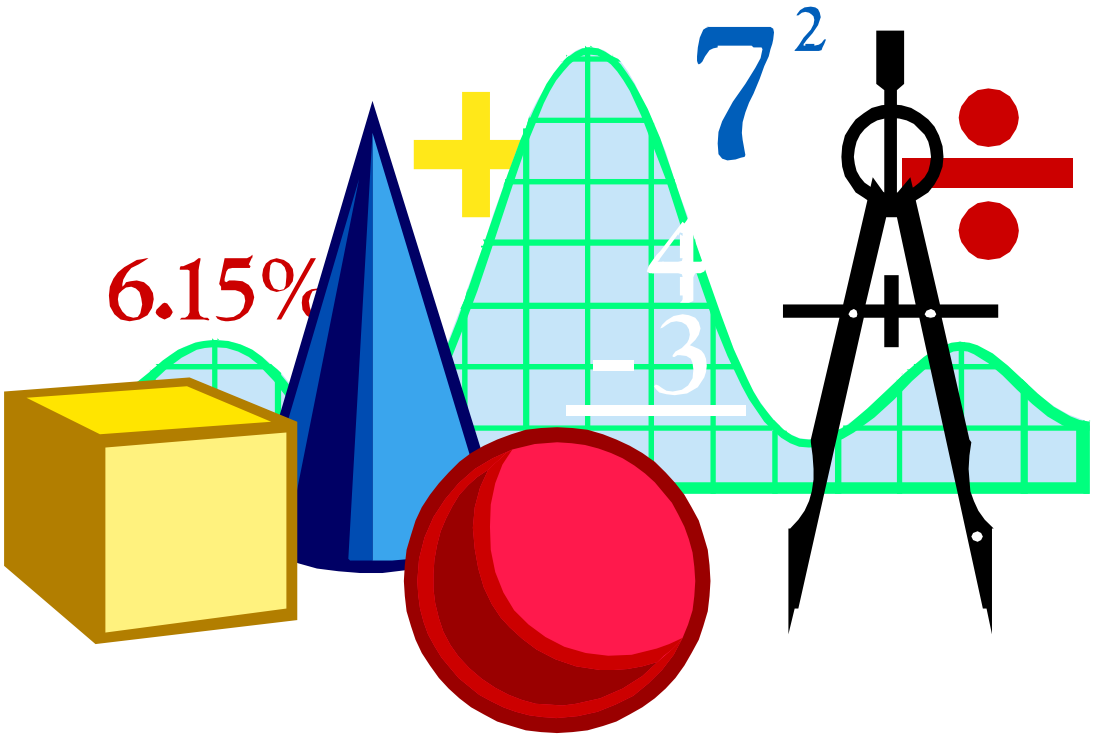
Meteorological Equations and Conversion Algorithms



Autumn 1997

Heat Index & Wind Chill Index

Last Updated: August 25, 1997

Heat Index

Calculates the heat index.

Source Code #1

```
function calcHeatIndex(t, rh)
{
    var t2=t*t;
    var rh2=rh*rh;

    var result =  -42.379 + 2.04901523 * t + 10.14333127 * rh -
        0.22475541 * t * rh - 6.83783e-3 * t2 - 5.481717e-2*rh2 +
        1.22874e-3*t2*rh + 8.5282e-4 * t * rh2 - 1.99e-6 * t2 * rh2;
    return result;
}
```

Source Code #2

```
function calcHeatIndex (t, rh)
{
    var t2 = t*t;
    var t3 = t2*t;
    var rh2 = rh*rh;
    var rh3=rh2*rh;

    var result = 16.923 + 0.185212*t + 5.37941*rh -
        0.100254*t*rh + (0.941695e-2)*t2 + (0.728898e-2)*rh2 +
        (0.345372e-3)*t2*rh - (0.814971e-3)*t*rh2 + (0.102102e-4)*t2*rh2 -
        (0.38646e-4)*t3 + (0.291583e-4)*rh3) + (0.142721e-5)*t3*rh) +
        (0.197483e-6)* t*rh3 - (0.218429e-7)*t3*rh2 + (0.843296e-9)*t2*rh3 -
        (0.481975e-10)*t3*rh3;

    return result;
}
```

Parameters

t is the temperature in degrees Fahrenheit.
rh is the relative humidity between 0 and 1 (or rh in percent / 100).

Results

Heat index in degrees Fahrenheit.

