We make use of the following packages:

```maple
> with(RegularChains): with(SemiAlgebraicSetTools):
```

Additionally, we use Maple code written at the University of Bath: The ProjectionCAD package (should be hosted alongside this worksheet).

```maple
> read("ProjectionCAD.mpl"):  with(ProjectionCAD):
```

The example set used in Section 8.2 is stored in Maple formats in the following textfile.

```maple
> read("Section82-ExampleSet.txt");
```

We can access examples by number. Two input formats are available:

The first gives input suitable for sign-invariant CAD

```maple
> I1, I2 := GenerateInput(19, "CAD");
```

```maple
I1, I2 := [x^2 + y^2 - 1, 4 x y - 1, x^2 + y^2 - 8 x - 2 y + 16, 4 x y - 4 x - 16 y + 15], [y, x]
```

We can build these CADs using either:

- Projection and Lifting implemented in the ProjectionCAD Package (FullCAD in Table 2);

```maple
> CADFull(I1, I2, method=McCallum, output=list): nops(%);
```

```maple
317
```

- Regular Chains implemented in the RegularChains Library of Maple (Maple in Table 2).

```maple
> CylindricalAlgebraicDecompose( I1, PolynomialRing(I2), output=list): nops(%);
```

```maple
317
```

The second gives output suitable for TTICAD by Projection and Lifting implemented in Projection CAD Package (TTICAD in Table 2).

```maple
> I1, I2 := GenerateInput(19, "TTICAD");
```

```maple
I1, I2 := [[y^2 + x^2 - 1, [4 y x - 1]], [y^2 + x^2 - 2 y - 8 x + 16, [4 y x - 16 y - 4 x + 15]]], [y, x]
```

```maple
> TTICAD( I1, I2, output=list): nops(%);
```

```maple
105
```

Note that the timings reported in Section 8.2 were found when running in command line mode using Maple 16.