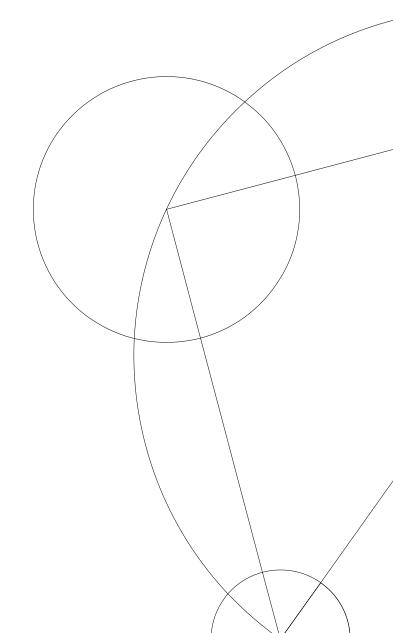


Assignment 3

Adit (hjg708)

IPS



Task 1

Statement:

Intermediate code:

```
1 Label L1:
     t1 = 0
      t2 = v1
if t2 == t1 then END else L2
3
6 Label L2:
     t3 = 1
      t4 = v0
     t5 = v1
t6 = t4 / t5
if t3 < t6 then L3 else END
9
10
11
12
13 Label L3:
16
      if t8 < t7 then L4 else L5
17
18 Label L4:
t11 = t9 - t10
v0 = t11
21
22
      goto L6
23
24
25 Label L5:
26 t12 = v0
27 t13 = v1
     t14 = t13 - t12
v1 = t14
goto L6
28
29
30
31
32 Label L6:
goto L1
35 Label END:
```

MIPS:

```
1 L1:
                r1, R0, v1
r2, R0, 0
r3, r4, END
    add
        add
        beg
6 L2:
                   r3, R0, 1
r5, R0, v0
     addi
add
                 r6, R0, v1
r4, r5, t6
r7, r3, r4
     add
div
slt
10
11
      beq
                 r7, R0, END
13
14 L3:
    add
15
                 r8, R0, v1
                r9, R0, v0
r10, r8, r9
r10, R0, L5
      add
slt
16
17
      beq
18
19
20
   L4:
    add
add
sub
mov
j
                 r12, R0, v0
21
                 r13, R0, v1
r11, r12, r13
v0, r11
22
23
24
                   L6
25
26
27 L5:
   add
add
sub
mov
j
                r15, R0, v0
r16, R0, v1
r14, r16, r15
29
30
31
                  v1, r14
                   L6
32
33
34 L6:
                 L1
35 j
37 END:
```

Task 2

IL pattern	replacement	
$z := x \ge y$	slt	r_w, r_x, r_y
w := !z	xori	$r_z, r_w, 1$
$z := x \ge y$ $w := !z^{last}$	slt	r_w, r_x, r_y
$z := x \ge y$	slt	r_z, r_x, r_y
	xori	$r_z, r_z, 1$
w := !z	xori	$r_z, r_z, 0$
	xori	$r_w, r_z, 1$

Task 3

a)

The scan function applies a binary operation to each element of the input array x, accumulating the results in a new array b, which is then returned as the output of the function.

```
bool* scan(bool myop(bool, bool), bool ne, bool* x) = {
   int len = length(x);
   bool* y = malloc(len);
   int i = 1;
   y[0] = myop(ne, x[0])
   while(i < len) {
      bool tmp = myop(y[i-1], x[i]);
      y[i] = tmp;
      i = i + 1;
   }
   return y;
}</pre>
```

I tested my implementation of the scan function by using the same inputs as provided in the group project description: scan(plus, 0, $\{1,2,3,4\}$). I did a step-by-step calculation for that specific, and in the end I ended up with the same output as the example, which is $\{1,3,6,10\}$

b)

```
1 1w
       R_{en} = 0 (R_x)
       R_y, R_hp
2 mv
4 slli R_tmp, R_len, 2
^{5} addi R_tmp, R_tmp, 4
  add
       R_hp, R_hp, R_tmp
      R_{len}, O(R_y)
7 SW
9 addi R_ix, R_x, 4
addi R_{iy}, R_{y}, 4
11 mv
      R_i, zero
12
13 loop_beg:
    sub R_tmp, R_i, R_len
14
    bgez R_tmp, loop_end
15
16
    lw
           R_tmp, O(R_ix)
17
    addi R_ix, R_ix, 4
18
19
           R_clen, 8
20
21
    mν
           R_c, R_hp
22
    add
           R_hp, R_hp, R_clen
    sw
           R_clen, O(R_c)
23
           R_ne, 4(R_c)
24
    sw
           R_{temp}, 8(R_c)
26
27
    \#Call\ myop\ with\ the\ parameter\ saved\ in\ R\_c
    call myop(R_c, R_tmp)
29
    sw
           R_tmp, O(R_iy)
          R_iy, R_iy, 4
R_i, R_i, 1
31
    addi
32
    addi
           loop_beg
    j
34
35 loop_end:
```

I used the MIPS code from slide 26 in 'Machine Code Generation'. Where I then implement the missing part with the 'myop' function.