Examples

t = 95, *rh* = .4, function returns: 100.4
t = 90, *rh* = .25, function returns: 87.3
t = 90, *rh* = .55, function returns: 97.7

Wind Chill In Fahrenheit

Calculates the wind chill in degrees Fahrenheit.

Source Code

```
function calcWindChillF(t, v)
{
        var wc = 0.0817*(3.71*Math.sqrt(v) + 5.81 - 0.25 * v) * (t - 91.4) +
91.4;
        return (wc < t) ? wc : t;
}
```

Parameters

t is the temperature in degrees Fahrenheit.
v is the wind speed in miles per hour.

Results

Returns the wind chill in degrees Fahrenheit.

Examples

t = 32, *v* = 10, function returns: 18.4
t = 28, *v* = 5, function returns: 24.8
t = 24, *v* = 17, function returns: -1.4

Wind Chill In Celsius

Source Code

Calculates the wind chill in degrees Celsius.

```
function calcWindChillC(t, v)
{
        var wc = 0.045(Math.sqrt(5.27*v) + 10.45 - 0.28*v)*(t - 33) + 33
        return (wc < t) ? wc : t;
}
```

Parameters

t is the temperature in degrees Celsius.
v is the wind speed in meters per second.

Results

Returns the wind chill in degrees Celsius.

Examples

t = 0, *v* = 5, function returns: -8.6
t = -2, *v* = 2, function returns: -2.9
t = 2, *v* = 8, function returns: -10.1

Wind Chill In Watts

Calculates the wind chill as heat loss in watts per second.

Source Code

```
function calcWindChillWatts(tc, v)
{
        return (12.1452 + 11.6222 * Math.sqrt(v) -1.16222 * v) * (33 - tc)
}
```

Parameters

t is the temperature in degrees Celsius.
v is the wind speed in meters per second.

Results

Returns the heat loss in watts per second.

Examples

t = 0, *v* = 5, function returns: 1066.6
t = -2, *v* = 2, function returns: 919.0
t = 2, *v* = 8, function returns: 1107.3

Calculating Humidity Properties

Last Updated: October 15, 1997

Vapor Pressure From Temperature

Calculates the vapor pressure for a given temperature. There is no one algorithm for calculating vapor pressure so I've included two that you can use. Which ever one you choose, make sure to use the corresponding source in **Error! Reference source not found.**.

Source Code #1

```
function calcVaporPressure(t)
{
    return 6.112 * Math.pow(10, (7.5 * t) / (237.7 + t));
}
```

Source Code #2

```
function calcVaporPressure(t)
{
    return 6.112 * Math.exp((17.67 * t) / (243.5 + t))
}
```

Parameters

t is the temperature in Celsius

Results

Returns the vapor pressure in millibars.

Examples

t = 22, function #1 returns: 26.3956
t = 11, function #1 returns: 13.1192
t = 22, function #2 returns: 26.4283
t = 11, function #2 returns: 13.1180

Temperature From Vapor Pressure

Finds the temperature from a given vapor pressure.

Source Code #1

```
function calcTcFromVaporPressure(vp)
{
    var logs = Math.log(vp / 6.112) / Math.LN10;
    return (237.7 * logs) / (7.5 - logs);
}
```

Source Code #2

```
function calcTcFromVaporPressure(vp)
{
    var logs = Math.log(vp / 6.112);
    return (243.15 * logs) / (17.67 - logs);
}
```

Parameters

vp is the vapor pressure in millibars

Results

Returns the corresponding temperature in Celsius.

Examples

vp = 26.3956, function #1 returns 22.0
vp = 13.1192, function #1 returns 11.0
vp =26.4283, function #2 returns 22.0
vp = 13.1180, function #2 returns 11.0

Relative Humidity From Vapor Pressure

Finds the relative humidity from the actual and saturated vapor pressures.

Source Code

```
function calcRelHumidityFromVp(actual, satur)
{
    return actual / satur * 100.0;
}
```

Parameters

actual is the vapor pressure in millibars found using the dewpoint temperature in degrees Celsius.
satur is the vapor pressure in millibars found using the drybulb temperature in degrees Celsius.

Results

Returns the relative humidity as a decimal. For a percentage, multiply by 100.

Examples

dewpoint = 11, *actual* = 13.1
drybulb = 22, *satur* = 26.4
result is: .496 or 49.6%

Wetbulb

Estimates the wetbulb temperature given pressure, drybulb, and dewpoint.

