In[129]:=

```
file = "C:\\Users\\sajib\\src\\DRalgo\\examples\\ah_pot.m";

raw = Import[file, "Text"];
```

In[131]:=

```
(*find the LAST V[phi_,T_]:=... line (DRalgo always puts at end)*)
lines = StringSplit[raw, "\n"];
potLine = Select[lines, StringContainsQ["V[phi_, T_]"]][[-1]];

rhs = StringTrim[StringSplit[potLine, ":="][[2]]];

Print["Extracted RHS of potential:"];
Print[rhs];
```

```
Extracted RHS of potential:
(msq*ϕ^2)/2 + (λ*ϕ^4)/4 - (g1^2*Yϕ^2*ϕ^2)^(3/2)/(6*Pi)
  + (g1^2*Yϕ^2*Sqrt[g1^2*Yϕ^2*ϕ^2]*Sqrt[msq + λ*ϕ^2])/(16*Pi^2) +
  (3*λ*(msq + λ*ϕ^2))/(64*Pi^2) - (msq + λ*ϕ^2)^(3/2)/(12*Pi) +
  (g1^2*Yϕ^2*Sqrt[g1^2*Yϕ^2*ϕ^2]*Sqrt[msq + 3*λ*ϕ^2])/(16*Pi^2) + (λ*Sqrt[msq
  + λ*ϕ^2]*Sqrt[msq + 3*λ*ϕ^2])/(32*Pi^2) + (3*λ*(msq + 3*λ*ϕ^2))/(64*Pi^2)
  - (msq + 3*λ*ϕ^2)^(3/2)/(12*Pi) - (3*λ^2*ϕ^2*(1/2 + Log[μ3US/(3*Sqrt[msq +
  3*λ*ϕ^2])]))/(16*Pi^2) + ((g1^4*Yϕ^4*ϕ^4)/(8*Pi^2) - (g1^2*Yϕ^2*ϕ^2*(-msq +
  2*g1^2*Yϕ^2*ϕ^2 - 3*λ*ϕ^2))/(16*Pi^2) + (g1^2*Yϕ^2*ϕ^2*Sqrt[g1^2*Yϕ^2*ϕ^2]*Sqrt[msq
  + 3*λ*ϕ^2])/(8*Pi^2) - ((msq + 3*λ*ϕ^2)^2*(1/2 + Log[μ3US/Sqrt[msq +
  3*λ*ϕ^2]]))/(16*Pi^2) + ((-msq + g1^2*Yϕ^2*ϕ^2 - 3*λ*ϕ^2)^2*(1/2 +
  Log[μ3US/(Sqrt[g1^2*Yϕ^2*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(8*Pi^2) - ((7*g1^4*Yϕ^4*ϕ^4
  + (-msq + g1^2*Yϕ^2*ϕ^2 - 3*λ*ϕ^2)^2 - 2*g1^2*Yϕ^2*ϕ^2*(msq + 3*λ*ϕ^2))*(1/2 +
  Log[μ3US/(2*Sqrt[g1^2*Yϕ^2*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(16*Pi^2))/(4*ϕ^2) +
  (-1/16*(Sqrt[g1^2*Yϕ^2*ϕ^2]*(g1^2*Yϕ^2*ϕ^2 - 2*λ*ϕ^2)*Sqrt[msq + λ*ϕ^2])/Pi^2
  + (Sqrt[msq + 3*λ*ϕ^2]*((g1^2*Yϕ^2*ϕ^2*Sqrt[msq + λ*ϕ^2])/(4*Pi) -
  (Sqrt[g1^2*Yϕ^2*ϕ^2]*(g1^2*Yϕ^2*ϕ^2 + 2*λ*ϕ^2))/(4*Pi)))/(4*Pi) + (λ^2*ϕ^4*(1/2 +
  Log[μ3US/(Sqrt[msq + λ*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(4*Pi^2) - ((g1^4*Yϕ^4*ϕ^4 +
  4*λ^2*ϕ^4 - 2*g1^2*Yϕ^2*ϕ^2*(2*msq + 4*λ*ϕ^2))*(1/2 + Log[μ3US/(Sqrt[g1^2*Yϕ^2*ϕ^2]
  + Sqrt[msq + λ*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(16*Pi^2))/(4*ϕ^2) +
  (-1/16*(Sqrt[g1^2*Yϕ^2*ϕ^2]*(g1^2*Yϕ^2*ϕ^2 + 2*λ*ϕ^2)*Sqrt[msq + 3*λ*ϕ^2])/Pi^2
  + (Sqrt[msq + λ*ϕ^2]*(-1/4*(Sqrt[g1^2*Yϕ^2*ϕ^2]*(g1^2*Yϕ^2*ϕ^2 - 2*λ*ϕ^2))/Pi
  + (g1^2*Yϕ^2*ϕ^2*Sqrt[msq + 3*λ*ϕ^2])/(4*Pi)))/(4*Pi) + (λ^2*ϕ^4*(1/2 +
  Log[μ3US/(Sqrt[msq + λ*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(4*Pi^2) - ((g1^4*Yϕ^4*ϕ^4 +
  4*λ^2*ϕ^4 - 2*g1^2*Yϕ^2*ϕ^2*(2*msq + 4*λ*ϕ^2))*(1/2 + Log[μ3US/(Sqrt[g1^2*Yϕ^2*ϕ^2]
  + Sqrt[msq + λ*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(16*Pi^2))/(4*ϕ^2) -
  (λ^2*ϕ^2*(1/2 + Log[μ3US/(2*Sqrt[msq + λ*ϕ^2] + Sqrt[msq + 3*λ*ϕ^2])]))/(16*Pi^2)
```

