

システム開発の方法

第7回

単体テスト

モジュール内の論理構造を検証

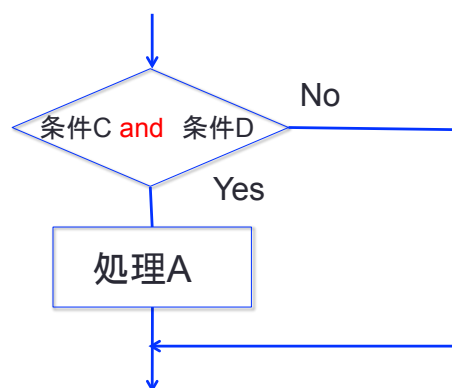
ホワイトボックステスト

モジュール内の内部構造に着目し、プログラムの内部構造や論理が記述された仕様書に基づくテスト

命令網羅

すべての命令(処理)を少なくとも1回実行する

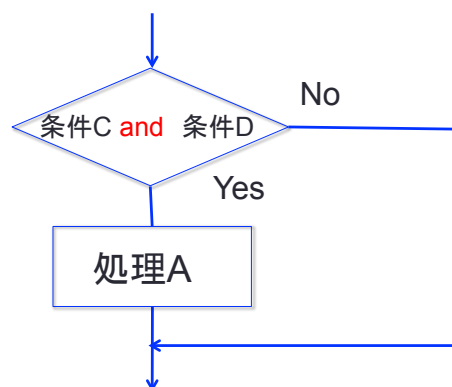
条件Cと条件DがともにYesであるテストデータがあればよい



判定条件網羅(分岐網羅)

