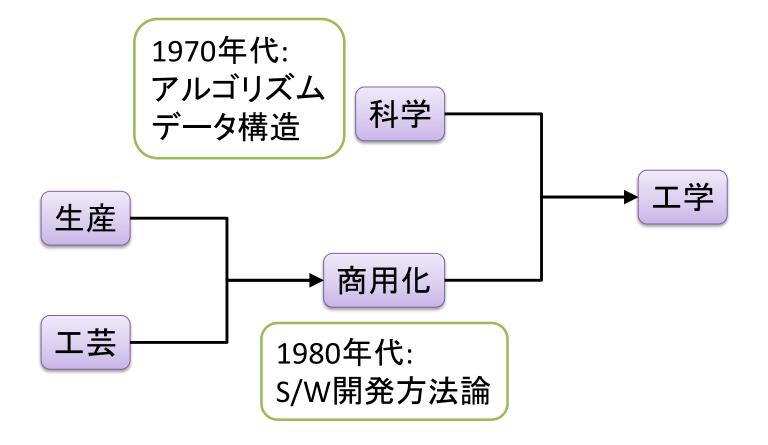


- ■「ソフトウェアエ学」の定義
  - □ IEEE Std 610
  - (1) ソフトウェアの開発,運用,保守に対する,系統的で統制され定量化可能な方法.すなわちソフトウェアへの工学の適用.(2) (1)のような方法の研究.



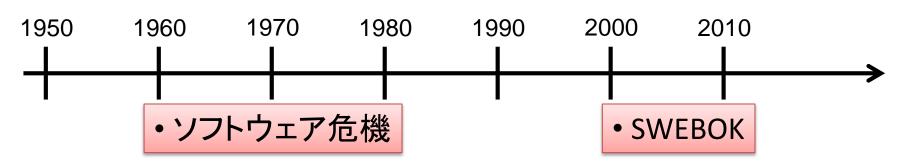


■ソフトウェア工学は発展途上



## N.

## 情報システムの歴史 (1)



- EDVAC報告書の一次稿(1945)
- ENIAC (1946)
- system/360 (1964年)
  - 構造化プログラミング
    - ・ソフトウェアの巨大化
    - ・ソフトウェアプロセス
      - オブジェクト指向
      - インターネットの普及
        - Webサービス技術





■ 2010年以降 ロスマートフォン・タブレット端末









## 情報システム開発の現状



- ■目的の変遷
  - □手作業の機械化から、業務支援、ネットワークへ
- ■電子技術の進化
  - ロスタンドアローンから、ネットワークへ
- ■ネットワーク技術の発展
  - ロネットワークの高速化・大規模化
- ■対象領域の拡大
  - ロ機能中心から利用者中心へ
- ■影響範囲の拡大
  - ロ専門家から一般ユーザへ

## 情報システム開発の問題点



- ■開発期間の短縮
  - ロ品質保証の困難さ
- Webシステム開発技術の未成熟
  - □古典的な開発手法と近年の開発手法の差 (技術的基盤が未だ未確立)
- ■開発技術者の育成
  - □技術者の不足

# N.

## プログラミングに必要とされる(?)才能

- 1970年代にA. P. Ershovによって提唱された
  - ロ第一級の数学者のような論理性
  - ロエジソンのような工学の才能
  - ロ銀行員の正確さ
  - ロ推理作家の発想力
  - ロビジネスマンの実務性
  - ロ協同作業を厭わず経営的な関心も理解する性向



「それは無理・・・」という反応ではなく、知的に取り組むように奮い立たせることが意図.

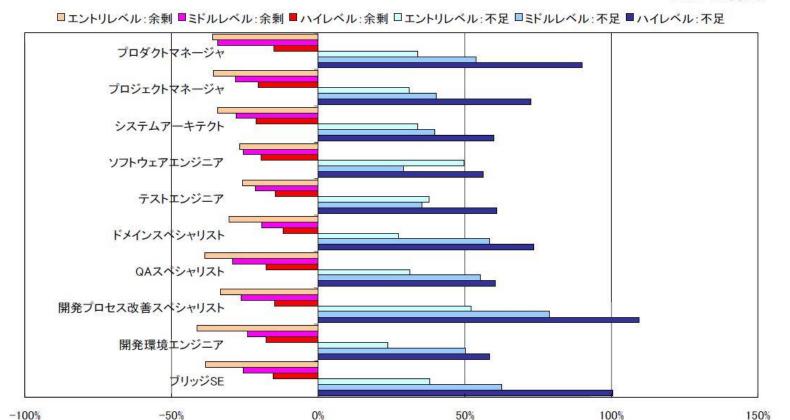






Q4-2Q4-3a 職種別キャリアレベル別現状と最適の比較:全体(社内+社外)

プロジェクト責任者



与経済産業省 Copyright © 2009 Ministry of Economy,

出典:経済産業省 2009年版 組込みソフトウェア産業実態調査



- ■講義に関して
- 情報システム
- "ソフトウェア"と "プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)
- ■演習課題

## "ソフトウェア"とは



この講義は
「ソフトウェア設計法及び演習」
「ソフトウェア工学概論及び演習」
ですが、
"ソフトウェア"とは?





