1 Введение

Текст Текст Текст

Listing 1: Текст

```
#define _CRT_SECURE_NO_WARNINGS
#define _USE_MATH_DEFINES

#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <time.h>
#include <csignal>
#include <fstream>
#include <string.h>
#include <iostream>

#define M_2PI 2*M_PI
using namespace std;
```

2 Введение

Текст Текст Текст

Listing 2: Текст

```
struct indexes
{
  int i,j,k;
};
typedef indexes matrix_ind[25];
```

3 Введение

Текст Текст Текст

Listing 3: Текст

```
long double
x_L, y_L, z_L, r_L,
   x1_L, x2_L, x3_L, x4_L, x5_L,
   *Hx, *Hy, *Hz,
   *Cx, *Cy, *Cz,
   nu,
   rho,
   Rn, R0,
   Rn_1, Rn_2,
   eps,
   p_left, p_right,
   ***U,
   ***U_1, ***U_2,
   ***R,
   ***Z;

int
Nx, Ny, Nz, dNz, Rd, Rd_2, Rd_4,
  Nx1, Nx2, Nx3, Nx4, Nx5,
  Nz1, Nz2, Nz3,
  iters,
  ***G;

matrix_ind ***arg, ***func;

int ***carg;

int ***cfunc;
int ***groups;
int num;
```

4 Введение

Текст Текст Текст

Listing 4: Текст

```
long double norm(long double*** v);
long double norm(long double*** v1, long double*** v2);
void residual();
void residual(int i, int j, int k);
void alloc(int ***&a, int x, int y, int z);
void alloc(matrix_ind ***&a, int x, int y, int z);
void alloc(long double ***&a, int x, int y, int z);
void del(int ***&a, int x, int y, int z);
void del(long double ***&a, int x, int y, int z);
void del(matrix_ind ***&a, int x, int y, int z);
long double A1(long double ***U1, long double ***U2, int i,
    int j, int k);
long double A2(long double ***U, int i, int j, int k);
long double A(long double ***U1, long double ***U2, int i,
   int j, int k);
void set_gr();
void print_gr();
double random_gr();
void add_gr(matrix_ind ***&m, int &n, int i, int j, int k,
   int l, int s, int t);
void init_gr();
void eval_scalars(long double ***u, long double ***R, int
   i1, int j1, int k1, long double &Rn_F1, long double &
   Rn_F2, long double &F1_F1, long double &F2_F2, long
   double &F1_F2);
long double calc_alpha(long double Rn_F1, long double Rn_F2
   , long double F1_F1, long double F2_F2, long double
   F1_F2);
int cubic(long double *x, long double a, long double b,
   long double c);
void alpha_iter();
void print_info();
void print_texplot_matrix();
void print_vtk_header(char *output_path, int sizeX, int
   sizeY, int sizeZ);
void print_vtk_data_header(char *output_path, int sizeX,
   int sizeY, int sizeZ);
void print_area();
void print_vtk_header_points(char *output_path, int sizeX,
   int sizeY, int sizeZ);
void print_area_points();
```

5 Введение

Текст Текст Текст

Listing 5: Текст

```
long double norm(long double*** v)
{
  long double s = 0;
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        s += v[i][j][k]*v[i][j][k]*Hx[i]*Hy[j]*Hz[k];
  return sqrt(s);
}
```

6 Введение

Текст Текст Текст

Listing 6: Текст

```
long double norm(long double*** v1, long double*** v2)
{
  long double s = 0, a;
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        if(G[i][j][k])
        {
          a = v1[i][j][k]-v2[i][j][k];
          s += a*a*Hx[i]*Hy[j]*Hz[k];
        }
  return sqrt(s);
}
```

7 Введение

Текст Текст Текст

Listing 7: Текст

```
void residual()
{
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        if(G[i][j][k])
          R[i][j][k] = A(U,U,i,j,k);
}
```

8 Введение

Текст Текст Текст

Listing 8: Текст

```
void residual(int i, int j, int k)
{
  int count_changed = carg[i][j][k];
  for(int oc=0; oc<count_changed; ++oc)
  {
    indexes &changed=arg[i][j][k][oc];
    int ii=changed.i;
    int jj=changed.j;
    int kk=changed.k;
    if(G[ii][jj][kk])
      R[ii][jj][kk] = A(U,U,ii,jj,kk);
  }
}
```

