

プログラミング言語論 第1回小テスト 解答例

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。

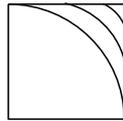


図1: a が表すキルト

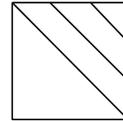
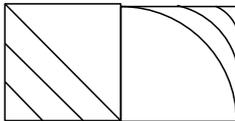


図2: b が表すキルト

- turn (e) : キルト e を 90 度右回転させたキルトを表す。
- sew (e₁, e₂) : 高さが同じキルト e₁, e₂ を左右に並べ、縫い合わせたキルトを表す (左が e₁、右が e₂)。

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



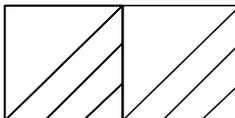
(2) let

```
    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)))
```

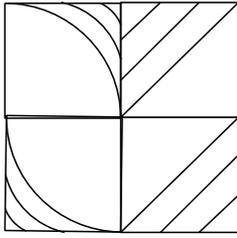
```
    val aa = pile (a, turn (turn (a)))
```

```
    val bb = pile (unturn (b), turn (b))
```

```
in
```

```
    sew (aa, bb)
```

```
end
```



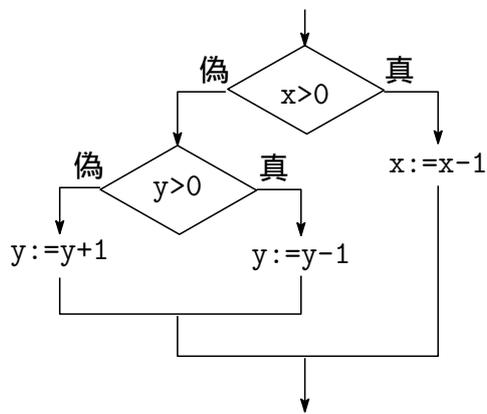
問2 命令型言語の制御フローについて以下の各問に答えよ。

(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

```

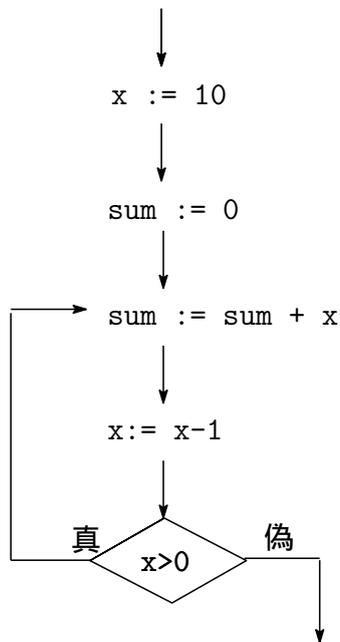


(2) 以下のプログラム断片の制御フローを図示せよ。

```

x := 10;
sum := 0;
L: sum := sum + x;
x := x - 1;
if x>0 then
    goto L

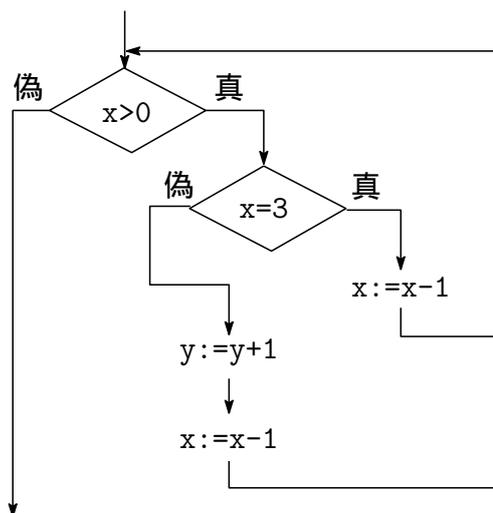
```



(3) 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
end
  
```

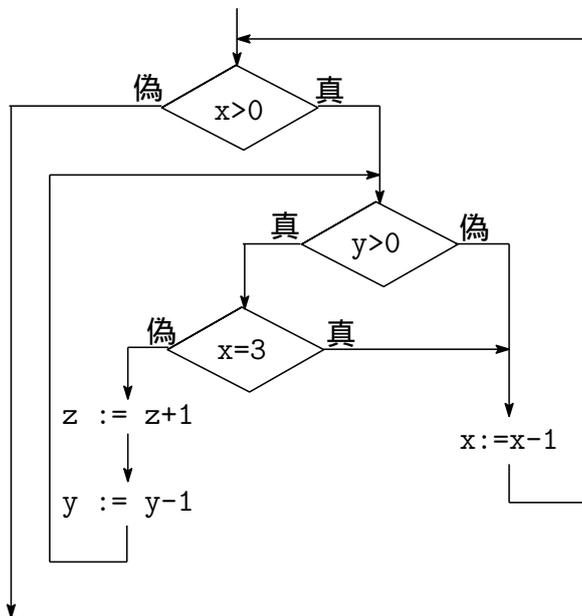


(4) 以下のプログラム断片の制御フローを図示せよ。

```

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