In[136]:=

```
expr = ToExpression[rhs /. {"ϕ" → ϕ, "Yϕ" → Yϕ}];

ClearAll[V];
V[ϕ_, T_] := expr;
```

In[156]:=
```
(*tiny floor for logs*)eps = 10.^-30;

exprSafe =
  expr /. {(*sqrt(x)→sqrt(|x|)*)Sqrt[x_] :> Sqrt[Abs[x]], (*log(x)→log(max(|x|,eps))*)
    Log[x_] :> Log[Max[Abs[x], eps]], (*any half-integer power x^(n/2)→ |x|^(n/2)*)
    Power[x_, p_Rational] /; Denominator[p] == 2 :> Power[Abs[x], p]};
```

In[269]:=
```
Clear[V];
V[ϕ_, T_] := exprSafe;
```

In[325]:=
```
(*4D input couplings*)g1Four = 0.54;
lam4 = 1.65 * 10^-3;
m0sq4 = -(130.^2);
Yphi4 = 1.0;

(*simple 4D→3D matching,as in Python*)
g1sq3[T_] := g1Four^2 * T;
lam3[T_] := lam4 * T;
msq3[T_] := m0sq4 + (0.30 * g1Four^2 + 0.10 * lam4) * T^2;
mu3USfun[T_] := Max[g1Four * T, 1.0];
g13[T_] := Sqrt[g1sq3[T]];


ClearAll[parRules];
```

In[335]:=
```
parRules[T_] := {g1 → g13[T], λ → lam3[T], msq → msq3[T], Yϕ → Yphi4, μ3US → mu3USfun[T]};
```

In[336]:=
```
ClearAll[Vnum, dVnum];

Vnum[phi_?NumericQ, T_?NumericQ] :=
  Module[{ex}, ex = exprSafe /. parRules[T] /. ϕ → phi;
   N[ex]];

dVnum[phi_?NumericQ, T_?NumericQ] := Module[{h}, h = 10^-3 * Max[1., Abs[phi]] + 10^-5;
    (Vnum[phi + h, T] - Vnum[phi - h, T]) / (2 h)];
```

