

プログラミング言語論 2014年度 第1回小テスト

学籍番号:

氏名:

問1 Little Quilt 言語の以下の各式 (1), (2), (3) が表すキルトを図示せよ。a, b, turn, sew の意味は以下の通りである。その他の Little Quilt 言語の構文要素 (let 式、val 宣言、fun 宣言) の意味は講義で説明したものとする。

- a, b は図1, 図2のキルトを表す。

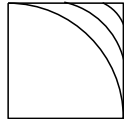


図1: a が表すキルト

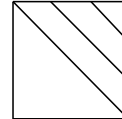


図2: b が表すキルト

- turn (e) : キルト e を 90 度右回転させたキルトを表す。
- sew (e_1, e_2) : 高さが同じキルト e_1, e_2 を左右に並べ、縫い合わせたキルトを表す (左が e_1 、右が e_2)。

(1) sew (turn (turn (b)), a)

```
(2) let
    val x = turn (b)
  in
    sew (x,x)
  end
```

```
(3) let
    fun unturn (x) = turn (turn (turn (x)))
    fun pile (x,y) = unturn (sew (turn (y), turn (x)))
    val aa = pile (a, turn (turn (a)))
    val bb = pile (unturn (b), turn (b))
  in
    sew (aa, bb)
  end
```

問2 命令型言語の制御フローについて以下の各問に答えよ。

(1) 以下のプログラム断片の制御フローを図示せよ。

```
if x>0 then x := x - 1
else if y>0 then y := y - 1
      else y := y + 1
```

(2) 以下のプログラム断片の制御フローを図示せよ。

```
while x>0 do
  begin
    if x=3 then
      begin
        x := x - 1;
        continue
      end;
    y := y + 1;
    x := x - 1
  end
```

問3 以下の Hoare triple(1), (2) を講義中に提示した規則を使って導け。

(1) $\{a = 3\} a := a + 1 \{a = 4\}$

(2) $\{a = 3\} a := a + 1; a := a + 2 \{a = 6\}$

(3) $\{x = 5\} \text{ while } x > 0 \text{ do } x := x - 1 \{x = 0\}$

問4 以下の Pascal プログラムを実行したときの出力結果を示せ。手続きの仮引数に var がついている場合、call by reference であることを表す。writeln は引数の値を出力後改行する。

```
program test;
var x : integer;
var y : integer;
procedure swap
  (var x: integer;
   var y : integer);
var z : integer;
begin
  z := x; x := y; y := z
end;
begin
  x := 3;
  y := 4;
  swap (x,y);
  writeln (x);
  writeln (y)
end.
```

問5 以下の Pascal プログラムを実行したときの出力結果を示せ。Pascal における変数の有効範囲 (scope) の定め方は static scope であることに注意せよ。

```
program P;
var n : char;
procedure W;
begin
  writeln(n)
end;
procedure D;
var n : char;
begin
  n := 'D';
  W
end;
begin
  n := 'L';
  D
end.
```

問6 以下の (1), (2) のプログラムの意味 (状態の変化) を、講義中に提示した規則にしたがって示せ。ただし、これらのプログラムは、講義中で意味の定義を紹介するときに定義した、C の非常に小さなサブセットによるプログラムである。(1)、(2) のプログラムの実行前の状態は、いずれも $\sigma = \{(X, 3), (Y, 1), (Z, 0)\}$ とする。

(1) $Z=(X+4);$

(2) $\text{while}(Y)\{Y=(Y-1);\}$

講義中に提示した規則 Hoare 論理

$$\frac{\{P\} S_1 \{Q\} \quad \{Q\} S_2 \{R\}}{\{P\} S_1; S_2 \{R\}} \text{ (composition rule)}$$

$$\frac{\{P \wedge E\} S_1 \{Q\} \quad \{P \wedge \neg E\} S_2 \{Q\}}{\{P\} \text{ if } E \text{ then } S_1 \text{ else } S_2 \{Q\}} \text{ (conditional rule)}$$

$$\frac{\{P \wedge E\} S \{P\}}{\{P\} \text{ while } E \text{ do } S \{P \wedge \neg E\}} \text{ (while rule)}$$

$$\{Q[E/x]\} x := E \{Q\} \text{ (assignment axiom)}$$

$$\frac{P \Rightarrow P' \quad \{P'\} S \{Q'\} \quad Q' \Rightarrow Q}{\{P\} S \{Q\}} \text{ (consequence rule)}$$

実行の規則

• 式の評価規則

– 数字列の場合 $\langle n, \sigma \rangle \rightarrow m$ (m は数字列 n に対応する整数)

– 変数の場合 $\langle x, \sigma \rangle \rightarrow \sigma(x)$

– 足し算の場合

$$\frac{\langle a_1, \sigma \rangle \rightarrow m_1 \quad \langle a_2, \sigma \rangle \rightarrow m_2}{\langle (a_1 + a_2), \sigma \rangle \rightarrow m} \text{ (} m \text{ は } m_1 \text{ と } m_2 \text{ の和)}$$

– 引き算の場合

$$\frac{\langle a_1, \sigma \rangle \rightarrow m_1 \quad \langle a_2, \sigma \rangle \rightarrow m_2}{\langle (a_1 - a_2), \sigma \rangle \rightarrow m} \text{ (} m \text{ は } m_1 \text{ と } m_2 \text{ の差)}$$

– 掛け算の場合

$$\frac{\langle a_1, \sigma \rangle \rightarrow m_1 \quad \langle a_2, \sigma \rangle \rightarrow m_2}{\langle (a_1 * a_2), \sigma \rangle \rightarrow m} \text{ (} m \text{ は } m_1 \text{ と } m_2 \text{ の積)}$$

• 文の実行に関する規則

– 代入文の場合

$$\frac{\langle a, \sigma \rangle \rightarrow m}{\langle x = a; , \sigma \rangle \rightarrow \sigma[m/x]}$$

ただし、 $\sigma[m/x]$ は以下で定義される。

$$(\sigma[m/x])(y) = \begin{cases} m & \text{if } y = x \\ \sigma(y) & \text{if } y \neq x \end{cases}$$

– 文の並びの場合

$$\frac{\langle c_1, \sigma \rangle \rightarrow \sigma_1 \quad \langle c_2, \sigma_1 \rangle \rightarrow \sigma_2}{\langle c_1 \ c_2, \sigma \rangle \rightarrow \sigma_2}$$

– while 文の場合

$$\frac{\langle a, \sigma \rangle \rightarrow m \quad \langle c, \sigma \rangle \rightarrow \sigma_1 \quad \langle \text{while } (a) \{c\}, \sigma_1 \rangle \rightarrow \sigma_2}{\langle \text{while } (a) \{c\}, \sigma \rangle \rightarrow \sigma_2} \text{ (if } m \neq 0)$$