(Debug) In[1]:=

```
Print[];
Print[" The data for the paper  "]
Print[" Structure and energetics of carbon, hexagonal boron nitride, "];
Print[" and carbon/hexagonal boron nitride single-layer and bilayer nanoscrolls "];
Print[" / A.I. Siahlo, N.A. Poklonski, A.V. Lebedev, I.V. Lebedeva, A.M. Popov, S.A. Vyrk
Print[" // Phys. Rev. Materials.— 2018.— V. 2, № 3.— P. 036001 (9 pp.). [DOI: 10.1103/Phy
Print[" ----------------------------------------------------"];
Print[" I. All Input Parameters and Constants--------------"];
Print[" I.1.The dimensions"];
eV=1;atom=1;nm=1;
AA = 0.1 nm; m = 10^9 nm; meV = eV/1000;
Print[" I.2. The Input Geometry Parameters"];
NumberOfLayers1 = 1; NumberOfLayers2 = 2;
NumberOfLayersp = NumberOfLayers1;
Print[" Number of the layers in carbon nanoscroll NumberOfLayers=",
  NumberOfLayersp];
L15nm = 15 nm; L1p = L15nm;
Print[" The length of a carbon nanoribbon L1=", L1p/nm, "nm"];
w1nm = 1 nm; wp = w1nm; wdth = w1nm;
Print[" The carbon nanoribbon width w=", wp/nm, "nm"];
RIn1nm = 1.1 nm;
RIn2nm = 2.047 nm;
RIn1p = RIn2nm;
Print[" The inner radius of the nanoscroll RIn1=", RIn1p/nm, "nm"];
Print[" I.3. The Input Energy Constants"];
Print[" eps - the interlayer interaction energy per one atom of"];
Print[" the nanoscroll:"];
eps35 = 35.0 meV/atom; epsp = eps35;
Print[" eps=", epsp/(eV/atom), "eV/atom"];
Print[" C - the bending elastic constant:"];
C201 = 2.01 eV AA^2/atom;
CBN1328 = 1.328 eV AA^2/atom;
CCp = C201;
CBNp = CBN1328;
CBNp = CCp;
Print[" CCelast=", CCp/(eV AA^2/atom), "eV AA^2/atom"];
Print[" CCBNelast=", CBNp/(eV AA^2/atom), "eV AA^2/atom"];
Print[" I.4.The Input Geometry constants--------------"];
Print[" The interatomic distance aCC and the interlayer distance h"];
aCC142AA = 1.42 AA; aCCp = aCC142AA;
h335nm = 0.3354 nm; hp = h335nm;
Print[" aCC=",aCCp/nm,"nm, h=", hp/nm, "nm"];
Print[" dPhi12 - The difference of the inner angles of the spirales"];
Print["   of the Layers"];
dPhi12eq0 = 0.0 Pi;
dPhi12eqPi = 1.0 Pi;
dPhi12LowLeq0 = 0.0 Pi;
dPhi12HighLeqPi = Pi;
dPhi12p = 0.0 Pi;
dPhi120 = 0.0 Pi;
dPhi12Pi03 = 0.3 Pi;
dPhi12Pi05 = 0.5 Pi;
dPhi12Pi07 = 0.7 Pi;
dPhi12Pi=Pi;
dPhi12HighLp = Pi;
Print[" dPhi12=", dPhi12p/(2 Pi), "(2Pi),for the high L dPhi12HighL=",
  dPhi12HighLp/(2 Pi), "(2Pi)"];
Print[" I.5.The parameters for the visualisation"];
RIn1MinMonoScroll = hp/5;
RIn1MinBiScroll = hp/5;
```

```
RIn1MaxMonoScroll = 4 nm;
RIn1MaxBiScroll = 8 nm;
PlotRangeMonoScroll = {-4eV/atom, 12eV/atom};
PlotRangeBiScroll = {-10eV/atom, 30eV/atom};
ShowSpirales = True;
ShowThePlot = True;
Print[" I.6. The parameters of the output file"];
NanoscrollNamep=StringJoin["Nanoscroll",ToString[NumberOfLayersp],"L",ToString[L1p/nm],"n
Print[" NanoscrollName=",NanoscrollNamep];
CarbonNanoscrollEnergyVsRInFileName=StringJoin[NanoscrollNamep,".txt"];
Print[CarbonNanoscrollEnergyVsRInFileName];
Print[" (The output of the data to a file Is Not Performed)"];
npRIn1=1000;
Print[" The number of the output points = ",npRIn1];
Print[" I.7. The Input Numerical Constants used in the programm"];
Print[" The Indexes used for the work with EVdW[...] function"];
iEVdW = 1; iEVdW1Un1 = 2; iEVdW1Ov1 = 3; iEVdW1Un2 = 4; iEVdW1Ov2 = 5;
5; iEVdW2Un1 = 6; iEVdW2Ov1 = 7;
Print[" --------End Of The Input---------------"];
Print[" II. The derivated parameters and the functions required"];
Print[" II.1. The derivated parameters"];
fSa[aCC_] := aCC^2 3 Sqrt[3]/4; fSa[aCCp]; Sap = fSa[aCCp];
Print[" The cell area Sa=", fSa[aCC], "=", Sap/nm^2, "nm^2"];
Print[" II.2. The required functions--------------"];
Print[" II.2.1. The function"];
Print[" fSpiraleLen[NumberOfLayers,PhiIn, PhiOut, h]"];
Print[" defines the Length of a Spirale with the inner agle PhiIn and
the outer angle PhiOut"];
fSpiraleLen[NumberOfLayersv_,PhiInv_, PhiOutv_, hv_] :=
