## Notes on solutions to Exercise 4 Graph colouring

Compilers

Paul Kelly

Imperial College London

```
S1:
      A = 100;
P1:
S2:
      B = 200;
P2:
      C = A + B;
S3:
P3:
      D = A * 2;
S4:
P4:
      E = B * 2;
S5:
P5:
S6:
   F = D - C;
P6:
S7:
      G = E + F;
P7:
```

```
S1: A = 100;
P1:
      B = 200;
S2:
P2:
      C = A + B;
S3:
P3:
      D = A * 2;
S4:
P4:
      E = B * 2;
S5:
P5:
S6:
   F = D - C;
P6:
S7:
      G = E + F;
P7:
```

```
S1: A = 100;
P1:
S2:
    B = 200;
P2:
      C = A + B;
S3:
P3:
      D = A * 2;
S4:
P4:
      E = B * 2;
S5:
P5:
S6:
      F = D - C;
P6:
S7:
      G = E + F;
P7:
```

```
S1: A = 100;
P1:
S2:
   B = 200;
P2:
      C = A + B;
S3:
P3:
      D = A * 2;
S4:
P4:
S5:
      E = B * 2;
P5:
S6:
      F = D - C;
P6:
S7:
      G = E + F;
P7:
```

```
S1: A = 100;
P1:
S2:
   B = 200;
P2:
     C = A + B;
S3:
P3:
      D = A * 2;
S4:
P4:
      E = B * 2;
S5:
P5:
     F = D - C;
S6:
P6:
      G = E + F;
S7:
P7:
```

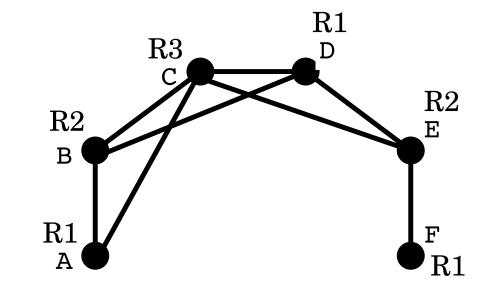
```
S1: A = 100;
P1:
S2:
   B = 200;
P2:
S3: C = A + B;
P3:
S4: D = A * 2;
P4:
     E = B * 2;
S5:
P5:
     F = D - C;
S6:
P6:
     G = E + F;
S7:
P7:
```

```
S1: A = 100;
P1:
S2:
    B = 200;
P2:
      C = A + B;
                     В
S3:
P3:
      D = A * 2;
S4:
P4:
                                   G
      E = B * 2;
S5:
P5:
S6:
      F = D - C;
P6:
      G = E + F;
S7:
P7:
```

E

F

```
S1: A = 100;
P1:
                                     R1
                           R3
S2:
      B = 200;
                                     D
                                           R2
P2:
                      R2
                                           E
      C = A + B;
S3:
                      В
P3:
                      R1
A
                                           F
       D = A * 2;
S4:
P4:
                                     G
S5:
       E = B * 2;
                                     R1
P5:
S6:
       F = D - C;
P6:
       G = E + F;
S7:
P7:
```



G

R1

S1: 
$$A = 100$$
  $R1 = 100$ 

S2: B = 200 R2 = 200

S3: C = A + B R3 = R1 + R2

S4: D = A \* 2 R1 = R1 \* 2

S5: E = B \* 2 R2 = R2 \* 2

S6: F = D - C R1 = R1 - R3

S7: G = E + F R1 = R2 + R1

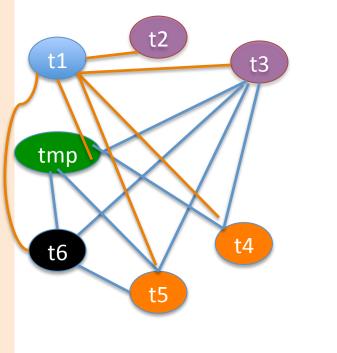
```
(i coexists with everything
           t1 := size-1 j coexists with everything
                              k coexists with everything)
           k := 0
                              t1 coexists with everything
                              t2 coexists with t1 (and i,j,k but asked to ignore them).
L1:
                              t3 coexists with t1, tmp, t4, t5, t6
                              tmp coexists with t1,t3,t4,t5,t6
           cmp k,t1
                              t4 coexists with t1,tmp,t3
                              t5 coexists with t1,t6,t3,tmp
           bgt End
                              t6 coexists with t1,t3,tmp,t5
           t2 := Address(A) + i
           t3 := t2+k
           tmp := LoadIndirect(t3)
           t4 := Address(A) + j
           t5 := t4+k
           t6 := LoadIndirect(t5)
           StoreIndirect(t6, t3)
           StoreIndirect(tmp, t5)
           k := k+1
           jmp L1
End:
```

```
(i coexists with everything
           t1 := size-1 j coexists with everything
                              k coexists with everything)
           k := 0
                              t1 coexists with everything
                              t2 coexists with t1 (and i,j,k but asked to ignore them).
L1:
                              t3 coexists with t1, tmp, t4, t5, t6
                              tmp coexists with t1,t3,t4,t5,t6
            cmp k,t1
                              t4 coexists with t1,tmp,t3
                              t5 coexists with t1,t6,t3,tmp
           bgt End
                              t6 coexists with t1,t3,tmp,t5
           t2 := Address(A) + i
           t3 := t2+k
                                                          t2
           tmp := LoadIndirect(t3)
                                                  t1
           t4 := Address(A) + j
            t5 := t4+k
                                                 tmp
           t6 := LoadIndirect(t5)
           StoreIndirect(t6, t3)
                                                  t6
            StoreIndirect(tmp, t5)
                                                          t5
            k := k+1
            jmp L1
End:
```

t3 t4

t3

```
(i coexists with everything
           t1 := size-1 j coexists with everything
                              k coexists with everything)
           k := 0
                              t1 coexists with everything
                              t2 coexists with t1 (and i,j,k but asked to ignore them).
L1:
                              t3 coexists with t1, tmp, t4, t5, t6
                              tmp coexists with t1,t3,t4,t5,t6
            cmp k,t1
                              t4 coexists with t1,tmp,t3
                              t5 coexists with t1,t6,t3,tmp
           bgt End
                              t6 coexists with t1,t3,tmp,t5
            t2 := Address(A) + i
           t3 := t2+k
           tmp := LoadIndirect(t3)
                                                  t1
           t4 := Address(A) + j
            t5 := t4+k
            t6 := LoadIndirect(t5)
           StoreIndirect(t6, t3)
                                                 t6
            StoreIndirect(tmp, t5)
            k := k+1
            jmp L1
```



End:

```
(i coexists with everything
          t1 := size-1 j coexists with everything
                            k coexists with everything)
          k := 0
                            t1 coexists with everything
                            t2 coexists with t1 (and i,j,k but asked to ignore them).
L1:
                            t3 coexists with t1, tmp, t4, t5, t6
                            tmp coexists with t1,t3,t4,t5,t6
           cmp k,t1
                           t4 coexists with t1,tmp,t3
                            t5 coexists with t1,t6,t3,tmp
          bgt End
                            t6 coexists with t1,t3,tmp,t5
          t2 := Address(A)+i
           t3 := t2+k
          tmp := LoadIndirect(t3)
                                              RO: t1
          t4 := Address(A)+j
                                              R1: t2, t3
          t5 := t4+k
           t6 := LoadIndirect(t5)
                                              R2: t4,t5
           StoreIndirect(t6, t3)
                                              R3: t6
           StoreIndirect(tmp, t5)
           k := k+1
                                              R4: tmp
           jmp L1
End:
```