初級システムアドミニストレータ講座

システム設計

大阪メディカルラボラトリー

佐藤　宏美

CUI (Character User Interface)
プログラム実行などのコマンド入力を、キーボードからの打鍵によって指示する

GUI (Graphical User Interface)
マウスなどを使用し画面上の絵文字 (アイコン) を指示することでコマンド入力する

スクロールバー
スクロール操作で範囲外の部分を表示させる

チェックボックス
処理条件の有効/無効を判別する (複数選択可)

ラジオボタン
複数項目から一つを選択する (***排他的***)

プルダウンメニュー
項目を選択すると、項目に対するサブメニューが階層化して表示される

ポップアップメニュー
右クリックなどの動作によって浮き出てくる形式のメニュー

テキストボックス
文字 (テキスト) を入力するために使用

スピンボタン (スピンエディットボックス)
特定範囲で値を増減させる

平均修復時間（MTTR：Mean Time To Repair）

一回の故障を修理するのに必要な時間の平均

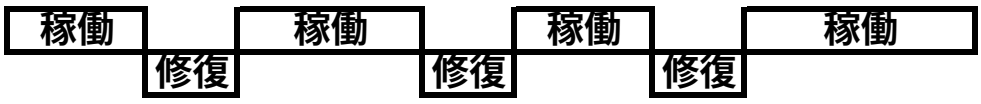
平均故障間隔（MTBF：Mean Time Between Failires）

故障から次の故障までの平均間隔
正常に稼働している時間の平均

稼働率

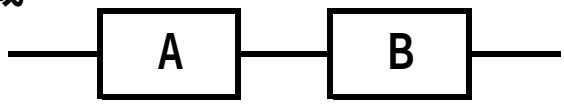
$$\text{稼働率} = \frac{\text{稼働時間}}{\text{全運転時間}} = \frac{\text{全運転時間} - \text{修復時間合計}}{\text{全運転時間}}$$

$$= \frac{MTBF}{MTBF + MTTR}$$



複数系列の稼働率

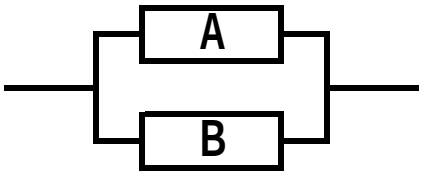
直列編成



A の稼働率 ： a
B の稼働率 ： b

稼働率 ＝ a・b

並列編成



A の稼働率 ： a

B の稼働率 ： b

稼働率 ＝ 1－{(1－a)・(1－b)}

稼働率 ＝ 1－((1－a)・(1－b))

基本計画　ユーザの要求をまとめシステム化の目的を明確にする
（システム計画書･要求仕様書）

外部設計　ユーザインタフェースを中心とした設計
（外部設計書）

内部設計　システム開発側にたった設計
（内部設計書）

プログラム設計　プログラムをモジュール単位に分割する
（プログラム設計書）

プログラム開発　実際のコードに置き換える

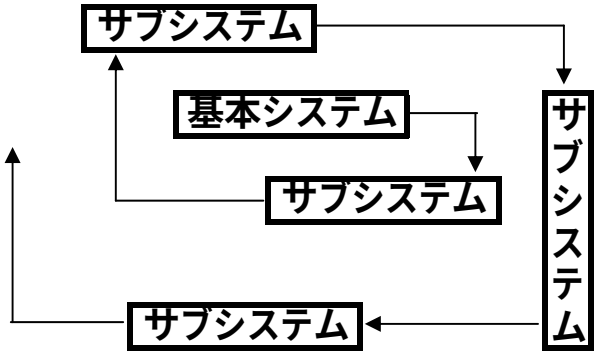
テスト　エラーの発見と修正

運用保守

ウォータフォールモデル
滝の流れのように上流行程から下流行程へと進めていく
後戻りすることなく開発を進めることが原則
大規模な開発に向いている

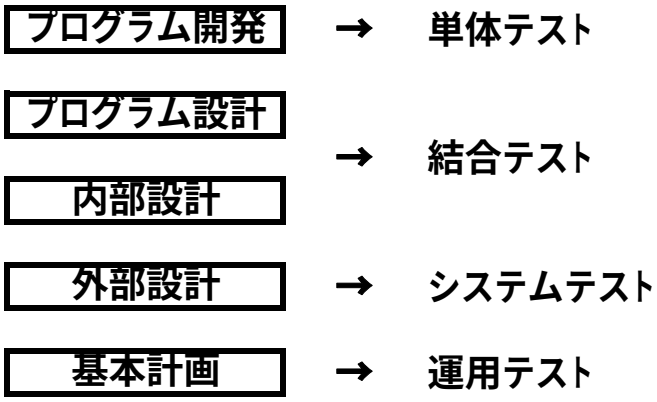
プロトタイピング
システム開発に早い段階でプロトタイプ（***試作品***）を作成
ユーザの要求をフィードバックさせやすい

スパイラルモデル
独立性の高いサブシステムに分割
サブシステム単位に***らせん状***に開発を積み重ねていく



テスト
　作成したシステムが、設計通り作成されているかどうかを検証する行程

バグ(bug)
　「bug」は、虫。コンピュータプログラムに含まれる誤りや不具合のこと
　プログラムの誤りを小さな虫になぞらえたのが語源
　人間が作成する以上、よほど小規模のものでない限り、
　バグのまったくないプログラムを作成するのは不可能である
　このため、ソフトウエアの開発工程ではバグを取り除く作業（デバッグ debug）
　が非常に重要である



単体テスト
　個々のプログラム（モジュール）単位に行われるテスト

　ホワイトボックステスト
　　プログラムの内部構造に着目したテスト
　　プログラム構造の正しさを確認する

　ブラックボックステスト
　　ユーザインタフェースを意識したテスト
　　決められた機能が実施されるかを確認する
　　プログラムの内部構造は意識せず、入力内容と出力結果で検証する

結合テスト
　モジュールを複数組み合わせて実施するテスト

　ボトムアップテスト
　　末端の機能のプログラム（下位モジュール）から順に行うテスト

　トップダウンテスト
　　メニューで呼び出されるプログラム（上位モジュール）から順に行うテスト

システムテスト（総合テスト）
　結合テストの後、開発側主体で行う最終テスト
　シスアドも参加してシステム全体をテストする

　機能テスト
　　システムの利用者側からの要求仕様を満たしているかを評価する

　性能テスト
　　データの処理速度、システムの利用者が行った操作に対する応答時間
　　など、システムの性能を評価する

　負荷テスト
　　大量データや同時処理要求を行って、負荷に耐えられるか評価する

　その他
　　操作性テスト、障害回復テスト、例外処理テスト、耐久テスト

運用テスト
　実際に運用する稼働環境で行うテスト

　承認テスト（受入れテスト）
　　開発側がユーザ側の承認を得るためのテスト
　　シスアドがデータを用意して、要求仕様どおりの結果が得られるかテストする

　導入テスト
　　実際の稼働環境に導入する（現行システムと並行稼働させる場合もある）
　　テストデータも実務で使われているデータ、または実データに近いものを
　　使い、本来の業務担当者が、実際の業務と同じ処理を行って問題がないか
　　テストする

レグレッションテスト（退行テスト）
　システムの修正、変更、機能追加などを行った時、その変更により
　それまで正常に動作していた他のプログラムが影響を受けていないか
　を確認するテスト