### "ソフトウェア"の語源

■ 米国数学月報に1958年1月号のプリンストン大学の統計学専門のJ・W・Tukeyによる論説が初出[1].

今日では、精細に作られた解釈ルーチン、コンパイラ、その他自動プログラミングからなるソフトウェアは、真空管、トランジスタ、回路、テープなどのハードウェアと少なくとも同程度に、現代の電子計算機(Electronic Calculator)にとって重要である。

[1] 玉井哲雄: "ソフトウェアの語源", bit, Vol.33, No.1(2001), pp.54-57



### 用語: ソフトウェア, プログラム

JIS X0001-1994(情報処理用語-基本用語)

- ソフトウェア (software):
  - ロ"情報処理システムのプログラム, 手続き, 規則 及び関連文書の全体又は一部分."
    - 備考: ソフトウェアは、それを記録した媒体とは無関係な知的創作物である.
- プログラム (program)
  - ロ"アルゴリズムの記述に適した人工言語の規則に 従った構文上の単位であって、ある機能若しくは 仕事の遂行又は問題の解決のために必要な宣 言と文若しくは命令とから構成されるもの。"



- ■講義に関して
- ■情報システム
- "ソフトウェア"と "プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)
- ■演習課題

### ソフトウェア開発作業



- ■ソフトウェア開発作業の特徴
  - 1. 多くのメンバーによる共同作業である
  - 2. 使用者によって多様な操作法が求められる
  - 3. 開発作業中にも要求仕様が変わる

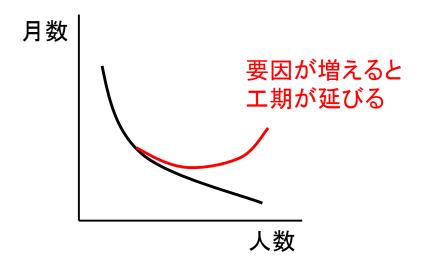


## ソフトウェア開発作業の特徴(1)

- 多くのメンバーによる共同作業である
  - ロBrooksの法則[2]:

「遅れているソフトウェアプロジェクトへの要員追加は、プロジェクトを更に遅らせる」

IBMのOS360開発からの経験則



[2] F. P. Brooks(著), 滝沢徹他(訳): "人月の神話", 丸善出版



## ソフトウェア開発作業の特徴(2)

- ■使用者によって多様な操作法が求められる
  - □ 多用な操作を可能とするためには、多くのプログラムを開発しなければならず、開発作業の労力が増加する?

開発者使用者	1人	多数
開発者本人	1	× 3
他人	× 3	× 9

# N.

## ソフトウェア開発作業の特徴(3)

- 開発作業中にも要求仕様が変わる
  - ロ222の法則:
    - •「計画の2倍の費用と2倍の工期がかかるが、1/2の機能しか実現できない」
  - ロ2423の法則:
    - 契約時の規模・予算は2、要求分析により規模が4に、 3の規模の予算で4の規模のシステムを開発。
      - → ユーザの不満: 予算2→3
      - → 開発者の不満: 予算3で4の規模の開発





■ Brooksの法則や2423の法則などの課題に、 解決策はないのか?



- ソフトウェア設計法 (ソフトウェア開発方法論)
  - ロソフトウェアの作成方法を体系化したもの (本講義で取り上げる主題)



- ■講義に関して
- ■情報システム
- "ソフトウェア"と "プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)
- ■演習課題



### 演習1: astah\* Professional

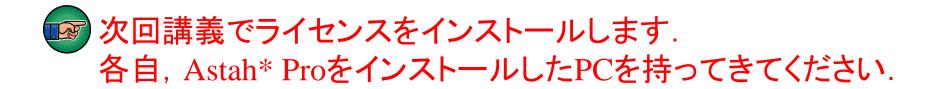
- astah\* Pro.評価版をインストールしておくこと
  - ロAstah\*: システム設計, 開発支援ツール
    - フローチャートやUMLの描画をサポート
    - 本講義の設計ツールとして使用
  - □情報工学科ではastah\* Pro.のサイトライセンスを 保有
    - ・ライセンス期間: 2016年6月25日まで(6月に更新案内)
    - 演習室のiMacにはインストール済み





## 演習1: astah\* Pro.インストール手順

- astah\* Pro.評価版のインストール
- 1. 入手先のURL: <a href="http://astah.change-vision.com/ja/download.html#professional">http://astah.change-vision.com/ja/download.html#professional</a>
- 2. "Astah\* Professional評価版"をダウンロード "50-day Trial"版と記されている
- 3. Astah\* Pro.をインストールする



#### まとめ



- ■講義に関して
- ■情報システム
- "ソフトウェア"と"プログラム"
- ■ソフトウェア開発
  - ロソフトウェア開発の特徴
  - ロソフトウェア設計法(ソフトウェア開発方法論)

次回講義でライセンスをインストールします. 各自, Astah\* ProをインストールしたPCを持ってきてください.