  UnitStep[PhiOutv -
     PhiInv] (1/(4 Pi) hv NumberOfLayersv (-PhiInv Sqrt[1 + PhiInv^2] +
       PhiOutv Sqrt[1 + PhiOutv^2] - ArcSinh[PhiInv] +
       ArcSinh[PhiOutv]));
Print[" II.2.2. The function fElast[PhiIn,PhiOut] is required to
calculate an elastic energy "];
fElast[PhiInv_,
   PhiOutv_] := (Sqrt[PhiInv^2 + 1]/PhiInv -
    Sqrt[PhiOutv^2 + 1]/PhiOutv - ArcSinh[PhiInv] + ArcSinh[PhiOutv]);
Print[" fElast[PhiIn,PhiOut]=", fElast[PhiIn, PhiOut]];
Print[" II.2.3. The function fPhiOutvsPhiInLh[NumberOfLayers,PhiIn,L,h] is a
 good approximation"];
Print[" to obtain the value of PhiOut for the defined PhiIn,L,h "];
fPhiOutvsPhiInLh[NumberOfLayersv_,PhiInv_, Lv_, hv_] :=
  Sqrt[4 π Lv/(NumberOfLayersv hv) + PhiInv^2];
Print[" fPhiOutvsPhiInLh[NumberOfLayers,PhiIn,L,h]=",
  fPhiOutvsPhiInLh[NumberOfLayers,PhiIn, L, h]];
Print["   and the inverse function fPhiInvsPhiOutLh[NumberOfLayers,PhiOut, L, h]:"];
fPhiInvsPhiOutLh[NumberOfLayersv_,PhiOutv_, Lv_, hv_] :=
  Sqrt[PhiOutv^2-4 π Lv/(NumberOfLayersv hv)];
Print[" fPhiInvsPhiOutLh[NumberOfLayers,PhiOut,L,h]=",
 fPhiInvsPhiOutLh[NumberOfLayers,PhiOut, L, h]];
Print[" III. Begin of Calculation "];
Print[" III.1. The inner and the outer angles of the spirales"];
Print[" PhiIn1=RIn1 2 Pi/(NumberOfLayers h),
PhiOut1=fPhiOutvsPhiInLh[NumberOfLayers,,PhiIn1,L1,h]."];
Print[" For RIn1=", RIn1p/nm, "nm,h=", hp/nm, "nm,NumberOfLayers=",
  NumberOfLayersp, ":"];
fPhiIn1[NumberOfLayersv_, RIn1v_, hv_] := RIn1v 2 Pi/(NumberOfLayersv hv);
PhiIn1p = fPhiIn1[NumberOfLayersp, RIn1p, hp];
fPhiOut1[NumberOfLayersv_, L1v_, RIn1v_, hv_] :=
  fPhiOutvsPhiInLh[NumberOfLayersv,fPhiIn1[NumberOfLayersv, RIn1v, hv], L1v, hv];
```

```
PhiOut1p = fPhiOut1[NumberOfLayersp, L1p, RIn1p, hp];
ROut1p = PhiOut1p NumberOfLayersp hp/(2 Pi);
Print[" PhiIn1=", PhiIn1p/(2 Pi), "(2Pi),PhiOut1=", PhiOut1p/(2 Pi),
  "(2Pi)"];
fPhiIn2[NumberOfLayersv_, RIn1v_, hv_, dPhi12v_] :=
  fPhiIn1[NumberOfLayersv, RIn1v, hv] + dPhi12v;
PhiIn2dPhi120p =
  fPhiIn2[NumberOfLayersp, RIn1p, hp,0];
PhiIn2dPhi12Pip =
  fPhiIn2[NumberOfLayersp, RIn1p, hp,Pi];
fPhiOut2[NumberOfLayersv_, L1v_, RIn1v_, hv_, dPhi12v_] :=
  fPhiOutvsPhiInLh[NumberOfLayersv,fPhiIn2[NumberOfLayersv, RIn1v, hv, dPhi12v], L1v, hv]
PhiOut2dPhi120p =
  fPhiOut2[NumberOfLayersp, L1p, RIn1p, hp,0];
Print[" for dPhi12=0: PhiIn2=", PhiIn2dPhi120p/(2 Pi), "(2Pi),PhiOut2=", PhiOut2dPhi120p/
  "(2Pi)"];
PhiOut2dPhi12Pip =
  fPhiOut2[NumberOfLayersp, L1p, RIn1p, hp,Pi];
Print[" for dPhi12=Pi: PhiIn2=", PhiIn2dPhi12Pip/(2 Pi), "(2Pi),PhiOut2=", PhiOut2dPhi12P
  "(2Pi)"];
Print[" Plot Spirales of the layers for dPhi12=0 and dPhi12=Pi"];
Spirale1Plot =
  PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv, PhiIn1p,
    PhiOut1p},
   PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
      1.1 ROut1p/nm}}, PlotStyle -> {Red, Thin}, Axes -> None];
Spirale2dPhi120Plot =
  If[NumberOfLayersp == 2,
   PolarPlot[(Phiv - Pi) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
     PhiIn2dPhi120p + Pi, PhiOut2dPhi120p + Pi},
    PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
       1.1 ROut1p/nm}}, PlotStyle -> {Blue, Thin}, Axes -> None], {}];
Spirale2dPhi12PiPlot =
  If[NumberOfLayersp == 2,
   PolarPlot[(Phiv - Pi) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
     PhiIn2dPhi12Pip + Pi, PhiOut2dPhi12Pip + Pi},
    PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
       1.1 ROut1p/nm}}, PlotStyle -> {Blue, Thin}, Axes -> None], {}];
Print[Show[{Spirale1Plot, Spirale2dPhi120Plot}],Show[{Spirale1Plot, Spirale2dPhi12PiPlot}
If[NumberOfLayersp == 1,
  Spirale1OverSpirale1Plot =
   If[PhiIn1p + 2 Pi < PhiOut1p,
    PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv, PhiIn1p + 2 Pi,
