# 再帰関数(recursive function)

直接的または間接的に自分自身を用いて定義されている関数。実行時には自分自身の呼び出し(再帰呼び出し, recursive call)が内部で発生することになる。

例：リストを逆順に並び替える関数 reverse

```
fun reverse(L) =
  if L = nil then nil
  else reverse(tl(L)) @ [hd(L)];
val reverse = fn : ''a list -> ''a list
```

- ''a list：適当な型（実行時の引数により具体的に決定される）を要素とするリスト
- 再帰呼び出しを含まない部分：基底(basis)
- 再帰呼び出しを含む部分：帰納段階(induction step)
    - 再帰呼び出しのときの引数は、自分の引数よりも「小さくする」
        - 整数nが引数なら、n-1を引数にして再帰呼び出し
        - リストLが引数なら、tl(L)を引数にして再帰呼び出し

## 再帰関数の実行

- 原則：関数が呼び出されると、現在の環境で有効な変数の束縛を用いて、式が順次評価されていく
- 以下の2点に注意
    - 引数の束縛は関数が呼び出されたときに作られ、その関数の実行中存在し続ける
    - 同名の識別子に対する束縛が環境中に複数ある場合は、一番上のものが有効

例：reverse([1, 2, 3])

1. 実行前：トップレベル環境(reverseの定義などを含む)
2. reverse([1, 2, 3])
    a. 引数Lと値[1,2,3]の束縛が追加、その下でreverse()の右辺の式を評価
    b. reverse(tl(L)) => reverse([2,3])の評価で再帰呼び出しが発生
3. reverse([2, 3])
    a. 引数Lと値[1,2,3]の束縛が追加、その下でreverse()の右辺の式を評価
4. reverse([3])
    a. reverse(nil) => 返り値nil、すなわちreverse([3])中でのreverse(nil)の値が決定
5. reverse([3]) => [3]
6. reverse([2,3]) => [3, 2]
7. reverse([1,2,3]) => [3, 2, 1]

## 再帰関数をどう実装するか

リストが対象の場合（f(L)を実装するとすると）

- Lが空リストのとき何が得られればよいか考えておく
- f(tl(L))によって正しい結果が得られると仮定する
    - f(tl(L))の結果の値からf(L)の結果を作る方法（アルゴリズム）を考える
    - f(tl(L))の結果の値を f(tl(L)) という式に置き換える

## 2回以上の再帰呼び出しを含む再帰関数

- 例：組み合わせ nCm = (n, m)
    - 公式(n, m) = (n-1, m) + (n-1, m-1)を用いる

```
fun comb(n, m) =
    if m = 0 orelse m = n then 1
    else comb(n-1, m) + comb(n-1, m-1);
```

## 相互再帰(mutual recursion)

- 2つ以上の関数が相互に再帰呼び出しを行う
- 例
    - リストLの奇数番目の要素のみからなるリストを得る関数take(L)
    - リストLの偶数番目の要素のみからなるリストを得る関数skip(L)
- 基底：Lがnilであれば、take(L) = skip(L) = nil
- 帰納段階：take(L) = hd(L) の後ろに skip(tl(L)) の結果をつなげたもの。skip(L) = take(tl(L))
- 実装結果

```
fun take(L) =
    if L = nil then nil
    else hd(L) :: skip(tl(L))
and
    skip(L) =
    if L = nil then nil
    else take(tl(L));
```

- and: 相互再帰関数を定義するときに用いられる記法
    - Cのプロトタイプ宣言のようなものがないので、2つの関数を同時に定義しなければならない

### 型推論の規則

- 算術演算子の演算数および結果は同じ型でなければならない
- 比較演算子の演算数は同じ型でなければならず、結果は bool 型でなければならない
- if E then F else G において、Eはbool型、FとGは同じ型でなければならない
- 関数の仮引数の型と、関数呼び出しの(対応する)実引数の型は同じでなければならない

- 関数の定義式の型と、関数呼び出しの返り値の型は同じでなければならない
- 算術演算子の演算数のデフォルトの型は int

# 練習問題

1. 整数 i とリスト L に対し、L を i 回巡回させる関数。L = [$a_1$, $a_2$, …, $a_n$] のとき [$a_{i+1}$, $a_{i+2}$, …, $a_n$, $a_1$, $a_2$, …, $a_i$]を返す。
2. リストの各要素を複製する関数。[$a_1$, $a_2$, …, $a_n$] に対し[$a_1$, $a_1$, $a_2$, $a_2$, …, $a_n$, $a_n$] を返す。
3. リストの長さ（要素数）を求める関数。