# Sunrise/Sunset Algorithm

```
Source:
        Almanac for Computers, 1990
        published by Nautical Almanac Office
        United States Naval Observatory
        Washington, DC 20392

Inputs:
        day, month, year:      date of sunrise/sunset
        latitude, longitude:   location for sunrise/sunset
        zenith:                Sun's zenith for sunrise/sunset
          offical      = 90 degrees 50'
          civil        = 96 degrees
          nautical     = 102 degrees
          astronomical = 108 degrees

        NOTE: longitude is positive for East and negative for West
        NOTE: the algorithm assumes the use of a calculator with the
        trig functions in "degree" (rather than "radian") mode. Most
        programming languages assume radian arguments, requiring back
        and forth convertions. The factor is 180/pi. So, for instance,
        the equation RA = atan(0.91764 * tan(L)) would be coded as RA
        = (180/pi)*atan(0.91764 * tan((pi/180)*L)) to give a degree
        answer with a degree input for L.


1. first calculate the day of the year

        N1 = floor(275 * month / 9)
        N2 = floor((month + 9) / 12)
        N3 = (1 + floor((year - 4 * floor(year / 4) + 2) / 3))
        N = N1 - (N2 * N3) + day - 30

2. convert the longitude to hour value and calculate an approximate time

        lngHour = longitude / 15

        if rising time is desired:
          t = N + ((6 - lngHour) / 24)
        if setting time is desired:
          t = N + ((18 - lngHour) / 24)

3. calculate the Sun's mean anomaly

        M = (0.9856 * t) - 3.289

4. calculate the Sun's true longitude

        L = M + (1.916 * sin(M)) + (0.020 * sin(2 * M)) + 282.634
        NOTE: L potentially needs to be adjusted into the range [0,360) by adding/subtracting 360

5a. calculate the Sun's right ascension

        RA = atan(0.91764 * tan(L))
        NOTE: RA potentially needs to be adjusted into the range [0,360) by adding/subtracting 360

5b. right ascension value needs to be in the same quadrant as L

        Lquadrant  = (floor( L/90)) * 90
```

```
        RAquadrant = (floor(RA/90)) * 90
        RA = RA + (Lquadrant - RAquadrant)

5c. right ascension value needs to be converted into hours

        RA = RA / 15

6. calculate the Sun's declination

        sinDec = 0.39782 * sin(L)
        cosDec = cos(asin(sinDec))

7a. calculate the Sun's local hour angle

        cosH = (cos(zenith) - (sinDec * sin(latitude))) / (cosDec * cos(latitude))

        if (cosH >  1)
          the sun never rises on this location (on the specified date)
        if (cosH < -1)
          the sun never sets on this location (on the specified date)

7b. finish calculating H and convert into hours

        if if rising time is desired:
          H = 360 - acos(cosH)
        if setting time is desired:
          H = acos(cosH)

        H = H / 15

8. calculate local mean time of rising/setting

        T = H + RA - (0.06571 * t) - 6.622

9. adjust back to UTC

        UT = T - lngHour
        NOTE: UT potentially needs to be adjusted into the range [0,24) by adding/subtracting 24

10. convert UT value to local time zone of latitude/longitude

        localT = UT + localOffset
```