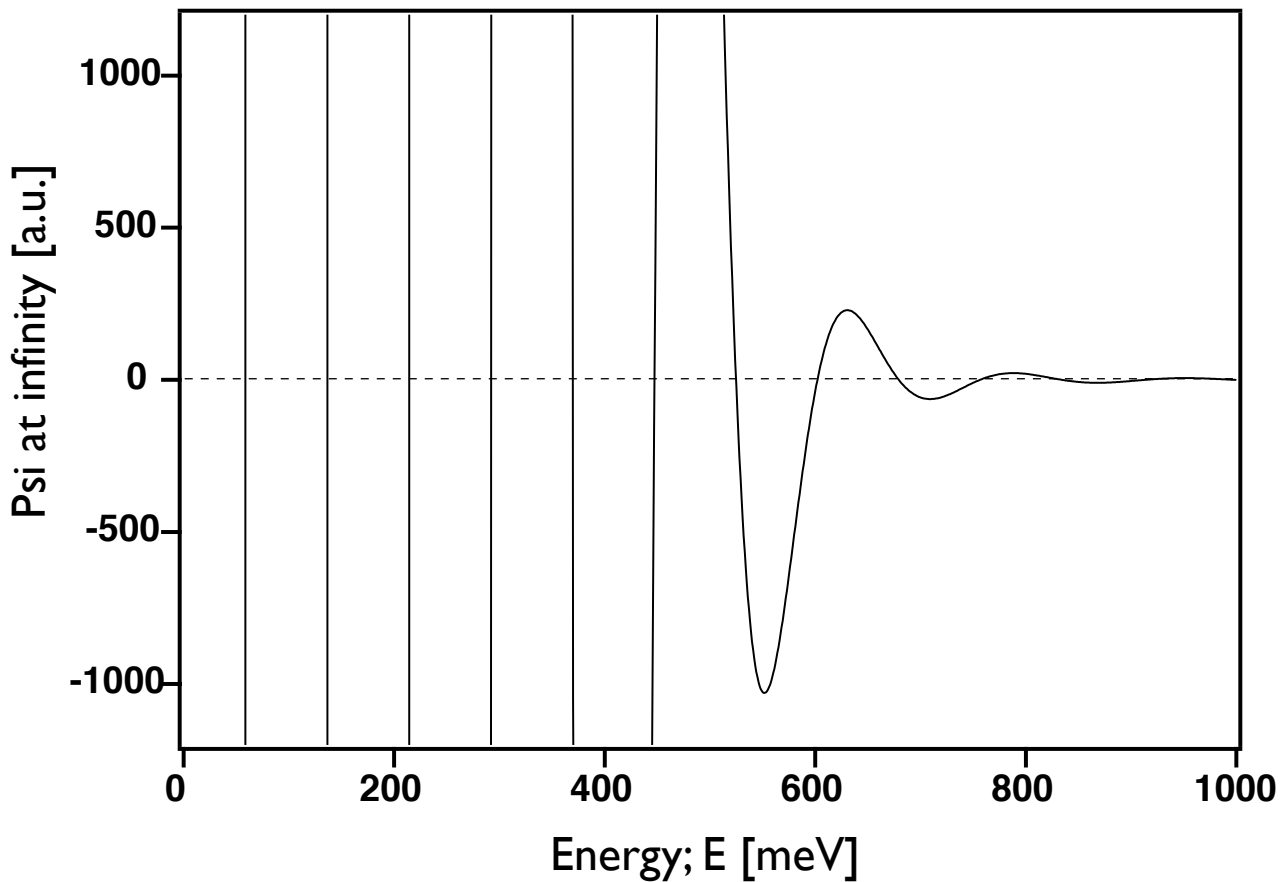


```

1 Shooting method program
2
3 #define hbar 1.05459e-34 /* プランク定数 */
4 #define m 9.109534e-31 /* 真空中の電子の質量 */
5 #define e_0 1.602189e-19 /* 電気素量 */
6 main()
7 {
8 float dE=1e-3*e_0; /* エネルギーの増分; dE*/
9 float dx=1e-10; /*座標の有限差分; dx*/
10 float E; /* the energy E */
11 float psi0,psi1,psi2; /* psi(x-dx), psi(x), and psi(x+dx) */
12 float x; /* 座標*/
13
14 for(E=0;E<e_0;E+=dE) /*エネルギーを0から変化させる*/
15 {
16 psi0=0;psi1=1; /*波動関数の初期値*/
17 for(z=dx;x<100e-10;x+=dx) /*波動関数を変化させる; 初期値x=0+dx*/
18 {
19 psi2=(2*m*(dx/hbar)*(dx/hbar)*
20 (e_0*(x/100e-10)*(x/100e-10)-E)+2)*psi1-psi0; /*Shooting法でPsi2を求める*/
21 psi0=psi1; /*座標のシフト; */
22 psi1=psi2;
23 }
24 printf("E=%fmeV psi(infty)=%f\n",E/(1e-3*e_0),psi2); /*結果の出力meV単位*/
25 }
26
27 }

