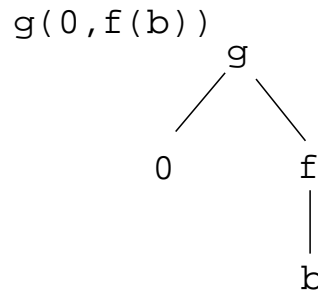


記号処理

Prolog はデータとして、数値だけではなく、項を扱うことができる。

- 変数 (variable) は大文字または '_' で始まる文字列である。
- 定数 (atom) は小文字で始まる文字列である。
- 構造 (structure) は $\text{func}(T_1, \dots, T_n)$ の形をしている。ただし、 func はファンクタと呼ばれる演算子、 T_1, \dots, T_n は項である。
- 項 (term, データオブジェクト) は変数か定数か構造である。

構造の木表現



注意：項はデータ、すなわち引数として扱うものなので、ゴールと混同しないこと。

ゴールは $\text{pred}(T_1, \dots, T_n)$ の形をしている。ただし、 pred は述語記号 (関係)、 T_1, \dots, T_n は項である。

テストデータが複雑な形なので、タイプミスのないようプログラム中に

```
test :- isTerm(g(0,f(b))).
```

と定義しておくなど工夫せよ。

また、デバッグはこのような複雑なデータではなく、もっと単純なデータを使ってエラーの原因を特定すること。

練習問題

- 以下の文法 (\mathcal{G}_1) が定義されているとき、与えられた表現が項 (Term) か否かを判定する $\text{isTm}(X)$ のプログラムを作成せよ。 $\text{isTm}(f(a))$, $\text{isTm}(f(f(b)))$ は成功し、 $\text{isTm}(c)$ は失敗することを確認せよ。 \mathcal{G}_1 はプログラムが対象とする言語であり、プログラミング言語と混同しないように注意。

(\mathcal{G}_1)

```
Term ::= Alphabet | f(Term)
Alphabet ::= a | b
```

- 以下の文法 (\mathcal{G}_2) が定義されているとき、与えられた表現が項 (Term) か否かを判定する $\text{isTerm}(X)$ のプログラムを作成せよ。データ $f(a)$, $g(0,f(b))$, $f(c)$, $g(a)$ について動作確認せよ。

(\mathcal{G}_2)

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

演習問題 (r6)

- (1) 文法 \mathcal{G}_1 が定義されているとき, 項 $T1$ をコピーした結果が項 $T2$ であるような `copy_term(T1,T2)` のプログラムを作成せよ.
- (2) 文法 \mathcal{G}_1 が定義されているとき, 項 $T1$ に出現する a をすべて c に書き換えた結果が項 $T2$ であるような `subst_atoc(T1,T2)` のプログラムを作成せよ. たとえば, `subst_atoc(f(f(a)),X)` は `X=f(f(c))` となって成功し, `subst_atoc(f(f(b)),X)` は `X=f(f(b))` となって (書き換えなしで) 成功する. ただし, $T1$ には文法 \mathcal{G}_1 になかったものしか入力されないものとする.
- (3) 文法 \mathcal{G}_2 が定義されているとき, 項 $T1$ に出現するアトム a をすべて b に書き換えた結果が項 $T2$ であるような `subst_atob(T1,T2)` のプログラムを作成せよ. たとえば, `subst_atob(f(g(1,f(a))),X)` は `X=f(g(1,f(b)))` となって成功する. ただし, $T1$ には文法 \mathcal{G}_2 になかったものしか入力されないものとする.
- (4) 文法 \mathcal{G}_2 が定義されているとき, 与えられたアトム A が与えられた項 $Term$ に出現する回数が C である関係を表す述語 `count_ocr(A,Term,C)` のプログラムを作成せよ. `count_ocr(a,f(g(a,f(a))),C)`, `count_ocr(b,f(g(a,f(a))),C)`, `count_ocr(a,f(g(b,f(a))),C)` についてそれぞれ動作確認せよ. ただし, $T1$ には文法 \mathcal{G}_2 になかったものしか入力されないものとする.
- (5) 文法 \mathcal{G}_2 が定義されているとき, 与えられたアトム A が与えられた項 $Term$ に出現するかどうかを判定する述語 `ocr_check(A,Term)` のプログラムを作成せよ. `ocr_check` のみを用いて (他の述語を使わず) 再帰的に定義すること. `ocr_check(a,f(g(a,f(a))))`, `ocr_check(b,f(g(a,f(a))))`, `ocr_check(a,f(g(b,f(a))))` について動作確認せよ. ただし, $T1$ には文法 \mathcal{G}_2 になかったものしか入力されないものとする.
- (6) r6 の練習問題 2 についてレポートせよ. (i) プログラムの論理的意味 (ii) ?- `isTerm(g(0,f(b)))`. 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.)