```



(5) (4) のプログラム断片における if 文 (if x=3 then break;) の入口、出口はそれぞれいくつか答えよ。

入口1つ、出口2つ

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

$$\frac{a = 3 \Rightarrow a + 1 = 4 \quad \frac{}{\{a + 1 = 4\} a := a + 1 \{a = 4\}} \text{(assign)} \quad a = 4 \Rightarrow a = 4}{\{a = 3\} a := a + 1 \{a = 4\}} \text{(conseq)}$$

授業時にも言った通り、上記証明の中で、 $a = 4 \Rightarrow a = 4$ の部分は \Rightarrow の左右が同じなので省略し、以下のように書いてもよい。

$$\frac{a = 3 \Rightarrow a + 1 = 4 \quad \frac{}{\{a + 1 = 4\} a := a + 1 \{a = 4\}} \text{(assign)}}{\{a = 3\} a := a + 1 \{a = 4\}} \text{(conseq)}$$

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

$$\frac{\frac{a = 3 \Rightarrow a + 1 = 4 \quad \overline{\{a + 1 = 4\} a := a + 1 \{a = 4\}} \text{ (assign)}}{\{a = 3\} a := a + 1 \{a = 4\}} \text{ (conseq)} \quad \frac{a = 4 \Rightarrow a + 2 = 6 \quad \overline{\{a + 2 = 6\} a := a + 2 \{a = 6\}} \text{ (assign)}}{\{a = 4\} a := a + 2 \{a = 6\}} \text{ (conseq)}}{\{a = 3\} a := a + 1; a := a + 2 \{a = 6\}} \text{ (composition)}$$

この証明では、2カ所の consequence rule の適用のところで、 $a = 4 \Rightarrow a = 4$ と $a = 6 \Rightarrow a = 6$ を省略した。

(3) $\{a = 4\} \text{ if } a = 4 \text{ then } a := a + 2 \text{ else } a := a - 3 \{a = 6\}$

スペースの関係で、3つに分けて記述する。

$$\frac{\frac{\text{この部分は下に記述}}{\{a = 4 \wedge a = 4\} a := a + 2 \{a = 6\}} \text{ (conseq)} \quad \frac{\text{この部分は下に記述}}{\{a = 4 \wedge \neg a = 4\} a := a - 3 \{a = 6\}} \text{ (conseq)}}{\{a = 4\} \text{ if } a = 4 \text{ then } a := a + 2 \text{ else } a := a - 3 \{a = 6\}} \text{ (conditional)}$$

$$\frac{a = 4 \wedge a = 4 \Rightarrow a + 2 = 6 \quad \overline{\{a + 2 = 6\} a := a + 2 \{a = 6\}} \text{ (assign)}}{\{a = 4 \wedge a = 4\} a := a + 2 \{a = 6\}} \text{ (conseq)}$$

$$\frac{a = 4 \wedge \neg a = 4 \Rightarrow a - 3 = 6 \quad \overline{\{a - 3 = 6\} a := a - 3 \{a = 6\}} \text{ (assign)}}{\{a = 4 \wedge \neg a = 4\} a := a - 3 \{a = 6\}} \text{ (conseq)}$$

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

スペースの関係で、2か所に分けて記述する。

$$\frac{a = 5 \Rightarrow a \geq 0 \quad \frac{\text{この部分は下に記述。}}{\{a \geq 0\} \text{ while } a > 0 \text{ do } a := a - 1 \{a \geq 0 \wedge \neg a > 0\}} \quad a \geq 0 \wedge \neg a > 0 \Rightarrow a = 0}{\{a = 5\} \text{ while } a > 0 \text{ do } a := a - 1 \{a = 0\}} \text{ (conseq)}$$

$$\frac{a \geq 0 \wedge a > 0 \Rightarrow a - 1 \geq 0 \quad \overline{\{a - 1 \geq 0\} a := a - 1 \{a \geq 0\}} \text{ (assign)}}{\{a \geq 0 \wedge a > 0\} a := a - 1 \{a \geq 0\}} \text{ (conseq)} \quad \frac{\{a \geq 0 \wedge a > 0\} a := a - 1 \{a \geq 0\}}{\{a \geq 0\} \text{ while } a > 0 \text{ do } a := a - 1 \{a \geq 0 \wedge \neg a > 0\}} \text{ (while)}$$

この下側の証明図でも、 $a \geq 0 \Rightarrow a \geq 0$ の部分は \Rightarrow の左右が同じなので、省略して以下のように書いてよい。

$$\frac{a \geq 0 \wedge a > 0 \Rightarrow a - 1 \geq 0 \quad \overline{\{a - 1 \geq 0\} a := a - 1 \{a \geq 0\}} \text{ (assign)}}{\{a \geq 0 \wedge a > 0\} a := a - 1 \{a \geq 0\}} \text{ (conseq)} \quad \frac{\{a \geq 0 \wedge a > 0\} a := a - 1 \{a \geq 0\}}{\{a \geq 0\} \text{ while } a > 0 \text{ do } a := a - 1 \{a \geq 0 \wedge \neg a > 0\}} \text{ (while)}$$

上記証明において、assignment axiom を assign, consequence rule を conseq, while rule を while, composition rule を composition, conditional rule を conditional と略記した。