In[342]:=
```
Vnum[10., 0.]
Vnum[10., 150.]
```

Out[342]=
```
-961 554.
```

Out[343]=
```
-845 177.
```

In[380]:=
```
Vnum[10, 150]
dVnum[10, 150]
```

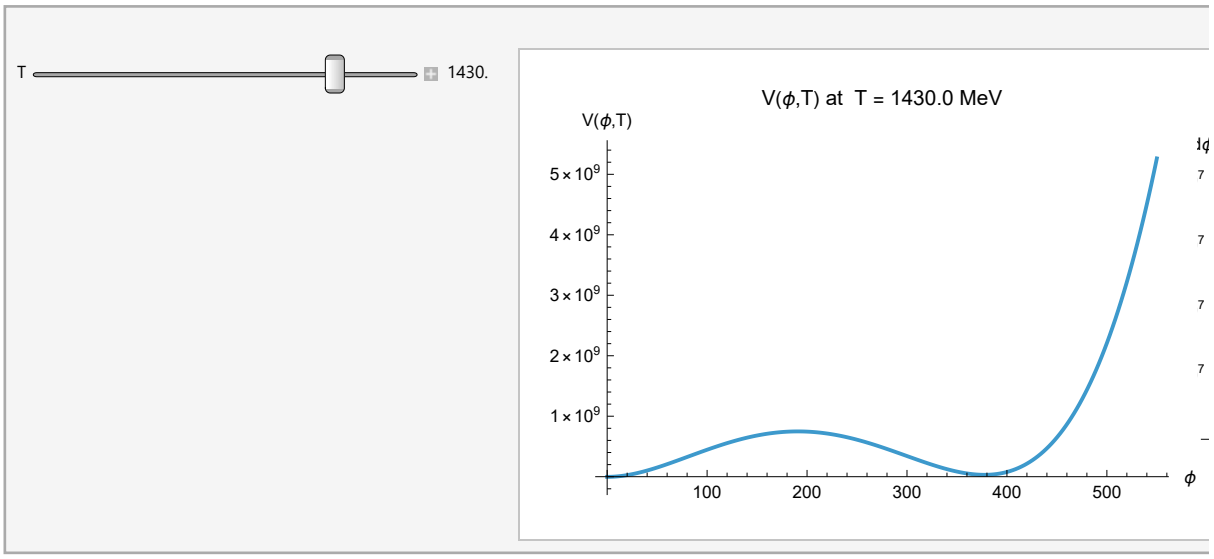
Out[380]=
```
-845 177.
```

Out[381]=
```
-152 752.
```

In[422]:=
```
vev = 500.;
phiMin = 0.;
phiMax = 1.1 * vev;
step = 1.;  (*sampling resolution for ListLinePlot*)

Manipulate[Module[{vPlot, dVdata, dVPlot}, (*1) V vs phi (standard Plot)*)
  vPlot = Plot[Vnum[ϕ, T], {ϕ, phiMin, phiMax}, PlotRange → All,
    PlotPoints → 200, AxesLabel → {"ϕ", "V(ϕ,T)"}, PlotLabel →
     Row[{"V(ϕ,T) at  T = ", NumberForm[T, {5, 1}], " MeV"}], ImageSize → 400];
  (*2) Sample derivative explicitly,then ListLinePlot*)
  dVdata = Table[{phi, dVnum[phi, T]}, {phi, 1., phiMax, step}];
  dVPlot =
   ListLinePlot[dVdata, PlotRange → All, AxesLabel → {"ϕ", "dV/dϕ (ϕ,T)"}, PlotLabel →
     Row[{"dV/dϕ at  T = ", NumberForm[T, {5, 1}], " MeV"}], ImageSize → 300];
  GraphicsRow[{vPlot, dVPlot}, Spacings → 2]],
 {{T, 0.}, 0., 3.5 * vev, Appearance → "Labeled"}]
```

Out[426]=



PacletInstall: No appropriate paclet named WolframLanguageForJupyter is available for download from any currently enabled paclet sites.