

練習問題

1. 整数の剰余を計算する組込み関数 `mod` を利用して 要素が整数であるリスト `L1` を, 3 で割り切れる要素のリスト `L2` と 3 で割り切れない要素のリスト `L3` に分割する `divide_by_three(L1,L2,L3)` のプログラムを作成せよ. たとえば, `divide_by_three([1,2,3,4,5,6],L2,L3)` は `L2=[3,6],L3=[1,2,4,5]` となって成功する.

演習問題 (r5)

- (1) `are_edges(N,M)` は, 有向グラフにおいてノード `N` を始点とするエッジの終点のリストが `M` であるという関係を表すとする. 図 5.1 において成り立つ `are_edges` をすべて記述してデータベースを作成せよ. (たとえばノード `a` を始点とする場合は `are_edges(a,[b,c])` となる.) 次に, ノード `N` を始点とするエッジの数が `K` 本であるという関係を表す述語 `n_of_edges(N,K)` のプログラムを作成せよ. 図 5.1 だけでなく, 一般のグラフも扱えるようにすること. たとえば, `n_of_edges(a,K)` は `K=2` となって成功する. これら以外にも述語が必要ならそれも定義すること.

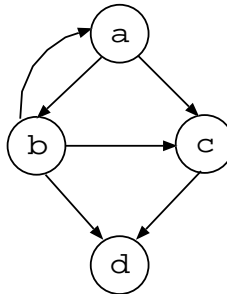


図 5.1

- (2) 有向グラフにおいて, 全ノードのリストを `G`, 各ノード `N` に対して, それを始点とするエッジのリストが (1) のように述語 `are_edges` を使ってデータベースに記述されているとき, このグラフのエッジの数は `E` であるという関係を表す述語 `n_of_all_edges(G,E)` のプログラムを作成せよ. たとえば, 図 5.1 の場合, `n_of_all_edges([a,b,c,d],E)` は `E=6` となって成功する.
- (3) 要素が整数であるリスト `List` の要素を, ある整数 `X` 以上のもののリスト `L` と, `X` 未満のもののリスト `S` に分割することを表す述語 `partition(List,X,L,S)` のプログラムを作成せよ. `List`, `X` の具体値が与えられた時, `L,S` の具体値が求めればよいものとする. たとえば, `partition([4,6,2],5,L,S)` は `L=[6],S=[4,2]` となって成功する.
- (4) 要素が整数である 2 つのリスト `L1`, `L2` と, ある整数 `X` に対して, `L1` と `L2` の間に `X` を挿入したリストが `L` であるという関係を表す述語 `connect_lists(L1,X,L2,L)` のプログラムを, 組み込み述語 `append` を使って作成せよ. たとえば, `connect_lists([1,2],3,[4],L)` は `L=[1,2,3,4]` となって成功する.
- (5) (3) で作成した述語 `partition` および (4) で作成した述語 `connect_lists` を使って, 要素が整数であるリスト `L1` をクイックソートによって昇べきの順 (小さいものから大きなものへの順) に並びかえたリストが `L2` であるという関係を表す述語 `qsort(L1,L2)` のプログラムを作成せよ. ただし, 与えられたリストの最初の要素をピボット (大きい数と小さい数をわけるための基準値) としてとるものとし, `L1` の具体値が与えられた時, `L2` の具体値が求めればよいものとする. たとえば, `qsort([5,4,6,2],L2)` は `L2=[2,4,5,6]` となって成功する.

- (6) r5 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) `?- divide_by_three([1,2,3,4,5,6],L2,L3).` 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．)