

[応用プログラムコース]

練習問題

1. 整数を要素とする (要素の重複を認める) リスト L が小さいものから大きなものへと順番にならんでいる場合は成功し, そうでなければ失敗する述語 $\text{increase}(L)$ のプログラムを作成せよ。

演習問題 (r11)

- (1) 練習問題を参考にして L が等差数列になっているかどうかを判定する述語 $\text{arith_prog}(L)$ のプログラムを作成せよ。ただし, 長さ 2 以下のリストは入力されないとしてよい。たとえば, $\text{arith_prog}([-2,0,2,4,6])$ は成功する。
- (2) L をアトムから成るリストだとする。 L の要素がすべて同一かどうかを判定する述語 $\text{same_elements}(L)$ のプログラムを作成せよ。
- (3) 以下のような木構造が定義されており, 各ノードには異なるラベル a, b, \dots, z が付加されているものとする。このように定義された木 Tree において, 指定されたラベル Label の付加されたノードの深さが Depth であるという関係を表す述語 $\text{node_depth}(\text{Tree}, \text{Label}, \text{Depth})$ のプログラムを作成せよ。ただしルートノードの深さは 1 とする。また, Label には a, b, \dots, z 以外の入力はないと仮定してよい。

```
Tree ::= t(Node, Tree, Tree) | t(Node, Tree) | Node
Node ::= a | b | ... | z
```

たとえば, $\text{node_depth}(t(a, t(b, c, d), t(e, f)), c, \text{Depth})$ は $\text{Depth}=3$ となって成功する

- (4) ex10-4 のプログラムを改訂してループのあるグラフに対しても有効であるような縦型探索を行う $\text{dfs_loop}(\text{Path})$ の繰り返し型プログラムを作成し図 11.1 のグラフに関して全解 (各ノードを高々 1 回しか通らないもののみを解とする) が求まることを確認せよ。たとえば, $\text{dfs_loop}(\text{Path})$ の解の 1 つは $\text{Path}=[g, f, d, c, a, s]$ であり, これを含めて合計 7 個の解答がある。

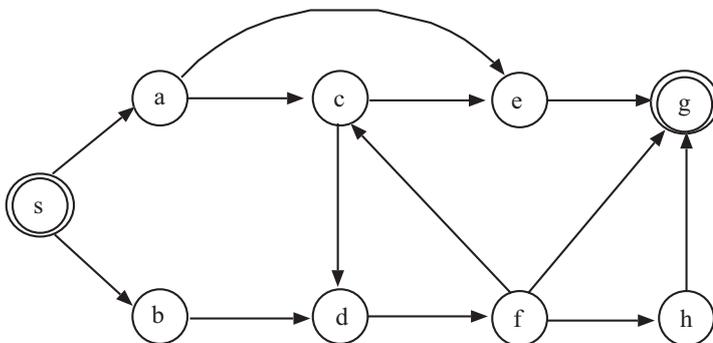


図 11.1

- (5) 表現 Exp が以下のような文法で定義されているとする,

$$\begin{aligned}\text{Exp} &::= \text{Nat} \mid \text{minus}(\text{Exp}) \mid \text{sum}(\text{Exp}, \text{Exp}) \mid \text{diff}(\text{Exp}, \text{Exp}) \\ \text{Nat} &::= 0 \mid 1 \mid 2\end{aligned}$$

与えられた表現を下の書き換え規則を使って diff を除去したものに書き換える.

書き換え規則:(任意の表現 A, B に対して左辺から右辺に書き換える.)

$$[\text{R1}] \text{diff}(A, B) = \text{sum}(A, \text{minus}(B))$$

$$[\text{R2}] \text{minus}(\text{minus}(A)) = A$$

与えられた表現 E をまず書き換え規則 $[\text{R1}]$ を使って書き換え, そのあと $[\text{R2}]$ を使って書き換えた結果が表現 F であるという関係を表す述語 $\text{elim}(E, F)$ のプログラムを作成せよ. ただし, この文法にしたがった表現以外のものは入力されないと仮定してよい. たとえば, $\text{elim}(\text{sum}(\text{diff}(2, \text{minus}(1)), \text{sum}(\text{diff}(1, 2), 1)), A)$ は $A = \text{sum}(\text{sum}(2, 1), \text{sum}(\text{sum}(1, \text{minus}(2)), 1))$ となって成功する.

- (6) レポート問題なし. ダミーとして何か text ファイルをいれてください.

演習問題 (r11b)

r11 で upload してください .

- (1) リスト L の頭から N 個の要素を取り除いたものが $L1$ であるという関係を表す述語 $\text{drop}(L,N,L1)$ のプログラムを作成せよ , ただし , N は 0 以上で L の長さ以下の整数とする . たとえば , $\text{drop}([1,2,3,4,5],2,L1)$ は $L1 = [3,4,5]$ となって成功する .
- (2) 正の整数 (0 は正の数ではない) から成るリスト L の要素で 3 または 5 の倍数の個数が C 個であるという関係を表す述語 $\text{div3or5}(L,C)$ のプログラムを作成せよ . ただし L には重複する要素はないものとする . たとえば , $\text{div3or5}([10,1,4,5,3],C)$ は $C = 3$ となって成功する .
- (3) 正の整数から成るリスト L の要素の平均が MV であるという関係を表す述語 $\text{mean_value}(L,MV)$ のプログラムを作成せよ . たとえば $\text{mean_value}([90,75,72],MV)$ は $MV=79$ となって成功する .

次に , これを使って , あるテストに対する全受験者の英語の点数のリスト E と , 数学の点数のリスト M に対して , 英語の平均点と数学の平均点を求め , 高い方が R であるという関係を表す述語 $\text{compare_value}(E,M,R)$ のプログラムを作成せよ . ただし , R の値は , 英語の平均点が高ければ `english` , 数学の平均点が高ければ `math` , 同点ならば `same` とする . たとえば $\text{compare_value}([90,75,72],[55,68,83],R)$ は $R=\text{english}$ となって成功する . 必要ならば指定されたもの以外の述語も定義すること .

- (4) 以下の文法 (G) が定義されているとき , 与えられた表現 X が式 (Exp) であることを判定する $\text{isExp}(X)$ のプログラムを作成せよ . たとえば $\text{isExp}(\text{plus}(0,\text{suc}(0)))$ は成功し , $\text{isExp}(\text{suc}(\text{plus}(0,0)))$ は失敗する .

(G)

```
Expr ::= plus(Expr,Term) | minus(Expr,Term) | Term
Term ::= 0 | suc(Term)
```

- (5) $T1, T2$ を以下のように定義される項とする . $T1$ が $T2$ の部分項であることを表す述語 $\text{subterm}(A,B)$ のプログラムを作成せよ .

```
Term ::= Alphabet | Digit | f(Term) | g(Term,Term)
Alphabet ::= a | b
Digit ::= 0 | 1
```

たとえば , $\text{subterm}(T1,g(a,f(0)))$ を実行すると , 以下のように複数解が得られる .

```
T1= g(a, f(0));
T1 = a ;
T1 = f(0) ;
T1 = 0
```

- (6) 解答例 r10b_5 についてレポートせよ .