      PhiOut1p},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}, PlotStyle -> {Red, Thick}, Axes -> None], {}];
  Spirale1UnderSpirale1Plot =
   If[PhiIn1p < PhiOut1p - 2 Pi,
    PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv, PhiIn1p,
      PhiOut1p - 2 Pi},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}, PlotStyle -> {Red, Thick}, Axes -> None], {}];
  Print[" {Spirale,
Spirale1UnderSpirale1},{Spirale1,Spirale1OverSpirale1}"];
  Print[Show[{Spirale1Plot, Spirale1UnderSpirale1Plot}],
   Show[{Spirale1Plot, Spirale1OverSpirale1Plot}]];];
If[NumberOfLayersp == 2,
Spirale1UnderSpirale2dPhi120Plot =
   If[PhiIn1p < PhiOut2dPhi120p - Pi,
    PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv, PhiIn1p,
      PhiOut2dPhi120p - Pi}, PlotStyle -> {Red, Thick},
```

```
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale1OverSpirale2dPhi120Plot =
   If[PhiIn1p + Pi +
      dPhi120 < PhiOut1p,
    PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
      PhiIn1p + Pi +
       dPhi120, PhiOut1p},
     PlotStyle -> {Red, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale2UnderSpirale1dPhi120Plot =
   If[PhiIn2dPhi120p + Pi < PhiOut1p,
    PolarPlot[(Phiv - Pi) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
      PhiIn2dPhi120p + Pi, PhiOut1p}, PlotStyle -> {Blue, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale2OverSpirale1dPhi120Plot =
   If[2 Pi + PhiIn2dPhi120p -
      dPhi120 <
     PhiOut2dPhi120p + Pi,
    PolarPlot[(Phiv - Pi) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
      2 Pi + PhiIn2dPhi120p -
       dPhi120,
      PhiOut2dPhi120p + Pi}, PlotStyle -> {Blue, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale1UnderSpirale2dPhi12PiPlot =
   If[PhiIn1p < PhiOut2dPhi12Pip - Pi,
    PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv, PhiIn1p,
      PhiOut2dPhi12Pip - Pi}, PlotStyle -> {Red, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale1OverSpirale2dPhi12PiPlot =
   If[PhiIn1p + Pi +
      dPhi12Pi < PhiOut1p,
    PolarPlot[(Phiv) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
      PhiIn1p + Pi +
       dPhi12Pi, PhiOut1p},
     PlotStyle -> {Red, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale2UnderSpirale1dPhi12PiPlot =
   If[PhiIn2dPhi120p + Pi < PhiOut1p,
    PolarPlot[(Phiv - Pi) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
      PhiIn2dPhi12Pip + Pi, PhiOut1p}, PlotStyle -> {Blue, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];
Spirale2OverSpirale1dPhi12PiPlot =
   If[2 Pi + PhiIn2dPhi12Pip -
      dPhi12Pi <
     PhiOut2dPhi12Pip + Pi,
    PolarPlot[(Phiv - Pi) NumberOfLayersp hp/(2 Pi)/nm, {Phiv,
      2 Pi + PhiIn2dPhi12Pip -
       dPhi12Pi,
      PhiOut2dPhi12Pip + Pi}, PlotStyle -> {Blue, Thick},
     PlotRange -> {{-1.1 ROut1p/nm, 1.1 ROut1p/nm}, {-1.1 ROut1p/nm,
        1.1 ROut1p/nm}}], {}];

  Print[" {Spirale1,Spirale2,Spirale1UnderSpirale2,Spirale2UnderSpirale1},"];
  Print["     {Spirale1,Spirale2,Spirale1OverSpirale2,Spirale2OverSpirale1}"];
```

```
  Print[" for dPhi12=0: ",Show[Spirale1Plot, Spirale2dPhi120Plot],
   Show[Spirale1Plot, Spirale2dPhi120Plot, Spirale1UnderSpirale2dPhi120Plot,
    Spirale2UnderSpirale1dPhi120Plot],
   Show[Spirale1Plot, Spirale2dPhi120Plot, Spirale1OverSpirale2dPhi120Plot,
    Spirale2OverSpirale1dPhi120Plot]];
Print[" for dPhi12=Pi: ",Show[Spirale1Plot, Spirale2dPhi12PiPlot],
   Show[Spirale1Plot, Spirale2dPhi12PiPlot, Spirale1UnderSpirale2dPhi12PiPlot,
    Spirale2UnderSpirale1dPhi12PiPlot],
   Show[Spirale1Plot, Spirale2dPhi12PiPlot, Spirale1OverSpirale2dPhi12PiPlot,
    Spirale2OverSpirale1dPhi12PiPlot]];
];
Print[" III.2. The nanoscroll energy calculation"];
Print[" III.2.1. The elastic energy calculation"];
fEelastCC[NumberOfLayersv_, L1v_, RIn1v_, hv_, aCCv_, CCv_] :=
