

COMMENTS:

- CP.register_allocation
- overlaps MQ_
- from: f01.ps4-4 revised from Rosen 7.8.21, S07.cp6m; S09.cp6r

keywords = [*simple_graph coloring scheduling conflict*]

Problem 1.

A portion of a computer program consists of a sequence of calculations where the results are stored in variables, like this:

| | | |
|---------|----------|-----------|
| | Inputs: | a, b |
| Step 1. | c | $= a + b$ |
| 2. | d | $= a * c$ |
| 3. | e | $= c + 3$ |
| 4. | f | $= c - e$ |
| 5. | g | $= a + f$ |
| 6. | h | $= f + 1$ |
| | Outputs: | d, g, h |

A computer can perform such calculations most quickly if the value of each variable is stored in a *register*, a chunk of very fast memory inside the microprocessor. Programming language compilers face the problem of assigning each variable in a program to a register. Computers usually have few registers, however, so they must be used wisely and reused often. This is called the *register allocation* problem.

In the example above, variables a and b must be assigned different registers, because they hold distinct input values. Furthermore, c and d must be assigned different registers; if they used the same one, then the value of c would be overwritten in the second step and we’d get the wrong answer in the third step. On the other hand, variables b and d may use the same register; after the first step, we no longer need b and can overwrite the register that holds its value. Also, f and h may use the same register; once $f + 1$ is evaluated in the last step, the register holding the value of f can be overwritten.

(a) Recast the register allocation problem as a question about graph coloring. What do the vertices correspond to? Under what conditions should there be an edge between two vertices? Construct the graph corresponding to the example above.

2

(b) Color your graph using as few colors as you can. Call the computer’s registers $R1$, $R2$, etc. Describe the assignment of variables to registers implied by your coloring. How many registers do you need?

(c) Suppose that a variable is assigned a value more than once, as in the code snippet below:

```
...  
 $t = r + s$   
 $u = t * 3$   
 $t = m - k$   
 $v = t + u$   
...
```

How might you cope with this complication?