K GEIコンピュータ応用学科

# 「情報処理概論」
# CS1年・後期・必修(2)

担当:浦谷・荒井・行谷
{uratani, arai, yukiya}@cs.t-kougei.ac.jp
http://gakunai.t-kougei.ac.jp/lecture/cs/

K GEIコンピュータ応用学科

## 第2回:システム構築(2)

- システム構築の続き(入力項目など)
- システムの運用と管理の方法について
- 小テスト

K GEIコンピュータ応用学科

## 前回のアンケート結果

- 問3:本日の講義で分かりにくかった箇所は?
  - 説明が早すぎる
  - 専門用語の説明をもっと丁寧に
  - システムテスト・総合テストが分かりにくかった

K GEIコンピュータ応用学科

## システム開発プロセス(復習)

- 要件定義(基本計画):情報システムが満たすべき要件(システム要件仕様)を決める
- 外部設計
- 内部設計
- プログラム開発
- システムテスト
- システム導入

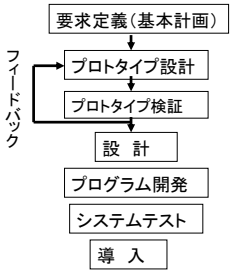
※上記の順に開発を進める開発プロセスを
ウォーターフォール型という



K GEIコンピュータ応用学科

## システム開発プロセスのモデル

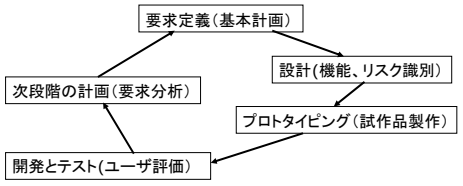
- ウォーターフォール型
- スパイラル型:後述
- プロトタイピング型
  - 全体のフローは右図
  - 試作品をユーザに見せながら開発する
  - 開発方針がユーザの要求と合っているか否かを早い段階で確認できる




K GEIコンピュータ応用学科

## スパイラルモデル

- 各プロセスを螺旋状に巡りながら、システムが次第に成長していくようなモデル
- システムの目標や制約が次々に変化することが予想されるような場合に向く (例えばEコマースシステム)



KOGEIコンピュータ応用学科

## ウォーターフォール方式への反省

- ウォーターフォール方式は建築の手法をまねたもの
  - 建築の世界では最初の厳密な設計が不可欠
  - 「設計」の工程に比べて「作る」工程のほうが圧倒的にコストのかかる大きな作業だから
  - 「設計者」と「施工者」に必要なスキルの違いは明確

しかし、・・・ソフトウェア開発でもそうか？

  - 設計書は建築ほど厳密か？それならコードと変わらない
  - コストの配分は？スキルの違いは？

KOGEIコンピュータ応用学科

## ウォーターフォール方式の問題点

- ドキュメントが中心(システム要件定義書、外部設計仕様書、・・・。記載漏れが許されない)
- ドキュメントの管理が大変
- 不定部分があるとドキュメントのメンテナンスが困難
- リリース(ユーザに引き渡すこと)後のプログラムの修正は仕様変更か？バグの修正か？
- 下流工程で見つかった改善案を上流工程に戻す道がない

30

KOGEIコンピュータ応用学科

## XP手法方式(1)

- Extreame Programming
- ウォーターフォール方式への反省から誕生
  - 従来の手法(重量開発手法)では工程が進んでしまうと、仕様漏れなど手戻りが必要になった時に大変なコストがかかる
  - ソフトウェア開発に伴う変化を、確定させて凍結しようとするのではなく、当然あるべきものとして積極的に対応する

KOGEIコンピュータ応用学科

## XP手法方式(2)

- ドキュメントを重視しない(ほとんど作成しない)
- 顧客を含めたチームでソフトウェア開発
- 2～3週間ごとにチェックできるようにタスク毎(小さなユニット)に開発； スモールリリース
- ストーリーカード：システムをどう使うか
- 受け入れテスト：ある段階までの仕様ができあがっているときにパスしなければならないテスト
- ペア・プログラミング：2人のプログラマが1台のコンピュータ上でプログラミング；1人はチェックのみ

KOGEIコンピュータ応用学科

## 入力項目の定義(外部設計)

- どんな入力項目があって、どんな性質を持っているか、データの量はどれくらいか？
- データをコード化することによるメリット
  - 識別機能：他のデータと識別しやすい
  - 分類機能：データをグループ化しやすい
  - 配列機能：データの並べ替えがしやすい
  - チェック機能：データの入力ミスがわかりやすい
- コード体系
  - ＜例＞：ISBNコード：国際的な書籍管理番号

KOGEIコンピュータ応用学科

## データチェックの方法

- ニューメリックチェック：データは数値か？
  - チェックディジット：誤り検出コード
    元データ12135007 → 格納データ121350079
- リミットチェック：データの値は規定の範囲内か？
- フォーマットチェック：データ形式は正しいか？
- 照合チェック：ほかのファイルや表のデータと照合し、同じコードか否かをチェック
- 重複チェック：同じ値のデータの有無

40

KOGEIコンピュータ応用学科

## 入力画面の設計（外部設計）

- ヒューマンインターフェース：
  - コンピュータと人間の接点（インターフェース）
  - 入力装置（キーボードなど）と出力装置（ディスプレイなど）
- 現在の主流はGUI(Graphical User Interface)
  - マウスでアイコンをクリックなど、初心者にも操作が直感的でわかりやすい
    テキストボックス、ラジオボタン、ポップアップメニュー
  - CUI(Character User Interface)に対して
    CUIはコンピュータ操作に習熟している人向き