  Module[{},
   Return[2 Pi CCv wdth/(hv fSa[aCCv]) fElast[
       fPhiIn1[NumberOfLayersv, RIn1v, hv],
       fPhiOut1[NumberOfLayersv, L1v, RIn1v, hv]]];];
fEelastCBN[NumberOfLayersv_, L1v_, RIn1v_, hv_, aCCv_, CBNv_] :=
  Module[{},
   Return[2 Pi CBNv wdth/(hv fSa[aCCv]) fElast[
       fPhiIn1[NumberOfLayersv, RIn1v, hv],
       fPhiOut1[NumberOfLayersv, L1v, RIn1v, hv]]];];
EelastCCp = fEelastCC[NumberOfLayersp, L1p, RIn1p, hp, aCCp, CCp];
EelastCBNp = fEelastCBN[NumberOfLayersp, L1p, RIn1p, hp, aCCp, CBNp];
Print[" EelastC=", EelastCCp/(eV/atom), "eV/atom"];
Print[" EelastBN=", EelastCBNp/(eV/atom), "eV/atom"];
Print[" III.2.2. The Van-der-Waals energy calculation"];
fEVdWdPhi12[NumberOfLayersv_, L1v_, RIn1v_, hv_, aCCv_, epsv_, dPhi12v_] :=
   Module[{EVdW, PhiIn1, PhiIn2, PhiOut1, PhiOut2,
    EVdW1Un1=0 (eV/atom), EVdW1Ov1=0 (eV/atom),
    EVdW1Un2=0 (eV/atom), EVdW1Ov2=0 (eV/atom),
    EVdW2Un1=0 (eV/atom), EVdW2Ov1=0 (eV/atom),
    Spirale1UnderSpirale1Length=0 nm, Spirale1OverSpirale1Length=0 nm,
    Spirale1UnderSpirale2Length=0 nm, Spirale1OverSpirale2Length=0 nm,
    Spirale2UnderSpirale1Length=0 nm, Spirale2OverSpirale1Length=0 nm},
   PhiIn1 = fPhiIn1[NumberOfLayersv, RIn1v, hv];
   PhiIn2 =
    fPhiIn2[NumberOfLayersv, RIn1v, hv, dPhi12v];
   PhiOut1 = fPhiOut1[NumberOfLayersv, L1v, RIn1v, hv];
   PhiOut2 =
    fPhiOut2[NumberOfLayersv, L1v, RIn1v, hv, dPhi12v];
   EVdW =
    Switch[NumberOfLayersv, 1,
     Spirale1UnderSpirale1Length =
      If[PhiIn1 < PhiOut1 - 2 Pi,
       fSpiraleLen[NumberOfLayersv,PhiIn1, PhiOut1 - 2 Pi,hv], 0 nm];
     Spirale1OverSpirale1Length =
      If[PhiIn1 + 2 Pi < PhiOut1,
       fSpiraleLen[NumberOfLayersv,PhiIn1 + 2 Pi, PhiOut1,hv], 0 nm];
     EVdW1Un1 = -epsv wdth/(2 fSa[aCCv]) Spirale1UnderSpirale1Length;
     EVdW1Ov1 = -epsv wdth/(2 fSa[aCCv]) Spirale1OverSpirale1Length;
     (EVdW1Un1 + EVdW1Ov1), 2,
     Spirale1UnderSpirale2Length =
      If[PhiIn1 < PhiOut2 - Pi,
       fSpiraleLen[NumberOfLayersv,PhiIn1, PhiOut2 - Pi,hv], 0 nm];
     Spirale1OverSpirale2Length =
      If[PhiIn1 + Pi +dPhi12v<PhiOut1,
       fSpiraleLen[NumberOfLayersv,
        PhiIn1 + Pi +dPhi12v,
        PhiOut1,hv], 0 nm];
```

```
     Spirale2UnderSpirale1Length =
      If[PhiIn1 + dPhi12v <
        PhiOut1 - Pi,
       fSpiraleLen[NumberOfLayersv,
        PhiIn1 + dPhi12v,
        PhiOut1 - Pi,hv], 0 nm];
     Spirale2OverSpirale1Length =
      If[PhiIn1 - dPhi12v + Pi <
         PhiOut2 - dPhi12v,
       fSpiraleLen[NumberOfLayersv,
        PhiIn1 -dPhi12v+ Pi,
        PhiOut2 - dPhi12v,
       hv], 0 nm];
     EVdW1Un2 = -epsv wdth/(2 fSa[aCCv]) Spirale1UnderSpirale2Length;
     EVdW1Ov2 = -epsv wdth/(2 fSa[aCCv]) Spirale1OverSpirale2Length;
     EVdW2Un1 = -epsv wdth/(2 fSa[aCCv]) Spirale2UnderSpirale1Length;
     EVdW2Ov1 = -epsv wdth/(2 fSa[aCCv]) Spirale2OverSpirale1Length;
     "EVdW=(EVdW1Un2+EVdW1Ov2+EVdW2Un1+EVdW2Ov1)";
     (EVdW1Un2 + EVdW1Ov2 + EVdW2Un1 + EVdW2Ov1)];
   Return[{EVdW, EVdW1Un1, EVdW1Ov1, EVdW1Un2, EVdW1Ov2, EVdW2Un1,
     EVdW2Ov1}];];
EVdWvardPhi12allp =
  fEVdWdPhi12[NumberOfLayersp, L1p, RIn1p, hp, aCCp, epsp, dPhi12p];
Print[" for dPhi12=",dPhi12p/Pi,"Pi EVdWvardPhi12allp[[iEVdW]]=",
  EVdWvardPhi12allp[[iEVdW]]/(eV/atom), "eV/atom"];
If[NumberOfLayersp == 1,
  Print[" EVdWvardPhi12allp[[iEVdW1Un1]]=",
   EVdWvardPhi12allp[[iEVdW1Un1]]/(eV/atom), "eV/atom"];
  Print[" EVdWvardPhi12allp[[iEVdW1Ov1]]=",
   EVdWvardPhi12allp[[iEVdW1Ov1]]/(eV/atom), "eV/atom"];];
If[NumberOfLayersp == 2,
  EVdWdPhi12eq0allp =
   fEVdWdPhi12[NumberOfLayersp, L1p, RIn1p, hp, aCCp, epsp, dPhi12eq0];
  Print[" For dPhi12=", dPhi12eq0/Pi, "Pi:"];
  Print[" EVdWvardPhi12allp[[iEVdW]]=",
   EVdWdPhi12eq0allp[[iEVdW]]/(eV/atom), "eV/atom"];
  Print[" EVdWvardPhi12allp[[iEVdW1Un2]]=",
