

Solutions to Written Assignment 9

1. Optimize the following code, using global constant propagation:

```

b := 1
d := 4

top:
  a := 3 // b+2
  c := 4 // a+b
  if p goto L

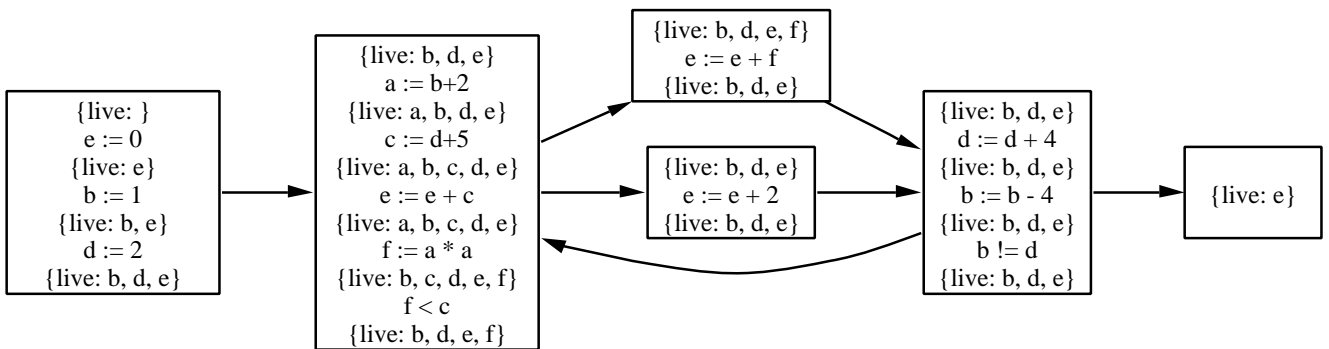
  d := 2*d
  if q goto L2
  goto end

L:
  d := d+1
  c := 3 // c-1

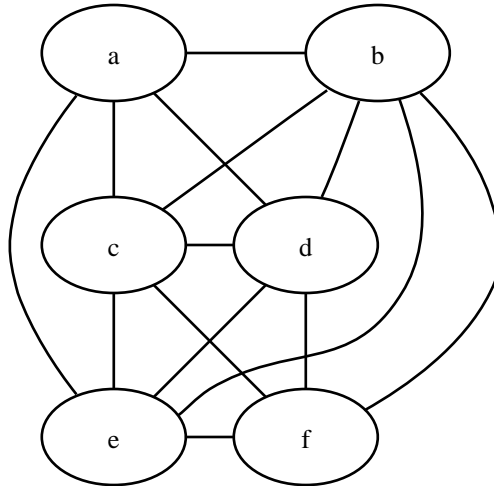
L2:
  c := d+1
  if q goto top
  goto end

end:
  print 3 // a
  
```

2. (a) It is convenient to first draw a control-flow graph and compute the live variable sets:



The register interference graph is then easy to compute:



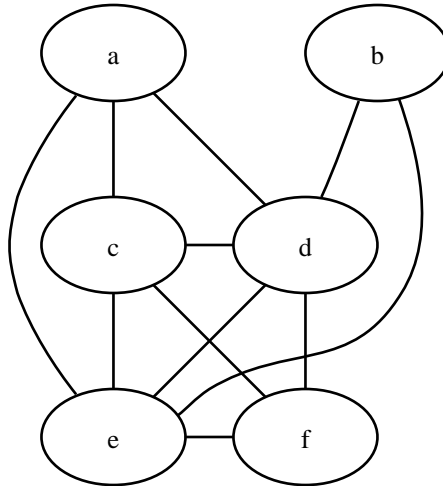
(b) Now we run the register allocation heuristics. Since we have only 4 registers, we have to pick some temporary to spill. Arbitrarily, we choose to spill b. Our code becomes:

```

L0: e := 0
    b := 1
    store b
    d := 2
L1: load b
    a := b+2
    c := d+5
    e := e + c
    f := a * a
    if f < c goto L3
L2: e := e + f
    goto L4
L3: e := e + 2
L4: d := d + 4
    load b
    b := b - 4
    store b
    load b /* You might omit this */
    if b != d goto L1
L5:

```

and the new register interference graph is



This can be 4-colored by our register allocation heuristics. One possible assignment of registers to temporaries is b and c to r1, d to r2, e to r3, and a, f to r4.

Our code becomes

```

L0: r3 := 0
    r1 := 1
    store r1
    r2 := 2
L1: load r1
    r4 := r1+2
    r1 := r2+5
    r3 := r3 + r1
    r4 := r4 * r4
    if r4 < r1 goto L3
L2: r3 := r3 + r4
    goto L4
L3: r3 := r3 + 2
L4: r2 := r2 + 4
    load r1
    r1 := r1 - 4
    store r1
    load r1 /* You might omit this */
    if r1 != r2 goto L1
L5:

```