KOGEIコンピュータ応用学科

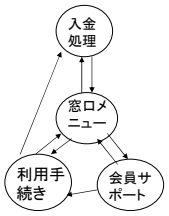
## 入力画面設計のポイント（1）

- 入力の際のミスを起こさせない設計
  - 入力のたびに確認メッセージではだめ！
  - データチェック手法を有効に使おう！
- テンプレートの利用
  - 入力項目は左から右に、上から下に配置
  - 1つの項目を入力し終わったら自動的にカーソルが飛ぶようになっているとベター
  - デフォールト値（初期値）の活用も

KOGEIコンピュータ応用学科

## 入力画面設計のポイント（2）

- 画面を複数に分けることも重要
  - 1画面に無理に全ての入力項目を詰め込むな
  - ファンクションキーで画面を切り替え
    - 画面をスキップしたり、前の画面に戻ることができるように
    - 開発者は画面遷移図を作成して、画面の相互関係を把握しておくことが重要
  - ヘルプ機能をつけておくとさらによい




KOGEIコンピュータ応用学科

## システムの管理（1）

- 管理台帳
  - 機器の1台1台に管理番号を付け、製造番号や設置場所を記入して管理
  - 保守履歴（点検や故障対応）も記入できるようにしておく
  - 建物の平面図も一緒に保管しておくと便利
- 配線図：LANケーブル、電気

※管理台帳をデータベース化してコンピュータで管理すると便利だが、この場合でも万一にそなえて紙の台帳を別にプリントしておく必要がある

50

KOGEIコンピュータ応用学科

## システムの管理（2）

- ヘルプデスクの活用
  - 新しいシステムの導入時に問い合わせに対応する部門を社内に設置して、一括して受け入れる体制を構築
  - しっかりした対応マニュアルを作成しておくことで知識とノウハウが十分ない人でも対応が可
  - SIサービス会社にアウトソーシングも可能

※SI：システム・インテグレーション

KOGEIコンピュータ応用学科

## システムの管理（3）

- 災害対策
  - 免震床：地震による倒壊を防ぐ
  - 高感度の煙センサーと特殊な消化剤：
    人体や電子機器に影響のないように
  - 自家発電装置、UPS（Uninterruptible Power Supply ）
- セキュリティ
  - 盗難、データ改ざん、破壊行為への防止策を
  - 入退室管理、暗証番号によるキーロック
  - コンピュータルームはその表示をしない（建物にも）

KOGEIコンピュータ応用学科

## システム障害に備える(1)

- システム故障に備える
  - フェールソフト(fail soft:縮退運転):
    - システムの一部が故障したとき、一部の機能が使えなくなってもシステムの全面的な停止にならないように設計すること
  - フェールセーフ(fail safe):
    - システムの一部に故障が発生したときに、その影響が安全側に働くようにしておくこと
      (交通信号なら故障時はすべて赤になるように設計)

KOGEIコンピュータ応用学科

## システム障害に備える(2)

- データ損失に備える
  - バックアップ
    - フルバックアップ
    - 差分バックアップ:前回との変更部分だけ
      時間、容量とも節約できるが元に戻すのに時間がかかるため、(更新作業が何回も必要)フルバックアップと使い分けるのが普通
      (1週毎にフルバックアップ、日毎に差分)

00

KOGEIコンピュータ応用学科

## システム障害に備える(3)

- ロールバック
  - 障害が発生した時に、データベースを直前(更新前)のチェックポイントまで戻す処理
  - データ更新中のエラー対応
- ロールフォワード
  - 更新後情報(ジャーナル)を用いて、障害発生時点にできるだけ近い状態に戻す処理
  - データ破損への対応

KOGEIコンピュータ応用学科

## システム障害に備える(4)

- バックアップサイト
  - 地理的に遠く離れた地域のデータセンターをアウトソーシングで利用するのが一般的
  - コールドサイト:建物と場所だけを確保
  - ウォームサイト:バックアップ用のハードは予め設置
  - ホットサイト:待機系サイトとして稼働中のシステムと同じものを準備。長時間の中断が防げる。
  - ミラーサイト:日頃から2つのシステムに全く同じ処理を行わせておき、瞬時に切り替え。

KOGEIコンピュータ応用学科

## システム障害に備える(5)

- RAID(Redundant Arrays of Inexpensive Disks:レイド;ディスクアレイ)の利用
  - RAID-0:ストライピング(複数のハードディスクにデータを分割して書き込む)ので読み書きのスピードアップになるが、安全性は向上しない
  - RAID-1:ミラーリング、デュプレキシング(コントローラも2つ) 利用できる容量は1/2
  - RAID-5:パリティを生成し、それを3台以上のに分散して書き込む 利用できる容量は (n−1)/n

KOGEIコンピュータ応用学科

## デフラグとは?

- ファイルの断片化(フラグメンテーション):
  ファイルの作成・変更・移動・消去を繰り返した結果、まとまった空きスペースがなくなり、1つのファイルがいくつかのスペースに分散して配置されていること
- デフラグメンテーション:
  ファイルの断片化を解消するため、ファイルを整理してディスクの先頭位置から配置し直すこと
  ファイルが断片化されているとシークタイム(ヘッドの移動時間)がかかり処理速度が低下する

10

KOGEIコンピュータ応用学科

## 次回（第3回）は

- プログラム開発について