   EVdWdPhi12eq0allp[[iEVdW1Un2]]/(eV/atom), "eV/atom"];
  Print[" EVdWvardPhi12allp[[iEVdW1Ov2]]=",
   EVdWdPhi12eq0allp[[iEVdW1Ov2]]/(eV/atom), "eV/atom"];
  Print[" EVdWvardPhi12allp[[iEVdW2Un1]]=",
   EVdWdPhi12eq0allp[[iEVdW2Un1]]/(eV/atom), "eV/atom"];
  Print[" EVdWvardPhi12allp[[iEVdW2Ov2]]=",
   EVdWdPhi12eq0allp[[iEVdW2Ov1]]/(eV/atom), "eV/atom"];
  EVdWdPhi12eqPiallp =
   fEVdWdPhi12[NumberOfLayersp, L1p, RIn1p, hp, aCCp, epsp, dPhi12eqPi];
  Print[" For dPhi12=", dPhi12eqPi/Pi, "Pi:"];
  Print[" EVdWvatdPhi12allp[[iEVdW]]=",
   EVdWdPhi12eqPiallp[[iEVdW]]/(eV/atom), "eV/atom"];
  Print[" EVdWvatdPhi12allp[[iEVdW1Un2]]=",
   EVdWdPhi12eqPiallp[[iEVdW1Un2]]/(eV/atom), "eV/atom"];
  Print[" EVdWvatdPhi12allp[[iEVdW1Ov2]]=",
   EVdWdPhi12eqPiallp[[iEVdW1Ov2]]/(eV/atom), "eV/atom"];
  Print[" EVdWvatdPhi12allp[[iEVdW2Un1]]=",
   EVdWdPhi12eqPiallp[[iEVdW2Un1]]/(eV/atom), "eV/atom"];
  Print[" EVdWvatdPhi12allp[[iEVdW2Ov2]]=",
   EVdWdPhi12eqPiallp[[iEVdW2Ov1]]/(eV/atom), "eV/atom"];
  EVdWEVdWdPhi12eq0p = EVdWdPhi12eq0allp[[iEVdW]];
  Print[" EVdWdPhi12eq0allp=", EVdWdPhi12eq0allp/(eV/atom),
   "eV/atom"];
  EVdWEVdWdPhi12eqPip = EVdWvardPhi12allp[[iEVdW]];
```

```
  Print[" EVdWEVdWdPhi12eqPip=", EVdWEVdWdPhi12eqPip/(eV/atom),
   "eV/atom"];];
Print[" III.3. The energy of flat places "];
fEnergyFlatPlates[NumberOfLayersv_, L1v_, aCCv_,
   epsv_]:=If[NumberOfLayersv==2,-epsv wdth/fSa[aCCv] L1v,0 eV/atom];
EnergyFlatPlatesp = fEnergyFlatPlates[NumberOfLayersp, L1p, aCCp, epsp];
Print[" EnergyFlatPlates=-eps width/Sa L1(NumberOfLayers-1) =",
  EnergyFlatPlatesp/(eV/atom), "eV/atom"];

Print[" III.4. The total energy of the nanoscroll"];
fScrollEnergydPhi[NumberOfLayersv_, L1v_, RIn1v_, hv_, aCCv_, epsv_, CCv_,
  CBNv_, dPhi12v_] :=
 Module[{ScrollEnergyv, EVdWv, EVdWt, iL1},
  EVdWv = fEVdWdPhi12[NumberOfLayersv, L1v, RIn1v, hv, aCCv, epsv,
     dPhi12v][[1]];
  If[NumberOfLayersv == 1,
   ScrollEnergyv =
    EVdWv + fEelastCC[NumberOfLayersv, L1v, RIn1v, hv, aCCv, CCv]];
  If[NumberOfLayersv == 2,
   ScrollEnergyv =
    EVdWv + fEelastCC[NumberOfLayersv, L1v, RIn1v, hv, aCCv, CCv] +
     fEelastCBN[NumberOfLayersv, L1v, RIn1v, hv, aCCv, CCv]];
  Return[ScrollEnergyv];];
fScrollEnergy[NumberOfLayersv_, L1v_, RIn1v_, hv_, aCCv_, epsv_, CCv_,
  CBNv_] :=
 Module[{ScrollEnergyv, EVdWv, EVdWt, iL1,
ScrollEnergydPhieq0v,ScrollEnergydPhieqPiv,
dPhi12eq0=0,dPhi12eqPi=Pi},
ScrollEnergydPhieq0v=fScrollEnergydPhi[NumberOfLayersv, L1v, RIn1v, hv, aCCv, epsv, CCv,
  CBNv, dPhi12eq0];
 ScrollEnergydPhieqPiv=fScrollEnergydPhi[NumberOfLayersv, L1v, RIn1v, hv, aCCv, epsv, CCv
  CBNv, dPhi12eqPi];
  ScrollEnergyv=Min[ScrollEnergydPhieq0v/(eV/atom),ScrollEnergydPhieqPiv/(eV/atom)](eV/atc
  Return[ScrollEnergyv];]
Print[" III.5. Determine the inner angles mismatch for the bi-layer nanoscroll
       for the high nanoribbon Length"];
ScrollEnergydPhi12Pip0p =
  fScrollEnergydPhi[NumberOfLayersp, L1p, RIn1p, hp, aCCp, epsp, CCp, CBNp,0];
ScrollEnergydPhi12PipPip =
  fScrollEnergydPhi[NumberOfLayersp, L1p, RIn1p, hp, aCCp, epsp, CCp, CBNp,Pi];
Print[" For L1=", L1p/nm, "nm,RIn=", RIn1p/nm, "nm,h=", hp/nm, "nm  and  dPhi12=0:"];
Print[" ScrollEnergy=", ScrollEnergydPhi12Pip0p/(eV/atom), "eV/atom"];
Print[" For L1=", L1p/nm, "nm,RIn=", RIn1p/nm, "nm,h=", hp/nm, "nm  and  dPhi12=Pi:"];
Print[" ScrollEnergy=", ScrollEnergydPhi12Pip/(eV/atom), "eV/atom"];

Print[" IV.The potential energy of the scroll"];
Print[" as a function of the inner radius RIn"];
PlotRangep =
  Switch[NumberOfLayersp, 1, PlotRangeMonoScroll, 2, PlotRangeBiScroll];
RIn1Minp =
  Switch[NumberOfLayersp, 1, RIn1MinMonoScroll, 2, RIn1MinBiScroll];
RIn1Maxp =
  Switch[NumberOfLayersp, 1, RIn1MaxMonoScroll, 2, RIn1MaxBiScroll];
PlotRangep =