```



Energy s.t. psi@infinity=0 [meV] : 58.5, 136.5, 214.5, 292.5, 369.5, 447.5, 524.5, 602.5, 679.5, 756.5
 $E_n/(n-1/2)$ [meV] : 39.0, 39.0, 39.0, 39.0, 38.9, 38.9, 38.9, 38.9, 38.8, 38.8

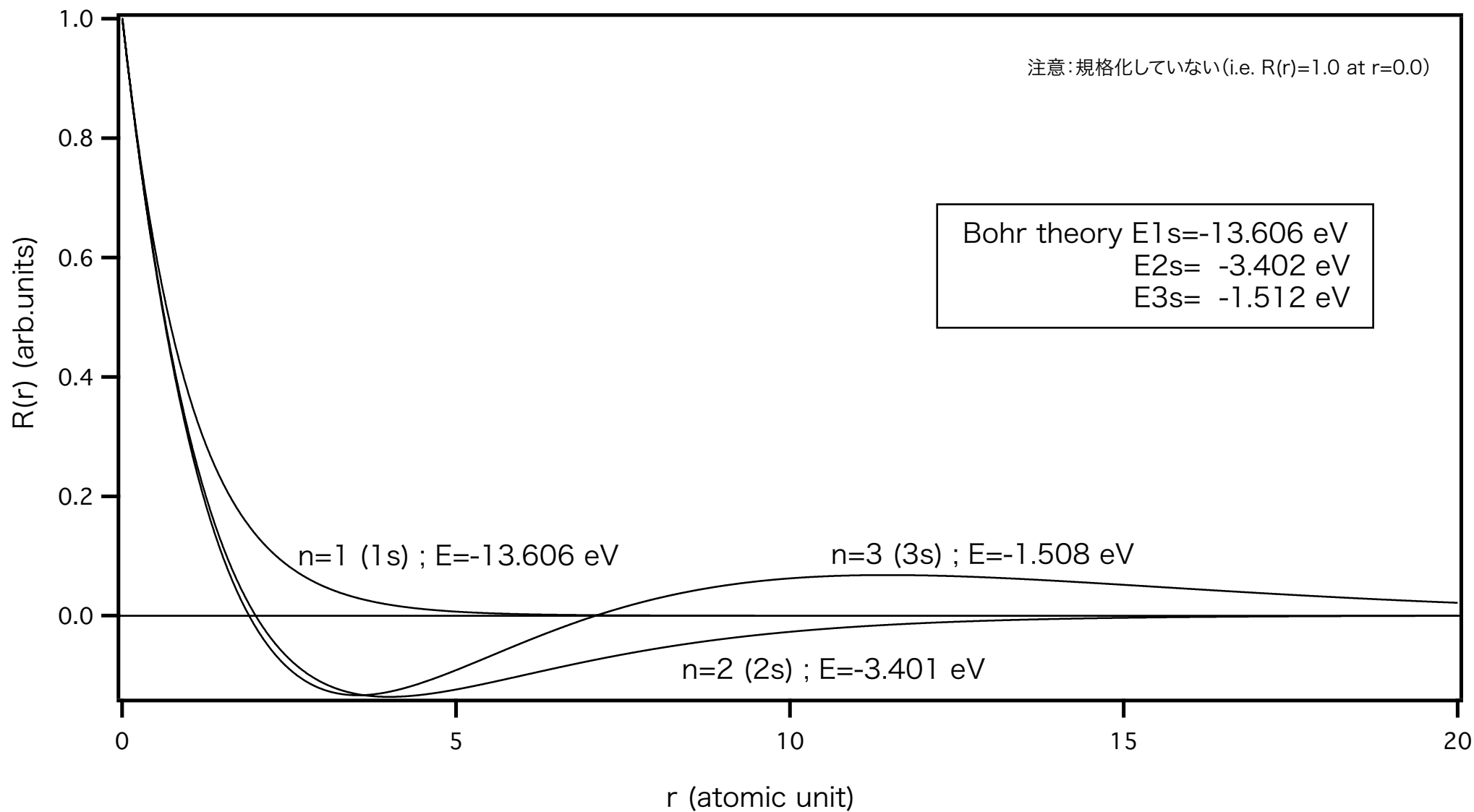
球対称ポテンシャル場中の電子状態(s状態)

```

1  #include <stdio.h>
2  #include <math.h>
3
4  main()
5  {
6  double  psi_at_inf(); /* r=∞での動径方向の波動関数の値 psi(∞,E) */
7  double  dE=5.0e-6; /* エネルギーの増分 */
8  double  dummy; /* dummy */
9  double  E; /* the energy E in Hartree */
10 double  E_eV; /* the energy E in eV */
11 double  E_soln; /* the solutions for the energy E */
12 double  Y1,Y2; /* psi(∞,E1), psi(∞,E2) */
13
14 Y1=psi_at_inf(-0.7,0); /* エネルギーの初期値 (この場合は-0.7 Hartree) での波動関数の値を評価 */
15
16 for(E=-0.7+dE;E<0.0;E+=dE) /* エネルギー探索ループの開始; E=0までスキャン */
17 {
18 Y2=psi_at_inf(E,0); /* calculate next value */
19 if(Y1*Y2<0.0) /* Y1*Y2が負になる→ psi(∞,E)がゼロ軸を横切った */
20 {
21 E_soln=fabs(Y1)*dE/(fabs(Y1)+fabs(Y2))+E-dE; /* psi(∞,E)=0となるエネルギー; Y1とY2の重率平均 */
22 E_eV=E_soln*27.2116; /* Hartree原子単位→eV単位 */
23 printf("E=%feV\n",E_eV);
24 dummy=psi_at_inf(E,1); /* print out wave function */
25 }
26 Y1=Y2;
27 }
28
29 } /* end main */
30
31 /* 各関数の定義 */
32
33 double
34 psi_at_inf(E,flag) /* r=∞での動径方向の波動関数の値 psi(∞,E)を計算するルーチン */
35
36 double  E; /* the energy E */
37 int     flag; /* 波動関数印字のフラッグ; flag=1ならprint out */
38 {
39 double  V(); /* ポテンシャル関数 */
