

練習問題

1. `member` の定義を参考に、リスト `L` の `N` 番目の要素 `X` を求める述語 `nth_element(L,N,X)` を定義せよ。ただし、`N` としてはリストの長さを越える値は考えなくてよい。また、リストの先頭は 0 番目ではなく 1 番目と数える（例：`[a,b,c]` というリストに対しては `N` は 1,2,3 しが入力されないとしてよい。）たとえば `nth_element([a,b,c],2,X)` を実行すると、`X=b` となって成功する。

演習問題 (r4)

以下の問題において、リストの先頭は 0 番目ではなく 1 番目と数える。

* のついている問題はオプションなのでできる者のみ解答せよ。

- (1) リスト `L1` の 1 番目から `N` 番目の要素を順に並べたものがリスト `L2` であるという関係を表す述語 `n_elements(L1,N,L2)` を定義せよ。ただし、`N` としてはリストの長さを越える値は考えなくてよい。たとえば `n_elements([a,b,c],2,L2)` を実行すると、`L2=[a,b]` となって成功する。
- (2) リスト `L` の要素がすべて整数であるとする。`L` の要素で値が 10 以下であるもののすべての和を `S` とするとき、`sum(L,S)` を定義せよ。たとえば `sum([15,2,9],X)` を実行すると、`X=11` となって成功する。
- (3) 定数 `A` をリスト `L1` の最後尾に挿入した結果がリスト `L2` であるような `insert_end(L1,A,L2)` のプログラムを作成せよ。たとえば、`insert_end([a,b],c,L2)` は `L2=[a,b,c]` となって成功する。(Hint: `copy_list` を参考にせよ。)
- (4) リスト `L1` の要素がすべて整数であるとする。`L1` の要素の中でその値が `Min` 以上 `Max` 以下のものすべてから構成したリストが `L2` であるような関係を表す述語 `mid(L1,Min,Max,L2)` のプログラムを作成せよ。たとえば `mid([5,12,2,25,18],10,20,L2)` は `L2=[12,18]` となって成功する。
- (5) リスト `L1` がリスト `L2` の先頭側サブリストかどうかを判定する述語 `prefix(L1,L2)` のプログラムを作成せよ。(サブリストは空リストおよびそれ自身を含む。) たとえば、`prefix([a,b],[a,b,c,d])` は成功する。`L1`、`L2` にはリストしか入力されないと仮定してよい。
- (6)* 練習問題のプログラム `nth_element(L,N,X)` を拡張してリストの長さが `N` に足りない場合は、エラーメッセージを出力するようにせよ。メッセージの出力には組み込み述語 `write` を利用し、`write('The length is less than N')` のように記述する。また、`N` は 1 以上の整数としてよい。
- (7) r4 の練習問題についてレポートせよ。以下を記述すること。(i) プログラムの論理的意味 (ii) `?- nth_element([a,b,c],2,X).` を実行したときの動作。(トレースを貼り付けてはいけない。「ゴール」「実行」「単一化(ユニフィケーション)」という用語をすべて用いてどのゴールとどの節のヘッドが単一化されて変数がどう書き換わり、どのゴールが呼ばれるなどを段階的に記述すること。),(iii) 自分が正しいプログラムができなかった場合、どこが間違ったか、なぜ間違ったかについての考察。正しいプログラムができていた場合は、「正しくできた」と書き、もし新たな知見や疑問があればそれを書く。(特になければ「正しくできた」だけでよい。)