Write a Pascal program that solves quadratic equations for real and imaginary roots.

```
program AssignmentSolveQuadraticEquation;

uses math;
var
  coeffA, coeffB, coeffC, disc, realPart, imagPart, sol1, sol2: real;
  count: integer;

function findDiscriminant(x, y, z: real): real;
begin
  findDiscriminant := (y * y) - (4 * x * z);
end;

begin
  count := 0;
  writeln('This application solves quadratic equations for real or
complex roots.');
  writeln('Provide the input as three coefficients A B C.');
  writeln('Enter 0 0 0 to exit.');

  repeat
    writeln;
    writeln('Input the values for a b c:');
    readln(coeffA, coeffB, coeffC);

    if (coeffA = 0) and (coeffB = 0) and (coeffC = 0) then
      break;

    disc := findDiscriminant(coeffA, coeffB, coeffC);

    if disc > 0 then
    begin
      sol1 := (-coeffB + sqrt(disc)) / (2 * coeffA);
      sol2 := (-coeffB - sqrt(disc)) / (2 * coeffA);
      writeln('Real roots found...');
      writeln('Solution 1 = ', sol1:0:16);
      writeln('Solution 2 = ', sol2:0:16);
    end
    else if disc = 0 then
    begin
      sol1 := -coeffB / (2 * coeffA);
      writeln('One real root...');
      writeln('Solution = ', sol1:0:16);
    end
    else
```

```
    begin
      realPart := -coeffB / (2 * coeffA);
      imagPart := sqrt(-disc) / (2 * coeffA);
      writeln('Complex roots detected...');
      writeln('Complex Part Root 1 = ', imagPart:0:16);
      writeln('Complex Part Root 2 = ', -imagPart:0:16);
      writeln('Real Part  Root - ', realPart:0:16)
    end;

    count := count + 1;

  until (coeffA = 0) and (coeffB = 0) and (coeffC = 0);

  writeln;
  writeln(count, ' equations processed.');
  readln;
end.
```