

順列生成

問題

リストを与えると、その要素のすべての並び方を返す関数 junretu を作りなさい。

例: (junretu '(a b c)) → ((a b c) (a c b) (b c a) (b a c) (c b a) (c a b))

解法の一例

リストのある要素に対して、残りの要素の順列並びの先頭に付け加える（この部分は再帰）。この操作を元のリストの全要素に対して行い、結果をくっつける。

たとえば、リスト (a b c) に対しては、a を残りの順列並びの結果 ((b c) (c b)) のそれぞれの先頭に加えて、((a b c) (a c b)) を得る。b に対しては ((a c) (c a)) に加えて、((b a c) (b c a)) を得る。c に対しては ((a b) (b a)) に加えて、((c a b) (c b a)) を得る。この 3 つの結果を合わせて、最終結果が得られる。

くっつける際は、次のような mapappend 関数が必要かもしれないが、作ればよいでしょう。

(mapappend #'(lambda (x) x)'(((a b c) (d e f)) ((g h)))) → ((a b c) (d e f) (g h))

（注）実は mapcan という同様の仕事をしてくれる（破壊的な）関数が存在するので、それを使っててもよい。

(mapcan #'(lambda (x) x)'(((a b c) (d e f)) ((g h)))) → ((a b c) (d e f) (g h))