

練習問題

1. 組み込み述語 `member` を使って、リスト `L` が重複する要素をもつかどうかを判定する述語 `has_duplication(L)` のプログラムを作成せよ。 `member(X,Y)` は `X` がリスト `Y` の要素である関係を表す述語である。たとえば、`has_duplication([a,b,a])` は `true` となり、`has_duplication([a,b])` は `false` となる。

演習問題 (r7)

* 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) 組み込み述語 `member` を使って、リスト `L1` から重複している要素をすべて削除した結果がリスト `L2` であるような関係を表す述語 `del_duplication(L1,L2)` のプログラムを作成せよ。ただし、重複している要素がある場合、リストの後の方のものを残し、前の方のものを削除するものとする。たとえば、`del_duplication([a,b,a,a],L2)` は `L2=[b,a]` となって成功する。
- (2) 英語と数学の点数の組を要素とするリスト `L` に対して、(つまり各要素は各学生の成績に相当する) 英語または数学の点数が 60 点未満である人数が `N` であるという関係を表す述語 `failures(L,N)` のプログラムを作成せよ。たとえば、`failures([(55,20),(55,80)],N)` は `N=2`、`failures([(55,80),(60,60),(90,20)],N)` は `N=2` となってそれぞれ成功する。
- (3) 参加者は名前と年齢の組で表される。すべての参加者の名前のリストが `NL` であるという関係を表す述語 `attendants(L,NL)` のプログラムを作成せよ。たとえば、`attendants([(ann,80),(bob,40),(jim,20),(liz,16),(tom,65)],NL)` は `NL = [ann,bob,jim,liz,tom]` となって成功する。
- (4) 名前と年齢の組を要素とするリスト `L` に対して、年齢が 60 歳以上の参加者の名前のリストが `NL` であるという関係を表す述語 `senior(L,NL)` のプログラムを作成せよ。たとえば、`senior([(ann,80),(bob,40),(jim,20),(liz,16),(tom,65)],NL)` は `NL = [ann,tom]` となって成功する。
- (5) 名前と年齢の組を要素とするリスト `L` に対して、年齢が 60 歳以上を `Senior`、20 歳以上 60 歳未満を `Adult`、20 歳未満を `Junior` というリストに要素を分類する述語 `classify(L,Senior,Adult,Junior)` のプログラムを作成せよ。たとえば、`classify([(ann,80),(bob,40),(jim,20),(liz,16),(tom,65)],S,A,J)` は `S = [(ann,80),(tom,65)]`、`A = [(bob,40),(jim,20)]`、`J = [(liz,16)]` となって成功する。(Hint: `r5_3` を参考にせよ。)
- (6)* 命題論理の論理式が以下のように定義されたとする。与えられた表現 `E` が命題論理の論理式であるかどうかを判定する述語 `isFormula(E)` のプログラムを作成せよ。たとえば、`isFormula(and(p,neg(q)))`、`isFormula(imp(and(p,q),r))` は成功し、`isFormula(or(and(p,q)))` は失敗する。

```
Fml ::= Atm | neg(Fml) | and(Fml,Fml) | or(Fml,Fml) | imp(Fml,Fml)
Atm ::= p | q | r
```

- (7) 今回の練習問題 1 の解答例についてレポートせよ。(i) 論理的意味, (ii) `has_duplication([a,b,a])` 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.)