Exercises on Compilers

Jordi Cortadella

February 7, 2022

Code optimization

1. Consider the following basic block:

```
g = x + y
h = u - v
i = x + y
x = u - v
u = g + h
v = i + x
w = u + v
```

   - Build the DAG representation.
   - Generate code under the assumption that the set of live variables is $\{g, h, u, w\}$.
   - Generate code under the assumption that the set of live variables is $\{u, h, x\}$.

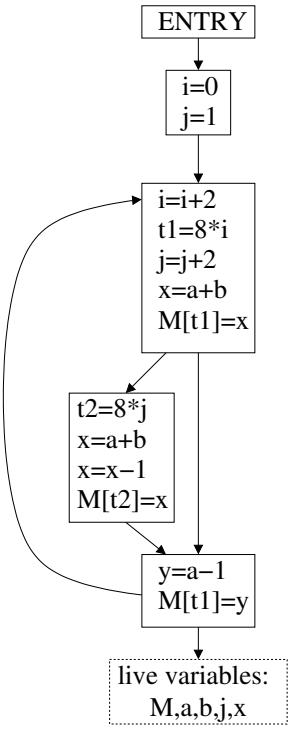
2. Consider the following code:

```
1:   m = 0
2:   v = 0
3:   t = x+4
4:   if v >= n goto 18
5:   r = v
6:   s = 0
7:   k = 0
8:   k = k+1
9:   z = k*4
10:  q = M[z]
11:  s = s+q
12:  if s <= m goto 14
13:  m = s
14:  r = r+1
15:  if r < n goto 8
16:  v = v+1
17:  goto 3
18:  return m
```

   - Calculate the basic blocks and draw the associated control-flow graph.
   - Calculate the tree of dominators.
   - Identify the back edges and loops of the control-flow graph.
   - Perform loop optimizations (assume that all variables are alive at the exit of the code).

3. Consider the code in the figure:

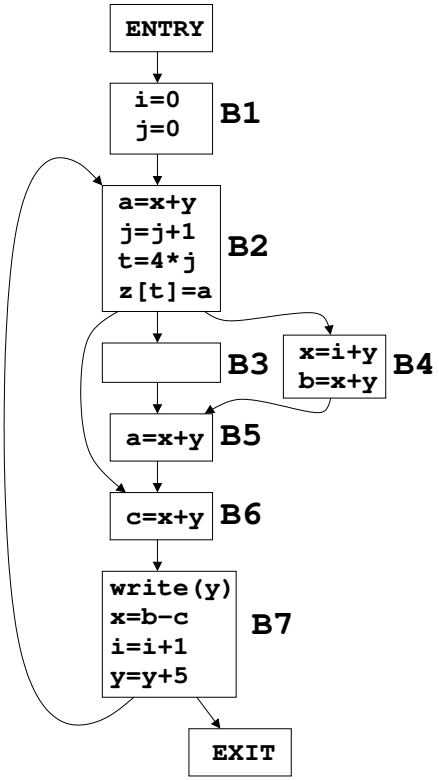
   - Apply all possible optimizations considering the live variables specified at the exit of the code.
   - Show the final code after the optimizations and explain the transformations produced by each optimization.



4.

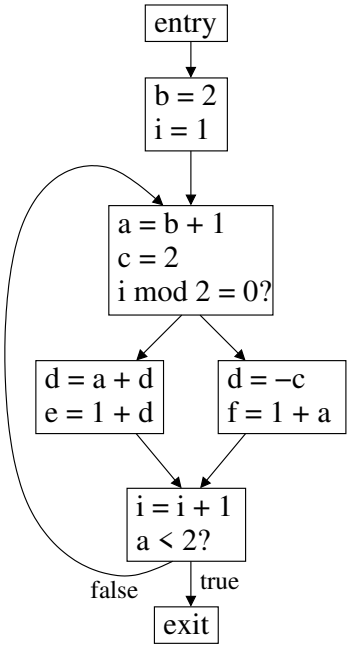
Consider the code in the figure and assume that the array `z` is the only variable alive at the exit of the code.

- Which variables are alive at the end of block B7?
- Which blocks dominate B7?
- Assume that `x` and `y` are parameters of the function initialized before the function call. Is there any variable that could be potentially used without having been initialized before?
- Apply all possible optimizations you know.



5. Consider the code in the figure:

   - Apply all possible optimizations.
   - Discuss how much information can be extracted at compile time about the execution of the code.



6.

```
1: a = y + c
2: d = a + x
3: b = x + y
4: a = b * d
5: e = x - a
6: y = b + x
7: a = e * b
```

Let us assume that the set of live variables at the end of the code at the left is $\{a, y\}$.

- Indicate which variables are alive at each point of the code.
- Rewrite the code using a minimum number of registers. Name the registers as `R1`, `R2`, `R3`, `R4`, etc.