

ML演習第3回課題1

飯塚, 後藤, 前田

平成 19 年 6 月 19 日

“Turtle”を表すデータ型と, その turtle を操作する関数群を定義せよ.

解説 turtle はプログラミング言語 LOGO に出てくる二次元平面上を動き回るカメである. このカメは LOGO で記述した指示通りに画面上を動き回り図形を描画する. カメは現在位置と頭の向きを記憶しており, LOGO で与えられた “前に n 歩進め”, “頭を左に k 度回転せよ” などの指示を解釈し動き回る. 今回はこの turtle を表すデータ型を定義して,

- 新しい turtle を (0, 0) の位置に作る.
- turtle を現在の頭の向きに n 歩進ませる.
- turtle を左に k 度回転させる.
- 現在の turtle の位置を返す.

の 4 つの turtle 操作関数を定義してもらおう.

仕様 以下に今回実装すべき関数の型を示す.

```
type turtle = (* any specification is O.K. *)
val new_turtle : unit -> turtle (* 座標 (0, 0) に新しい turtle を作る *)
val advance : turtle -> float -> unit (* turtle を現在の頭の向きに進ませる *)
val rotate : turtle -> float -> unit (* turtle を左に回転させる *)
val locate : turtle -> float * float (* turtle の現在位置を返す *)
```

実行例 turtle を作り, (1.0, 0.0), (1.0, 1.0), (0.0, 1.0), (0.0, 0.0) の各点に移動させる様子を以下に示す.

```
# let t1 = new_turtle ();;
val t1 : turtle = ...
# advance t1 1.0; locate t1;;
```

```
- : float * float = (1., 0.)
# rotate t1 90.0; advance t1 1.0; locate t1;;
- : float * float = (1., 1.)
# rotate t1 90.0; advance t1 1.0; locate t1;;
- : float * float = (0., 1.)
# rotate t1 90.0; advance t1 1.0; locate t1;;
- : float * float = (0., 0.)
(* 計算誤差で正確に (0., 0.) にならないことがあるが, 気にしないでよ
い. *)
```

ヒント

- OCaml では `sin`, `cos` 等の三角関数があらかじめ使えるようになっている (ちなみに, これらの関数で角度はラジアンで表現される)
- 円周率は `atan 1.0 *. 4.0` で表現できる.
- `turtle` のデータ型は変更可能なフィールドを持つレコードを使うといいかもしれない.