

# プログラムの性質

## プログラムの性質

- 再配置可能（リロケータブル）：メモリのどこに配置しても実行できるという性質
  - どの場所（ロケーション）に配置しても実行できることから
  - 場所：ロケーション 可能：エイブル
  - ベースレジスタを使ったメモリ管理を行うことで再配置が既に可能となっている（らしい）
  - 基本的に言葉の意味だけ理解していれば問題ない
  - 断片化を解消に伴い、プログラムの再配置が行われるなどのシナリオでこの性質が活きる
- 再入可能（リエントラント）：同時に使用されても問題が起きない性質
  - 特定のプログラムを他の複数のプログラムが同時に呼び出しても大丈夫である
    - ◆ 上について補足：同時であれ、そうでなくとも、どのプログラムから呼び出されても正しく動作し、一貫した結果を返す性質を持つ
  - ローカル変数を実装することでこの性質を備えることができる
  - 理由としてはメソッドはその処理が行われると、その内容が解放されるため、複数のプログラムごとの特有の内容でもそれぞれが独立して動作する
  - 総括：リエントラント（reentrant）とは、複数のプログラムやスレッドから同時に呼び出されても正しく動作するメソッドや関数のこと
- 再使用可能（リユーザブル）：読み込んだプログラムを何度も繰り返し利用できる
  - 複数の異なるケースに共通している部分があれば、その部分をケース別で何度も呼び出して利用できる設計のプログラムを設計する
  - 利用されるごとにプログラム内の変数などの初期化が行われている
  - 種類
    - ◆ 逐次再使用可能：同時にプログラムを実行できず、順番にしか行えない：再入不可能
    - ◆ 同時再使用可能：同時にプログラムを実行可能：再入可能
- 再帰（リカーシブ）：自分自身を呼び出す性質