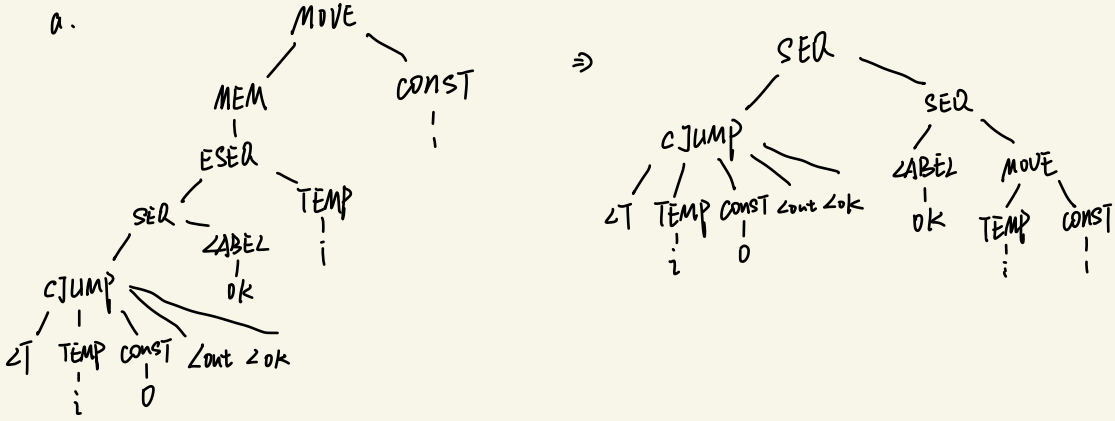
8.2 Draw each of the following expressions as a tree diagram, and then apply the rewriting rules of Figure 8.1 and Exercise 8.1, as well as the CALL rule on page 183.

a. MOVE(MEM(ESEQ(SEQ(CJUMP(LT, TEMP$_i$, CONST$_0$, L_{out}, L_{ok}), LABEL$_{ok}$), TEMP$_i$)), CONST$_1$)

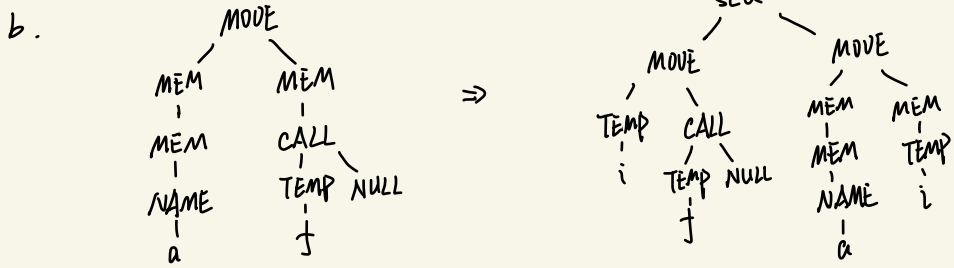
b. MOVE(MEM(MEM(NAME$_a$)), MEM(CALL(TEMP$_f$, []))

c. BINOP(PLUS, CALL(NAME$_f$, [TEMP$_x$]), CALL(NAME$_g$, [ESEQ(MOVE(TEMP$_x$, CONST$_0$), TEMP$_x$)]))

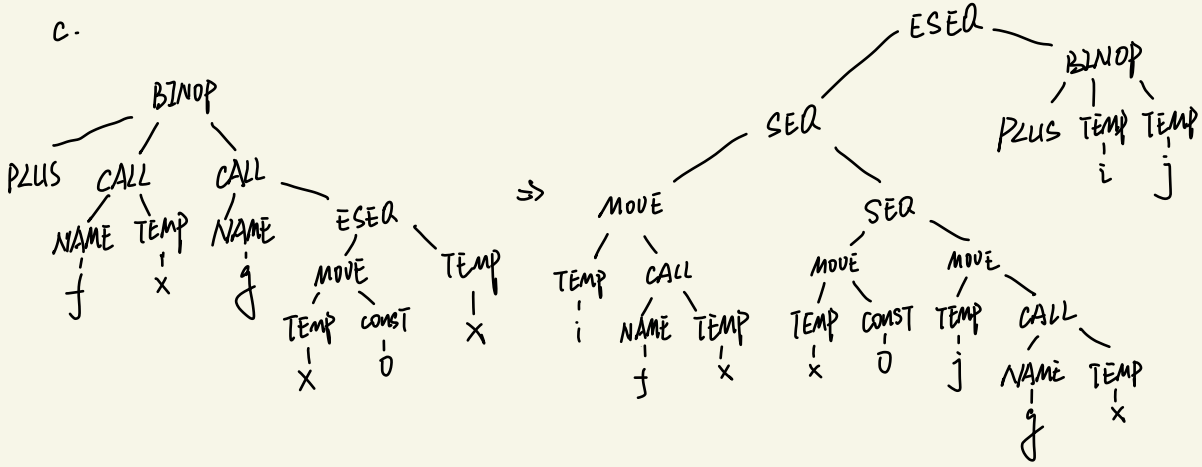
a.



b.



c.



Break this program into basic blocks.

```
1  m ← 0
2  v ← 0
3  if v ≥ n goto 15
4  r ← v
5  s ← 0
6  if r < n goto 9
7  v ← v + 1
8  goto 3
9  x ← M[r]
10 s ← s + x
11 if s ≤ m goto 13
12 m ← s
13 r ← r + 1
14 goto 6
15 return m
```

LABEL 1
m ← 0
V ← 0
goto LABEL 3

LABEL 3
if V ≥ n goto LABEL 15

LABEL 4
r ← V
S ← 0
goto LABEL 6

LABEL 6
if r < n goto LABEL 9

LABEL 7
V ← V + 1
goto LABEL 3

LABEL 9
x ← M[r]
S ← S + x
if S ≤ m goto LABEL 13

LABEL 12
m ← S
goto LABEL 13

LABEL 13
r ← r + 1
goto LABEL 6

LABEL 15
return m

8.7 Express the basic blocks of Exercise 8.6 as statements in the `Tree` intermediate form, and use Algorithm 8.3 to generate a set of traces.

SEQ(LABEL(1), MOVE(TEMP(m), CONST(0)))
MOVE(TEMP(V), CONST(0))
JUMP(3)
SEQ(LABEL(3), CJUMP(GE, TEMP(v), TEMP(n), 15, 4))
SEQ(LABEL(4), MOVE(TEMP(r), TEMP(v)))
MOVE(TEMP(s), CONST(0))
JUMP(6)
SEQ(LABEL(6), CJUMP(LT, TEMP(v), TEMP(n), 9, 7))
SEQ(LABEL(7), MOVE(TEMP(v), BINOP(PLUS, TEMP(v), CONST(1))))
JUMP(3)
SEQ(LABEL(9),
MOVE(TEMP(x), MEM(BINOP(PLUS, TEMP(m), BINOP(MUL, TEMP(r), CONST(k))))))
MOVE(TEMP(s), BINOP(PLUS, TEMP(s), TEMP(x)))
CJUMP(LE, TEMP(s), TEMP(m), 13, 12)
SEQ(LABEL(12), MOVE(TEMP(m), TEMP(s)))
JUMP(13)
SEQ(LABEL(13), MOVE(TEMP(r), BINOP(PLUS, TEMP(r), CONST(1))))
JUMP(6)
SEQ(LABEL(15), SEQ(MOVE(TEMP(a0), TEMP(m)), JUMP(TEMP(ra))))

Trace:

1 -> 3 -> 4 -> 6 -> 9 -> 13 -> 6 -> 7 -> 3 -> 15

12