  Switch[NumberOfLayersp, 1, PlotRangeMonoScroll, 2, PlotRangeBiScroll];
RIn1Maxp =
  Switch[NumberOfLayersp, 1, RIn1MaxMonoScroll, 2, RIn1MaxBiScroll];
tL1 = Switch[NumberOfLayersp, 1, {7 nm, 10 nm, 12.5 nm, 15 nm},
   2, {15 nm, 20 nm, 25 nm, 30 nm}];
Print[" NumberOfLayers=", NumberOfLayersp];
Print[" eps=", epsp/(eV/atom), "eV/atom,C=", CCp/(eV nm^2/atom),
```

```
  "(eV nm^2/atom),aCC=", aCCp, "nm,h=", hp/nm, "nm"];
Print[" Plot ScrollEnergy[RIn1/nm]/(eV/atom) for L1=", L1p/nm,
  "nm (NumberOfLayers=", NumberOfLayersp, ",w=", wdth/nm, "nm)"];
PlotScrollEnergyVsRIn1 = Plot[(fScrollEnergy[NumberOfLayersp, L1p, RIn1nmv nm, hp, aCCp,
        CCp, CBNp] -
       fEnergyFlatPlates[NumberOfLayersp, L1p, aCCp, epsp])/(eV/
        atom), {RIn1nmv, RIn1Minp/nm, RIn1Maxp/nm}, PlotRange -> PlotRangep/(eV/atom)];
Print[PlotScrollEnergyVsRIn1];
Print[" Plot ScrollEnergy[RIn1/nm]/(eV/atom) for L1=", tL1/nm,
  "nm (NumberOfLayers=", NumberOfLayersp, ",w=", wdth/nm, "nm)"];
PlotScrollEnergyVsRIn1L1th =
  Plot[(fScrollEnergy[NumberOfLayersp, tL1[[1]], RIn1nmv nm, hp, aCCp, epsp,
        CCp, CBNp] -
       fEnergyFlatPlates[NumberOfLayersp, tL1[[1]], aCCp, epsp])/(eV/
        atom), {RIn1nmv, RIn1Minp/nm, RIn1Maxp/nm}, PlotRange -> PlotRangep/(eV/atom)];
PlotScrollEnergyVsRIn1L2th =
  Plot[(fScrollEnergy[NumberOfLayersp, tL1[[2]], RIn1nmv nm, hp, aCCp, epsp,
        CCp, CBNp] -
       fEnergyFlatPlates[NumberOfLayersp, tL1[[2]], aCCp, epsp])/(eV/
        atom), {RIn1nmv, RIn1Minp/nm, RIn1Maxp/nm}, PlotRange -> PlotRangep/(eV/atom)];
PlotScrollEnergyVsRIn1L3th =
  Plot[(fScrollEnergy[NumberOfLayersp, tL1[[3]], RIn1nmv nm, hp, aCCp, epsp,
        CCp, CBNp] -
       fEnergyFlatPlates[NumberOfLayersp, tL1[[3]], aCCp, epsp])/(eV/
        atom), {RIn1nmv, RIn1Minp/nm, RIn1Maxp/nm}, PlotRange -> PlotRangep/(eV/atom)];
PlotScrollEnergyVsRIn1L4th =
  Plot[(fScrollEnergy[NumberOfLayersp, tL1[[4]], RIn1nmv nm, hp, aCCp, epsp,
        CCp, CBNp] -
       fEnergyFlatPlates[NumberOfLayersp, tL1[[4]], aCCp, epsp])/(eV/
        atom), {RIn1nmv, RIn1Minp/nm, RIn1Maxp/nm}, PlotRange -> PlotRangep/(eV/atom)];
Print[Show[{PlotScrollEnergyVsRIn1L1th, PlotScrollEnergyVsRIn1L2th,
    PlotScrollEnergyVsRIn1L3th, PlotScrollEnergyVsRIn1L4th}]];
Clear[eV, atom, nm];

Print[" Export the plot data to the files"];
tRIn1Regular=Table[RIn1Minp+(RIn1Maxp-RIn1Minp) iiRIn/(npRIn1),{iiRIn,1,npRIn1}];
tScrollEnergy=tRIn1Regular;
tScrollEnergymEnergyFlatPlates=tRIn1Regular;
eV=1;atom=1;nm=1;
tPlotEvsRin=Table[{},{ii,1,Length[tL1]}];
AllPlotsEVsRin={};
For[iiL1=1,iiL1<=Length[tL1],iiL1++,
L1p=tL1[[iiL1]];
NanoscrollNamep=StringJoin["Nanoscroll",ToString[NumberOfLayersp],"L",ToString[L1p/nm],"n
Print[" NanoscrollName=",NanoscrollNamep];
ScrollEnergyMEnergyFlatPlatesFileName=ToFileName[NotebookDirectory[],StringJoin["EvsRin1"
Print[" ScrollEnergyMEnergyFlatPlatesFileName=",ScrollEnergyMEnergyFlatPlatesFileName];
EnergyFlatPlatesp=fEnergyFlatPlates[NumberOfLayersp, L1p, aCCp, epsp];
For[iiRIn1=1,iiRIn1<=npRIn1,iiRIn1++,
RIn1i=tRIn1Regular[[iiRIn1]];
tScrollEnergy[[iiRIn1]]=fScrollEnergy[NumberOfLayersp, L1p, RIn1i, hp, aCCp, epsp,
        CCp, CBNp];
tScrollEnergymEnergyFlatPlates[[iiRIn1]]=tScrollEnergy[[iiRIn1]]-EnergyFlatPlatesp;
];
tPlotEvsRin[[iiL1]]=ListPlot[Transpose[{tRIn1Regular,tScrollEnergymEnergyFlatPlates}], Pl
"Print[tPlotEvsRin[[iiL1]]];";
AllPlotsEVsRin=Join[{AllPlotsEVsRin,tPlotEvsRin[[iiL1]]}];
CarbonNanoscrollEnergyVsRInFileName=StringJoin[NanoscrollNamep,"dat"];
Export[ScrollEnergyMEnergyFlatPlatesFileName,Transpose[{tRIn1Regular,tScrollEnergymEnergy
];
Print[" Plot ScrollEnergy[RIn1/nm]/(eV/atom) for L1=", tL1/nm,
```

```
  "nm (NumberOfLayers=", NumberOfLayersp, ",w=", wdth/nm, "nm)"];
Print[Show[AllPlotsEVsRin]];
Clear[eV, atom, nm];
```

