Tools and Behavioral Abstraction: A Direction for Software Engineering （K. Rustan M. Leino　著）

南山大学大学院　数理情報研究科数理情報専攻

青山研究室所属

M2012MM002

朝倉　知也

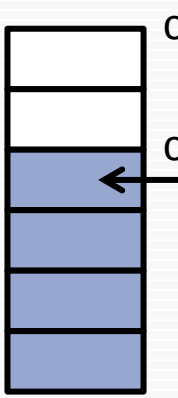
事前条件／事後条件の具体例

- 事前条件/事後条件の記述例(例：スタックのput)

```
indexing
    description:”スタック”
class STACK [G]

～略～

feature  -- 要素の変更
    put (x:G) is
        require
            not full
        do
            -- putルーチンの実装
        ensure
            not empty
            item = x
            count = old count + 1
        end
end
```



full : count = capacity なら true
empty : count = 0 なら true

事前条件

・スタックに空きがある

事後条件

・スタックが空でない
・スタックの一番上にプッシュした値がある
・countはputの実行で1増加した
(old count : put呼出し時のcount)

Bertrand Meyer 著，酒匂寛 訳，オブジェクト指向入門　第2版　原則・コンセプト，翔泳社，2007.

事前条件／事後条件の具体例

```
indexing
    description:”数学計算”
class MathCalc

～略～

feature  -- 平方根の計算
    sqrt (x:real): real is
        require
            x >= 0
        do
            --平方根の計算
        ensure
            Result >= 0
        end
end
```

X<0の処理をしない

- 非冗長性の原則
  「どんな事情があっても、ルーチンの事前条件に
  あたるテストを、ルーチンの本体で行ってはならない。」
  (平方根の計算の場合:『if x < 0 ~』)

  考えうる全ての場合に対するテストを全コンポーネントに記述
  ⇒ 複雑さが増加

- **責任の明確化**
  事前条件は，呼出し側が呼出し時に満たすべき条件
  ⇒ 事前条件違反は呼出し側の責任

  事後条件は，呼出される側が終了時に満たすべき条件
  ⇒ 事後条件違反は呼出される側の責任

Bertrand Meyer 著，酒匂寛 訳，オブジェクト指向入門　第2版　原則・コンセプト，翔泳社，2007.

事前条件／事後条件の具体例

- Javaでは assert を用いて実装可能

```
static double calcSqrt(double x){
    assert x >= 0 : "事前条件違反：入力が負の値です";
    double result = Math.sqrt(x);
    assert result >= 0 : "事後条件違反：出力が負の値です";
    return result;
}
```

✓ 実行例

```
入力値：2.0
計算結果：1.4142135623730951

入力値：-2.0
Exception in thread "main" java.lang.AssertionError: 事前条件違反：入力が負の値です
```