

```

1  #include <cstdio>
2  #include <cmath>
3  #include <fstream>
4  #include <iostream>
5  using namespace std ;
6
7  double
8  D,
9  Sx,
10 tau,
11 h
12 ;
13
14 void calc_init(double rho, double p, double vx, double W);
15 void root_solver(double rho, double p, double vx, double W);
16
17 void calc_init(double rho, double p, double vx, double W){
18     h = 1.+(5./2.)*(p/rho);
19     D = rho*W;
20     Sx = rho*h*W*W*vx;
21     tau = rho*h*W*W-p-D;
22     cout << "# parameters D=" << D << " Sx=" << Sx << " tau=" << tau << " h=" << h
23     ↵ << endl;
24 }
25 ;
26
27 void root_solver(double rho, double p, double vx, double W){
28     double func, f_p, p_delta, p_old, p_new;
29     p_delta = 0.1;
30     //cout << "# parameters D=" << D << " Sx=" << Sx << " tau=" << tau << " h=" <<
31     ↵ h << endl;
32     while (abs(p_delta) > 0.000001)
33     {
34         p_old=p;
35         vx = Sx / (tau + D + p);
36         W = 1./sqrt(1. - vx*vx);
37         cout << "# parameters rho=" << rho << " p=" << p << " vx=" << vx << " W=" <<
38         ↵ W << endl;
39         func = p - 2./3./(W*W)*(tau + D + p - D*W - p*W*W);
40         f_p = 1.- 2./3.*Sx*Sx/(pow(tau + D + p,3.))*(tau + D + p - D*W);
41         p_new = p - func/f_p;
42         p_delta=p_new - p_old;
43         cout << "# p_delta=" << p_delta << " func=" << func << " f_p=" << f_p << "
44         ↵ p=" << p_new << endl;
45         p = p_new;
46     }
47     cout << "# new pressure p=" << p << endl;
48 };
49
50 int main(int argc, char** argv)
51 {
52     double //
53     rho = stod(argv[1]),
54     p = stod(argv[2]),
55     vx = stod(argv[3]),
56     W = 1./sqrt(1.-vx*vx)
57     ;
58
59     cout << "# parameters rho=" << rho << " p=" << p << " vx=" << vx << " W=" << W
60     ↵ << endl;

```

```
57
58     calc_init(rho, p, vx, W);
59
60     root_solver(rho, p, vx, W);
61     return 0;
62 };
```