Pete

Could you help me write a scan for the close crossing above R2 and below S2...I tried but had no luck

```
input showlastswing = yes;
input swing_back1 = 5;
input swing_forward1 = 5;
input maxbars1 = 30;

def na = Double.NaN;
def bn = BarNumber();
def sb1 = swing_back1;
def sf1 = swing_forward1;
def lfor1 = Lowest(low, sf1)[-sf1];
def lback1 = Lowest(low, sb1)[1];
def swinglow1 = if low < lfor1 and low < lback1 then 1 else 0;
def swinglowbar = if low < lfor1 and low < lback1 then bn else na;
def hfor1 = Highest(high, sf1)[-sf1];
def hback1 = Highest(high, sb1)[1];
def swinghigh1 = if high > hfor1 and high > hback1 then 1 else 0;
def swinghighbar = if high > hfor1 and high > hback1 then bn else na;

#Swings identifed as points
plot r2 = if showlastswing == yes and bn == HighestAll(swinghighbar) then high else if
showlastswing == no and swinghigh1 then high else na;
plot s2 = if showlastswing == yes and bn == HighestAll(swinglowbar) then low else if
showlastswing == no and swinglow1 then low else na;

r2.SetStyle(Curve.POINTS);
r2.SetLineWeight(4);
r2.SetDefaultColor(Color.GREEN);
r2.HideBubble();
s2.SetStyle(Curve.POINTS);
s2.SetLineWeight(4);
s2.SetDefaultColor(Color.RED);
s2.HideBubble();


input bubbleoffsetfactor = 3;
input showbubbles_r2s2 = yes;
AddChartBubble(showbubbles_r2s2 and r2, high + TickSize() * bubbleoffsetfactor, "R2",
Color.GREEN, yes);
```

```
AddChartBubble(showbubbles_r2s2 and s2, low - TickSize() * bubbleoffsetfactor, "S2",
Color.RED, no);

#Store Previous Data
def r2save = if !IsNaN(r2) then r2 else r2save[1];
def s2save = if !IsNaN(s2) then s2 else s2save[1];

#Fibonacci
def data = CompoundValue(1, if (r2save == high) then data[1] + 1 else data[1], 0);
def datacount = (HighestAll(data) - data[1]) + 1;
input numberfibstoshow = 1;

input fib1level = .236;
input fib2level = .382;
input fibMlevel = .500;
input fib3level = .618;
input fib4level = .764;
input fib5level = 1.618;
input fib6level = 2.618;
input fib7level = -.618;
input fib8level = -1.618;

def fibh = r2save;
def fibl = s2save;
def range = fibh - fibl;

input showfiblines = yes;

plot fibHp = r2save;
plot fibLp = s2save;
plot fibM = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fibMlevel else na;
plot fib1 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib1level else na;
plot fib2 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib2level else na;
plot fib3 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib3level else na;
plot fib4 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib4level else na;
plot fib5 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib5level else na;
plot fib6 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib6level else na;
plot fib7 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib7level else na;
```

```
plot fib8 = if showfiblines == no then na else if datacount <= numberfibstoshow then fibl +
range * fib8level else na;

fibHp.SetPaintingStrategy(PaintingStrategy.DASHES);
fibLp.SetPaintingStrategy(PaintingStrategy.DASHES);
fibHp.SetLineWeight(2);
fibLp.SetLineWeight(2);
fibM.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib1.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib2.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib3.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib4.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib5.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib6.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib7.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fib8.SetPaintingStrategy(PaintingStrategy.HORIZONTAL);
fibHp.SetDefaultColor(Color.GREEN);
fibLp.SetDefaultColor(Color.RED);
fibM.SetDefaultColor(Color.WHITE);
fib1.SetDefaultColor(Color.CYAN);
fib2.SetDefaultColor(Color.YELLOW);
fib3.SetDefaultColor(Color.YELLOW);
fib4.SetDefaultColor(Color.CYAN);
fib5.SetDefaultColor(Color.YELLOW);
fib6.SetDefaultColor(Color.YELLOW);
fib7.SetDefaultColor(Color.YELLOW);
fib8.SetDefaultColor(Color.YELLOW);

fibM.HideBubble();
fib1.HideBubble();
fib2.HideBubble();
fib3.HideBubble();
fib4.HideBubble();
fib5.HideBubble();
fib6.HideBubble();
fib7.HideBubble();
fib8.HideBubble();

input showfib_bubbles = yes;
input bubble_mover = 5;
def n = bubble_mover;
def n1 = n + 1;
AddChartBubble(showfib_bubbles and !IsNaN(close[n1]) and IsNaN(close[n]), fibh[n1], "R2 "
+ AsText(fibh[n1]), Color.GREEN, yes);
AddChartBubble(showfib_bubbles and !IsNaN(close[n1]) and IsNaN(close[n]), fibl[n1], "S2 " +
AsText(fibl[n1]), Color.RED, yes);
```

```
AddChartBubble(showfib_bubbles and !IsNaN(close[n1]) and IsNaN(close[n]), fib5[n1],
"161.8%", Color.YELLOW, yes);
AddChartBubble(showfib_bubbles and !IsNaN(close[n1]) and IsNaN(close[n]), fib6[n1],
"261.8%", Color.YELLOW, yes);
AddChartBubble(showfib_bubbles and !IsNaN(close[n1]) and IsNaN(close[n]), fib7[n1],
"161.8%", Color.YELLOW, yes);
AddChartBubble(showfib_bubbles and !IsNaN(close[n1]) and IsNaN(close[n]), fib8[n1],
"261.8%", Color.YELLOW, yes);
```