

```

/*
pressure = p
specificvolume = v
temperature = t
internalenergy = u
entropy = s
enthalpy = h
quality = q
*/

public static int region_given_pvt(double p, double v, double t)

public static double p_given_vt(double v, double t)
{
    return IAPWS95.IAPWS95_pressure(1.0/v,t)*1000.0; //expects:Kg/M^3,K returns KPa
}

public static double v_given_pt(double p, double t)    works outside of the vapor dome
{
    return 1.0/IF97.rhomass_Tp(t,p/1000000.0); //expects:K,MPa returns Kg/M^3
}

public static double tsat_given_p(double p)
{
    return IF97.Tsat97(p/1000000.0); //UNIT CONVERSION UNTESTED!
}

public static double vliq_given_p(double p)    could do the same thing here with u (or h or s)
{
    return 1.0/IF97.rholiq_p(p/1000000.0); //expects:MPa returns Kg/M^3
}

public static double vvap_given_p(double p)    could do the same thing here with u (or h or s)
{
    return 1.0/IF97.rhovap_p(p/1000000.0); //expects:MPa returns Kg/M^3
}

public static double u_given_pt(double p, double t)    works outside the vapor dome
{
    return IF97.umass_Tp(t, p/1000000.0); //UNIT CONVERSION UNTESTED!
}

public static double u_given_vt(double v, double t)
{
    return IAPWS95.IAPWS95_internal_energy(1f/v,t); //UNIT CONVERSION UNTESTED!
}

public static double s_given_vt(double v, double t)
{
    return IAPWS95.IAPWS95_entropy(1f/v,t)*1000f; //UNIT CONVERSION UNTESTED!
}

```

```
public static double h_given_vt(double v, double t)
{
    return IAPWS95.IAPWS95_enthalpy(1f/v,t)*1000f; //UNIT CONVERSION UNTESTED!
}
```