The data for the paper

Structure and energetics of carbon, hexagonal boron nitride,

and carbon/hexagonal boron nitride single-layer and bilayer nanoscrolls

/ A.I. Siahlo, N.A. Poklonski, A.V. Lebedev, I.V.
Lebedeva, A.M. Popov, S.A. Vyrko, A.A. Knizhnik, Yu.E. Lozovik

// Phys. Rev. Materials.— 2018.— V. 2, №
3.— P. 036001 (9 pp.). [DOI: 10.1103/PhysRevMaterials.2.036001]

I. All Input Parameters and Constants----------

I.1.The dimensions

I.2. The Input Geometry Parameters

Number of the layers in carbon nanoscroll NumberOfLayers=1

The length of a carbon nanoribbon L1=15nm

The carbon nanoribbon width w=1nm

The inner radius of the nanoscroll RIn1=2.047nm

I.3. The Input Energy Constants

eps - the interlayer interaction energy per one atom of

the nanoscroll:

eps=0.035eV/atom

C - the bending elastic constant:

CCelast=2.01eV AA^2/atom

CCBNelast=2.01eV AA^2/atom

I.4.The Input Geometry constants----------

The interatomic distance aCC and the interlayer distance h

aCC=0.142nm, h=0.3354nm

dPhi12 - The difference of the inner angles of the spirales

of the Layers

dPhi12=0.(2Pi),for the high L dPhi12HighL=$\frac{1}{2}$(2Pi)

I.5.The parameters for the visualisation

I.6. The parameters of the output file

NanoscrollName=Nanoscroll1L15nm

Nanoscroll1L15nm.txt

(The output of the data to a file Is Not Performed)

The number of the output points = 1000

I.7. The Input Numerical Constants used in the programm

The Indexes used for the work with EVdW[...] function

--------End Of The Input---------------

II. The derivated parameters and the functions required

II.1. The derivated parameters

The cell area Sa=$\frac{3\sqrt{3}\ aCC^2}{4}$=0.0261938nm^2

II.2. The required functions--------------

II.2.1. The function

fSpiraleLen[NumberOfLayers,PhiIn, PhiOut, h]

defines the Length of a Spirale with the inner agle PhiIn and
the outer angle PhiOut

II.2.2. The function fElast[PhiIn,PhiOut] is required to
calculate an elastic energy

$$\text{fElast[PhiIn,PhiOut]}=\frac{\sqrt{1+\text{PhiIn}^2}}{\text{PhiIn}}-\frac{\sqrt{1+\text{PhiOut}^2}}{\text{PhiOut}}-\text{ArcSinh[PhiIn]}+\text{ArcSinh[PhiOut]}$$

II.2.3. The function fPhiOutvsPhiInLh[NumberOfLayers,PhiIn,L,h] is a
good approximation

to obtain the value of PhiOut for the defined PhiIn,L,h

$$\text{fPhiOutvsPhiInLh[NumberOfLayers,PhiIn,L,h]}=\sqrt{\text{PhiIn}^2+\frac{4\,\text{L}\,\pi}{\text{h NumberOfLayers}}}$$

and the inverse function fPhiInvsPhiOutLh[NumberOfLayers,PhiOut, L, h]:

$$\text{fPhiInvsPhiOutLh[NumberOfLayers,PhiOut,L,h]}=\sqrt{\text{PhiOut}^2-\frac{4\,\text{L}\,\pi}{\text{h NumberOfLayers}}}$$

III. Begin of Calculation

III.1. The inner and the outer angles of the spirales

PhiIn1=RIn1 2 Pi/(NumberOfLayers h),
PhiOut1=fPhiOutvsPhiInLh[NumberOfLayers,,PhiIn1,L1,h].

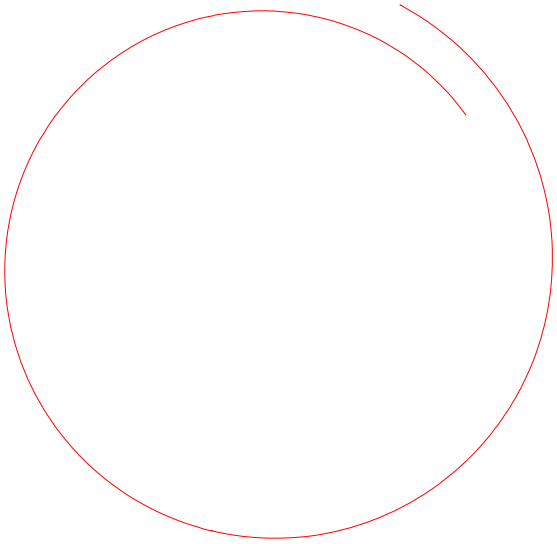
For RIn1=2.047nm,h=0.3354nm,NumberOfLayers=1:

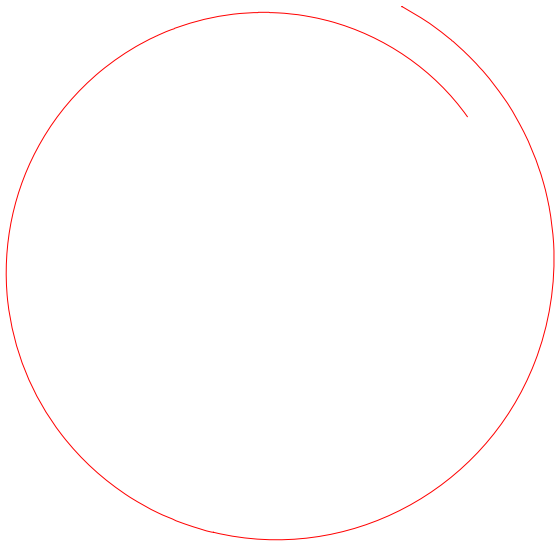
PhiIn1=6.10316(2Pi),PhiOut1=7.17525(2Pi)

for dPhi12=0: PhiIn2=6.10316(2Pi),PhiOut2=7.17525(2Pi)