Source Code

```
function calcWetbulb (press, t, dp)
{
    var tmin = Math.min(dp,t);
    var tmax = Math.max(dp,t);
    var e = calcVaporPressure(dp);
    var vpcur;
    var peq;
    var diff;
    var tcur;

    while (true)
    {
        tcur = (tmax + tmin) / 2;
        vpcur = calcVaporPressure(tcur);
        peq = 0.00066 * (1+0.00155 * tcur) * press * (t - tcur);
        diff = peq - vpcur + e;

        if (Math.abs(diff) < 0.01) break;

        if (diff < 0) tmax = tcur;
        else tmin = tcur;
    }
    return tcur;
}
```

Parameters

press is the barometric pressure in millibars
t is a temperature in degrees Celsius
dp is the dewpoint temperature in degrees Celsius

Results

Returns the estimate of the wetbulb temperature in degrees Celsius.

Examples

press = 1013, *t* = 22, *dp* = 11, function returns: 15.5
press = 950, *t* = 22, *dp* = 11, function returns: 15.3
press = 1013, *t* = 0, *dp* = -8, function returns: -2.6

Specific Humidity

Calculates the specific humidity from the actual vapor pressure and barometric pressure.

Source Code

```
function calcSpecHumidity(e, press)
{
    return (0.62197 * e) / (press - (0.37803 * e)) * 1000.0;
}
```

Parameters

e is the vapor pressure in millibars of the dewpoint temperature.
press is the barometric pressure in millibars.

Results

Returns the specific humidity in grams per kilogram.

Examples

e = 13.12 (from dewpoint of 11 °C), *press* = 1013, function returns: 8.09
e = 6.11 (from dewpoint of 0 °C), *press* = 1013, function returns: 3.76

Mixing Ratio

Calculates the mixing ratio from the actual vapor pressure and barometric pressure.

Source Code

```
function calcMixingRatio(e, press)
{
    return ((0.62197 * e)/(press - e)) * 1000.0;
}
```

Parameters

e is the vapor pressure in millibars of the dewpoint temperature.
press is the barometric pressure in millibars.

Results

Returns the mixing ratio in grams per kilogram.

Examples

e = 13.12 (from dewpoint of 11 °C), *press* = 1013, function returns: 8.2
e = 6.11 (from dewpoint of 0 °C), *press* = 1013, function returns: 3.8

Absolute Humidity

Calculates absolute humidity from the actual vapor pressure and temperature.

Source Code

```
function calcAbsHumidity(e, t)
{
    return (e * 100) / (461.51 * (273.15 + t)) * 1000.0;
}
```

Parameters

e is the vapor pressure in millibars of the dewpoint temperature.
t is the temperature in Celsius.

Results

Returns the absolute humidity in grams per cubed meter.

Examples

e = 13.12 (from dewpoint of 11 °C), *press* = 1013, function returns: 9.6
e = 6.11 (from dewpoint of 0 °C), *press* = 1013, function returns: 4.7

Weather Algorithms: Calculating With Humidity Variables

Last Updated: August 23, 1997

Calculations With Dewpoint

Source Code

```
function calcWithDp(drybulb, dewpoint, pressure)
{
    var result = new Object();
    var havePress = (pressure != 0);

    result.rh = calcRelHumidity(e, es);

// Use variable names familiar to meteorologists.
// Feel combine these four assignments into two

    var es = calcVaporPressure(drybulb);
    var e = calcVaporPressure(dewpoint);
    result.saturvp = es;
    result.actualvp = e;

    result.abshum = calcAbsHumidity(e, drybulb);

    if (!havePress) return;

    result.wetbulb = calcWetbulb(pressure, drybulb,  dewpoint);
    result.mixratio = calcMixingRatio(e, pressure);
    result.spechum = calcSpecHumidity(e, pressure);

    return result;
}
```

Parameters

drybulb is the drybulb in Celsius.
dewpoint is the dewpoint in Celsius.
pressure is the pressure in millibars or 0 if no pressure is to be assumed.

Results

An object with these properties:

- *rh* is the relative humidity
- *saturvp* is the vapor pressure in millibars from the drybulb temperature.
- *actualvp* is the vapor pressure in millibars from the dewpoint termperature.
- *abshum* is the absolute humidity in grams per cubed meter.
- If pressure is non-zero, *wetbulb* is the wetbulb estimate in degrees Celsius.
- If pressure is non-zero, *mixratio* is the mixing ratio in grams per kilogram.
- If pressure is non-zero, *spechum* is the specific humidity in grams per kilogram.

