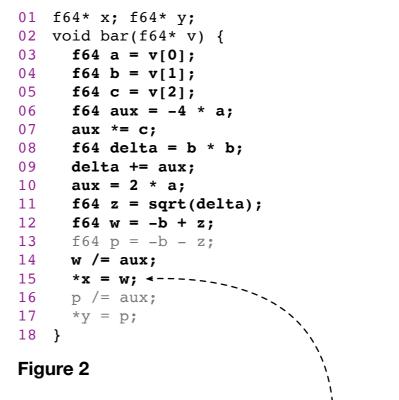The three programs below should store the same values in x and y. In boldface we show the subprograms that compute the value of -b + sqrt(b^2 - 4ac)/2a.

```
01 f64* x; f64* y;
02 void foo(f64 a, f64 b, f64 c) {
03    f64 d = b * b;
04    f64 t = -4 * a;
05    t *= c;
06    d += t;
07    f64 s = sqrt(d);
08    f64 m = -b + s;
09    f64 p = -b - s;
10    t = 2 * a;
11    m /= t;
12    p /= t;
13    *x = m;
14    *y = p;
15 }
```

Figure 1

```
01 f64* x, y
02 void bar(f64 a, f64 b, f64 c) {
03    f64 d = -4 * a * c + b * b;
04    *x = -b + d / 2*a;
05    *y = -b - d / 2*a;
06 }
```

Figure 3: Baskhara's formula.

```
01  f64* x; f64* y;
02  void bar(f64* v) {
03     f64 a = v[0];
04     f64 b = v[1];
05     f64 c = v[2];
06     f64 aux = -4 * a;
07     aux *= c;
08     f64 delta = b * b;
09     delta += aux;
10     aux = 2 * a;
11     f64 z = sqrt(delta);
12     f64 w = -b + z;
13     f64 p = -b - z;
14     w /= aux;
15     *x = w;  ←-------
16     p /= aux;
17     *y = p;
18  }
```

Figure 2

1. Can you show, somehow, that the programs in Fig-1 and Fig-2 are storing the same values in *x, assuming that they receive the same inputs in variables a, b and c?

2. What would be a nice way to represent the code to conclude that the two programs are actually doing the same thing?

3. More generally, can you think about a way to demonstrate that the program in Fig-3 is also storing the same value in *y, again, assuming equal inputs? Which data structures would you need to conclude that these programs are actually computing equivalent expressions?