すべての分岐を少なくとも1回実行する

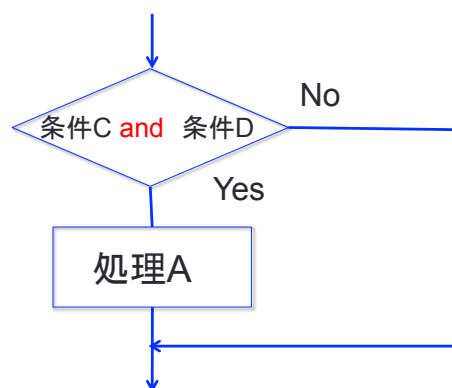
条件全体がYes, Noであるテストデータがそれぞれあればよい



条件網羅

個々の条件がYes,Noの値を少なくとも1回満たすように実行する

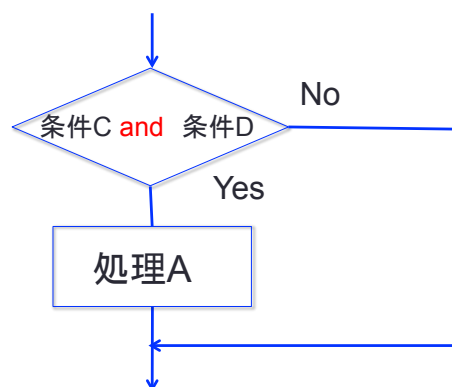
条件Cと条件DがYes,Noになるテストデータがそれぞれあればよい



判定条件／条件網羅

判定条件網羅 + 条件網羅

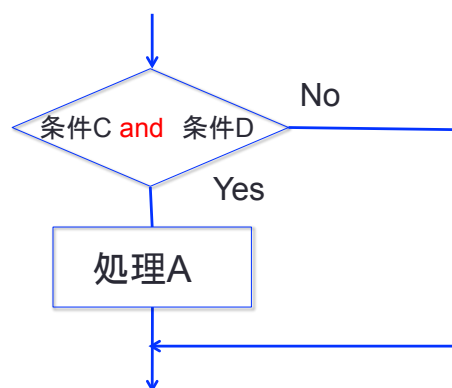
条件Cと条件DがYes,Noになるテストデータがそれぞれあればよい



複数条件網羅

個々の条件がとりうるYes, Noの値のすべての組合せを実行する

条件Cと条件DがYes, Noになるテストデータがそれぞれあればよい。ただし全体がYes, Noになる組合せも必要



結合テスト

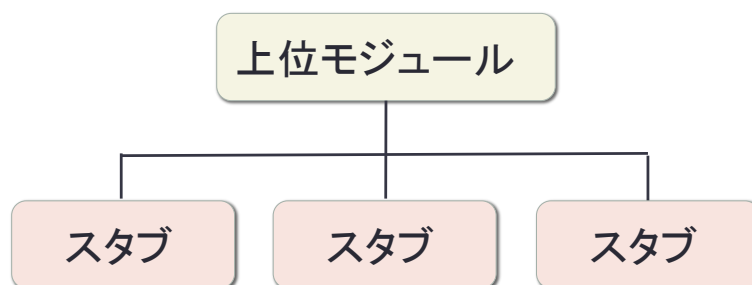
単体テストが終了したモジュールを徐々に結合させてテストする

モジュール集積テスト技法

結合テストを進めていくための技法

トップダウンテスト

上位モジュールからテストをしていく。



下位モジュールのテストが未完の場合は, 仮のモジュールをつけてインタフェースをテストする

トップダウンテストの特徴

- モジュール数の少ない上位部分から開発するので, プログラミングとテストの並行作業が困難
- 重要度の高い上位モジュールが何度もテストでき, 上位モジュールのインタフェースの信頼性が高くなる
- 新規システムの開発にむいている

ボトムアップテスト

下位モジュールからテストをしていく。



テスト未完の上位モジュールの代わりに、仮のモジュールをつけてインタフェースをテストする

ボトムアップテストの特徴

- モジュール数の多い下位部分から開発するので、初期段階からプログラミングとテストの並行作業が可能
- テストの最終段階でモジュール間インタフェースの問題が発見され、モジュールが変更されると、その下位モジュールすべてに影響する
- 既存のシステムを修正して開発する場合にむいている

サンドイッチ(折衷)テスト

- トップダウンテストとボトムアップテストを組合せた手法
- あらかじめ決めたラインの上位をトップダウンテストで, 下位をボトムアップテストで行う

ビッグバンテスト

- すべてのモジュールの単体テストが終わってから全部のモジュールを一気に結合してテストする
- すべてのモジュールを最初からテストできるが, モジュール間インタフェースのエラーの発見が難しく, デバッグが面倒
- 小規模システムのテストで使われる

一斉テスト

- 単体テストを省略し, 必要なモジュールをすべて結合してテストを行う
- 単体テストと結合テストを同時に行うことになるので, エラー発生時の原因特定が難しい
- 小規模システムのテストで使われる

システムテスト

システムとしての要件が満たされているかを確認

サブシステムの結合テスト	サブシステム間のインタフェースをチェック
機能テスト	ユーザから求められたシステム要件を満たしているかをチェック
性能テスト	スループット, レスポンスタイムなどの性能をチェック
操作性テスト	ユーザが操作しやすいかをチェック
障害回復テスト	障害発生への対策が十分か, 回復機能をチェック
負荷テスト	実際の稼働時またはそれ以上の負荷がかかったときのシステム性能をチェック
耐久テスト	長時間の連続稼働に耐えられるかをチェック
例外テスト	システムで制限されているデータの入力に対する対応が万全かをチェック

退行(レグレッション, 回帰)テスト

- システム保守の段階で行われる
- 現在稼働しているシステムの一部を修正したとき, この修正によって他の部分の動作に誤りが発生していないかをテスト

小テスト

「定額給付金」は, 12000円が支給される. ただし 18歳以下と65歳以上には20000円が支給される.

このことについて,

- (1) 原因結果グラフを作成しなさい.
- (2) 決定表(デシジョンテーブル)を作成しなさい.

18歳以下 ①

③ 12000円支給

65歳以上 ②

④ 20000円支給