Calculating With Relative Humidity

Source Code

```
function calcWithRh(drybulb, rh, pressure)
{
    var result = new Object();
    var havePress = (pressure != 0);
    var es = calcVaporPressure(drybulb);
    var e = es * rh / 100;

    result.dewpoint = calcTcFromVaporPressure(e);

    if (rh == 100) result.wetbulb = dewpoint;

    result.saturvp = es;
    result.actualvp = e;
    result.abshum = calcAbsHumidity(e, drybulb);

    if (!havePress) return;

    if (rh != 100) result.wetbulb = calcWetbulb(pressure, drybulb, dewpoint);

    result.mixratio = calcMixingRatio(e, press);
    result.spechum = calcSpecHumidity(e, press);

    return result;
}
```

Parameters

drybulb is the drybulb in Celsius.
rh is the relative humidity in decimal.
pressure is the pressure in millibars or 0 if no pressure is to be assumed.

Results

An object with these properties:

- *dewpoint* is the dewpoint in degrees Celsius.
- *saturvp* is the vapor pressure in millibars from the drybulb temperature.
- *actualvp* is the vapor pressure in millibars from the dewpoint termperature.
- *abshum* is the absolute humidity in grams per cubed meter.
- If pressure is non-zero, *wetbulb* is the wetbulb estimate in degrees Celsius.
- If pressure is non-zero, *mixratio* is the mixing ratio in grams per kilogram.
- If pressure is non-zero, *spechum* is the specific humidity in grams per kilogram.

Calculating With Wetbulb

Source Code

```
function calcWithWb(drybulb, wetbulb, pressure)
{
    var result = new Object();
    var es = calcVaporPressure(drybulb);
    var esw = calcVaporPressure(wetbulb);

    result.saturvp = es;

// Make a somewhat dangerous assumption
    if (pressure == 0) pressure = 1013;

    var e = esw - (0.00066 * (1 + 0.00155 * wetbulb) * press) * (drybulb -
wetbulb);

    result.actualvp = e;
    result.abshum = calcAbsHumidity(e, drybulb);
    result.dewpoint = calcTcFromVaporPressure(e);
    result.rh = calcRelHumidity(e,es);
    result.mixratio = calcMixingRatio(e, pressure);
    result.spechum = calcSpecHumidity(e, pressure);

    return result;
}
```

Parameters

drybulb is the drybulb in Celsius.
wetbulb is the wetbulb in Celsius.
pressure is the pressure in millibars or 0 if no pressure is to be assumed.

Results

An object with these properties:

- *dewpoint* is the dewpoint in degrees Celsius.
- *rh* is the relative humidity.
- *saturvp* is the vapor pressure in millibars from the drybulb temperature.
- *actualvp* is the vapor pressure in millibars from the dewpoint termperature.
- *abshum* is the absolute humidity in grams per cubed meter.
- *mixratio* is the mixing ratio in grams per kilogram.
- *spechum* is the specific humidity in grams per kilogram.

Unit Conversions

Last Updated: August 23, 1997

Temperature Unit Conversions

Celsius To Fahrenheit

fahrenheit = (celsius * 9 / 5) + 32

Fahrenheit To Celsius

celsius = (fahrenheit-32) * 5 / 9

Kelvin Conversions

kelvin = celsius + 273.15
celsius = kelvin - 273.15

Pressure Unit Conversions

Note: Millibars are also known as hectopascals

Inches To Millibars

millibars = inches * 33.8653

Millibars To Inches

inches = millibars / 33.8653

Kilopascal Conversions

millibars = kilopascals * 10
kilopascals = millibars / 10

Wind Unit Conversions

Convert wind units using the wind unit conversion table and the instructions below.

Find the unit you are converting in the **Convert From** column on the left.
Then find the unit you are converting in the **Convert To** column across the top.
The cell that intersects the selected unit on the left and the selected unit across the top contain the number that you multiply by to convert your unit.

Wind Unit Conversion Table

Convert From	Convert To			
	miles/hour	meters/sec	kilometers/hour	knots
miles/hour	---	0.44704	1.60934	1.15193
meters/sec	2.23693	---	3.6	1.96936
kilometers/hour	0.62137	0.27778	---	0.54704
knots	0.86811	0.50778	1.82800	---