1 Exercises

Exercise 1. The LIR instructions for the `compute()` method from the `SpimSum` program above are listed below.

```
B0

B1
0: LDC [0] [V32|I]
5: MOVE [V32|I] [V33|I]
10: MOVE $a0 [V34|I]

B2
15: LDC [0] [V35|I]
20: BRANCH [LE] [V34|I] [V35|I] B4

B3
25: LDC [-1] [V36|I]
30: ADD [V34|I] [V36|I] [V37|I]
35: ADD [V33|I] [V34|I] [V38|I]
40: MOVE [V37|I] [V34|I]
45: MOVE [V38|I] [V33|I]
50: BRANCH B2

B4
55: MOVE [V33|I] $v0
60: RETURN $v0
```

a. Compute the liveUse and liveDef sets (local liveness information) for each basic block in the method.

b. Compute the liveIn and liveOut sets (global liveness information) for each basic block in the method.

c. Compute the liveness interval for each virtual register in the method's LIR, with ranges and use positions.

Exercise 2. Using the liveness intervals for the virtual registers in the LIR for the `SpimSum.compute()` method

a. Build an interference graph G for the method.

b. Represent G as an adjacency matrix.

c. Represent G as an adjacency list.

d. Is G 2-colorable? If so, give a register allocation using two physical registers r_1 and r_2.

e. Is G 3-colorable? If so, give a register allocation using three physical registers r_1, r_2, and r_3.

2 Solutions to Exercises

Solution 1.

a.

```
B0
  liveUse:
  liveDef:

B1
  liveUse: $a0
  liveDef: V32, V33, V34

B2
  liveUse: $a0, V34
  liveDef: V35

B3
  liveUse: V33, V34
  liveDef: V33, V34, V36, V37, V38

B4
  liveUse: V33
  liveDef: $v0
```

b.

```
B4
  liveIn:  V33
  liveOut:

B3
  liveIn:  V33, V34
  liveOut: V33

B2
  liveIn:  $a0, V34
  liveOut: V33, V34

B1
  liveIn:  $a0
  liveOut: $a0, V34

B0
  liveIn:
  liveOut: $a0
```

c.

```
   v0: [55, 60]
   a0: [0, 10]
  V32: [0, 5]
  V33: [5, 35] [45, 55]
  V34: [10, 50]
  V35: [15, 20]
  V36: [25, 30]
  V37: [30, 40]
  V38: [35, 45]
```