40 double  dr=0.0050; /* shooting法におけるrの増分; δr */
41 double  psi0,psil,psi2; /* psi(z-dz), psi(z), and psi(z+dz) */
42 double  r; /* r */
43 double  m_eff; /* 有効質量 (m0を単位) */
44
45 m_eff=1.0; /* 有効質量 (この場合は真空中の電子を想定) */
46 psi0=1.0;psil=1.0; /* shooting法; はじめの2点の初期値 */
47 if(flag)printf("0.000000 %f\n",psi0); /* first point */
48 for(r=dr;r<30.0;r+=dr) /* rの始点はr=δr; r=30 a.u. (c.a.16 Å) までスキャン */
49 {
50 if(flag)printf("%e %f\n",r,psil); /* print wf */
51 psi2=(r*(2.0*m_eff*dr*dr*(V(r)-E)+2.0)*psil+(-r+dr)*psi0)/ /* shooting法のアルゴリズム */
52 (r+dr);
53 psi0=psil; /* 計算点をシフト */
54 psil=psi2;
55 }
56
57 return(psi2); /* psi(∞,E)の値をメインルーチンに返す */
58 }
59
60
61 double
62 V(r) /* ポテンシャル関数の定義 (この部分を変更すれば、任意の球対称場ポテンシャルに適用可能) */
63
64 double  r;
65 {
66 return(-1.0/r); /* 水素原子のポテンシャル (クーロン引力ポテンシャル) */
67 }

```

水素原子の束縛状態



球状量子ドットの束縛状態

GaAsの有効質量(電子) = $0.067 \times m_0$

伝導帯のバンドオフセット = 167 meV

→ 量子ドットの内部では $V = -0.167$ eV
量子ドットの外側では $V = 0.0$ eV

量子ドットの半径 = 12.0 nm = 226.8 a.u. (atomic unit)

プログラムはポテンシャル部分を書き換えれば良い

```
double
V(r)

double r;      /* spatial coordinate */
{
  if(r<226.8)  /* 226.8 a.u. = 12 nm ; radius of QD */
    return(-0.167/27.2116);
  else
    return 0.0;
}
```

$dE = 1.0 \times 10^{-6}$ Hartree

$dr = 0.05$ atomic unit

$m_{\text{eff}} = 0.067 m_0$

エネルギーの初期値 = $-0.167/27.2116$ Hartree

計算結果；束縛状態は2個