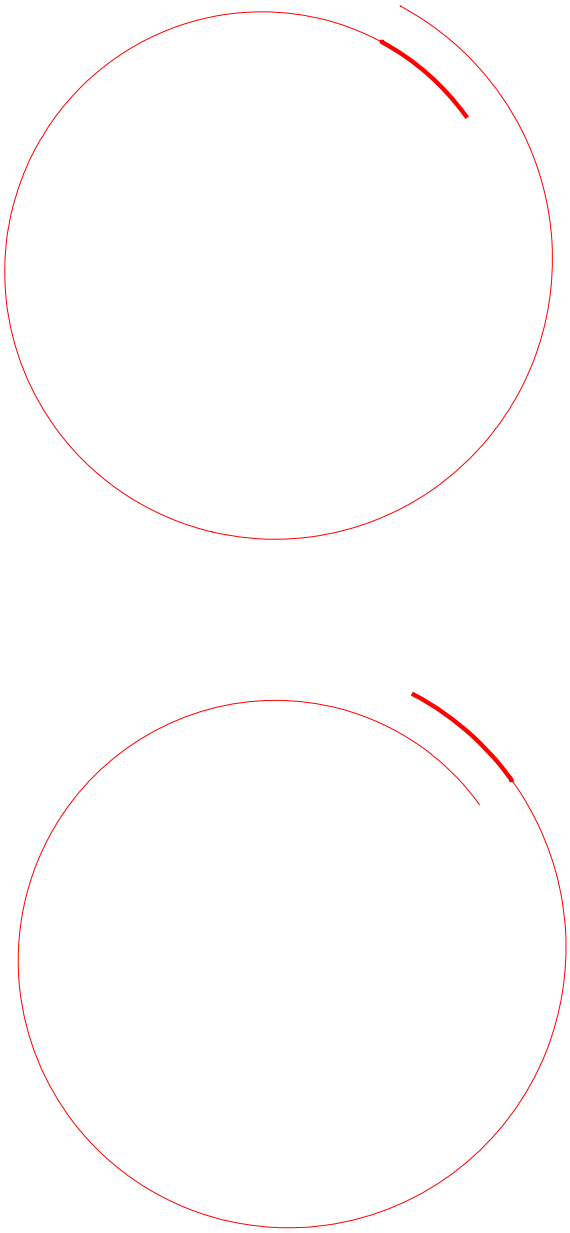
for dPhi12=Pi: PhiIn2=6.60316(2Pi),PhiOut2=7.60509(2Pi)

Plot Spirales of the layers for dPhi12=0 and dPhi12=Pi





{Spirale,
Spirale1UnderSpirale1},{Spirale1,Spirale1OverSpirale1}



III.2. The nanoscroll energy calculation

III.2.1. The elastic energy calculation

EelastC=2.32703eV/atom

EelastBN=2.32703eV/atom

III.2.2. The Van-der-Waals energy calculation

for dPhi12=0.Pi EVdWvardPhi12allp[[iEVdW]]=-1.34816eV/atom

EVdWvardPhi12allp[[iEVdW1Un1]]=-0.623346eV/atom

EVdWvardPhi12allp[[iEVdW1Ov1]]=-0.724818eV/atom

III.3. The energy of flat places

EnergyFlatPlates=-eps width/Sa L1(NumberOfLayers-1) =0eV/atom

III.4. The total energy of the nanoscroll

III.5. Determine the inner angles mismatch for the bi-layer nanoscroll
for the high nanoribbon Length

For L1=15nm,RIn=2.047nm,h=0.3354nm and dPhi12=0:

ScrollEnergy=0.978871eV/atom

For L1=15nm,RIn=2.047nm,h=0.3354nm and dPhi12=Pi:

ScrollEnergy=ScrollEnergydPhi12PipeV/atom

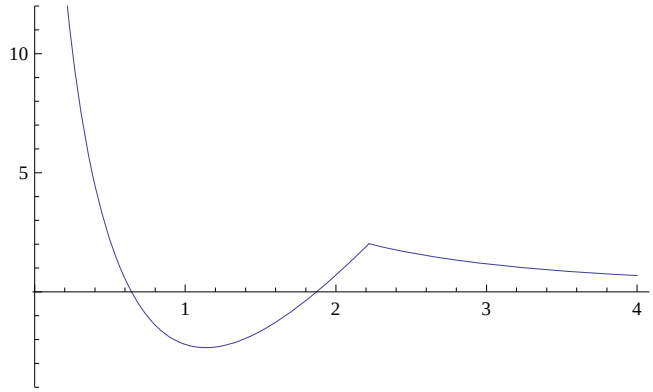
IV.The potential energy of the scroll

as a function of the inner radius RIn

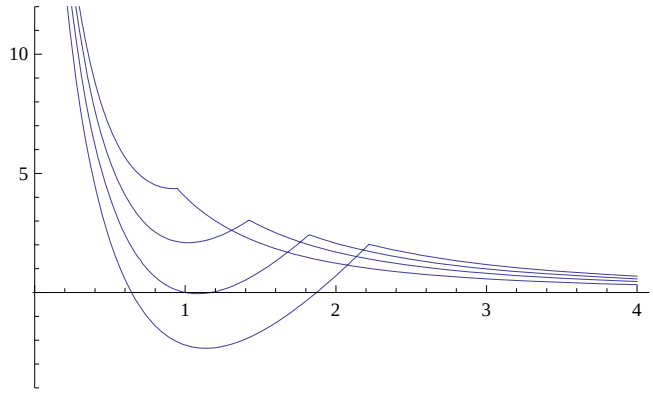
NumberOfLayers=1

eps=0.035eV/atom,C=0.0201(eV nm^2/atom),aCC=0.142nm,h=0.3354nm

Plot ScrollEnergy[RIn1/nm]/(eV/atom) for L1=15nm (NumberOfLayers=1,w=1nm)



Plot ScrollEnergy[RIn1/nm]/(eV/atom) for L1={7, 10, 12.5, 15}nm (NumberOfLayers=1,w=1nm)



Export the plot data to the files

NanoscrollName=Nanoscroll1L7nm

ScrollEnergyMEnergyFlatPlatesFileName=C:\1\EvsRin1Nanoscroll1L7nm.dat

NanoscrollName=Nanoscroll1L10nm

ScrollEnergyMEnergyFlatPlatesFileName=C:\1\EvsRin1Nanoscroll1L10nm.dat

NanoscrollName=Nanoscroll1L12.5nm

ScrollEnergyMEnergyFlatPlatesFileName=C:\1\EvsRin1Nanoscroll1L12.5nm.dat

NanoscrollName=Nanoscroll1L15nm

ScrollEnergyMEnergyFlatPlatesFileName=C:\1\EvsRin1Nanoscroll1L15nm.dat

```
Plot ScrollEnergy[RIn1/nm]/(eV/atom) for L1={7, 10, 12.5, 15}nm (NumberOfLayers=1,w=1nm)
```