9 Введение

Текст Текст Текст

Listing 9: Текст

```
void alloc(int ***&a, int x, int y, int z)
{
  a = new int **[x];
  for(int i=0; i<x; ++i)
  {
    a[i] = new int *[y];
    for(int j=0; j<Ny; ++j)
    {
      a[i][j] = new int [z];
      for(int k=0; k<Nz; ++k)
        a[i][j][k] = 0;
    }
  }
}
```

10 Введение

Текст Текст Текст

Listing 10: Текст

```
void alloc(matrix_ind ***&a, int x, int y, int z)
{
  a = new matrix_ind **[x];
  for(int i=0; i<x; ++i)
  {
    a[i] = new matrix_ind *[y];
    for(int j=0; j<Ny; ++j)
      a[i][j] = new matrix_ind [z];
  }
}
```

11 Введение

Текст Текст Текст

Listing 11: Текст

```
void alloc(long double ***&a, int x, int y, int z)
{
  a = new long double **[x];
  for(int i=0; i<x; ++i)
  {
    a[i] = new long double *[y];
    for(int j=0; j<Ny; ++j)
    {
      a[i][j] = new long double [z];
      for(int k=0; k<Nz; ++k)
        a[i][j][k] = 0;
    }
  }
}
```

12 Введение

Текст Текст Текст

Listing 12: Текст

```
void del(int ***&a, int x, int y, int z)
{
  for(int i=0; i<x; ++i)
  {
    for(int j=0; j<y; ++j)
      delete [] a[i][j];
    delete [] a[i];
  }
  delete [] a;
}
```

13 Введение

Текст Текст Текст

Listing 13: Текст

```
void del(long double ***&a, int x, int y, int z)
{
  for(int i=0; i<x; ++i)
  {
    for(int j=0; j<y; ++j)
      delete [] a[i][j];
    delete [] a[i];
  }
  delete [] a;
}
```

14 Введение

Текст Текст Текст

Listing 14: Текст

```
void del(matrix_ind ***&a, int x, int y, int z)
{
  for(int i=0; i<x; ++i)
  {
    for(int j=0; j<y; ++j)
      delete [] a[i][j];
    delete [] a[i];
  }
  delete [] a;
}
```

15 Введение

Текст Текст Текст

Listing 15: Текст

```
long double A1(long double ***U1, long double ***U2, int i,
    int j, int k)
{
  int g = G[i][j][k];
  if( g == 0 || k>=dNz*3 ) return 0;

  int uk = k%dNz;
  int vk = uk + dNz;
  int wk = vk + dNz;

  long double ddx, ddy, ddz;
  if(k==uk)
  {
```

```
  long double appv, appw;
  switch(g)
  {
  case 2:
    ddx = (U2[i+1][j][uk]-U2[i][j][uk])*2/(Hx[i+1]+Hx[i])
       *U1[i][j][uk];
    break;
  case 3:
    ddx = (U2[i][j][uk]-U2[i-1][j][uk])*2/(Hx[i]+Hx[i-1])
       *U1[i][j][uk];
    break;
  default:
    ddx = (U2[i+1][j][uk]-U2[i-1][j][uk])*2/(Hx[i+1]+2*Hx
       [i]+Hx[i-1])*U1[i][j][uk];
  }

  appv = 0.5*( Hy[j]/(Hy[j]+Hy[j+1])*(U1[i][j+1][vk]+U1[i
     -1][j+1][vk]) +
         Hy[j+1]/(Hy[j]+Hy[j+1])*(U1[i][j][vk]+U1[i-1][j
            ][vk]) );
  ddy = (U2[i][j+1][uk]-U2[i][j-1][uk])/(Hy[j+1]+Hy[j])*
     appv;

  appw = 0.5*( Hz[k]/(Hz[k]+Hz[k+1])*(U1[i][j][wk+1]+U1[i
     -1][j][wk+1]) +
         Hz[k+1]/(Hz[k]+Hz[k+1])*(U1[i][j][wk]+U1[i-1][j
            ][wk]) );
  ddz = (U2[i][j][uk+1]-U2[i][j][uk-1])/(Hz[k+1]+Hz[k])*
     appw;
}
else if(k==vk)
{
  long double appu, appw;
  switch(g)
  {
  case 2:
    appu = 0.5*( U1[i+1][j][uk] + U1[i+1][j-1][uk] );
    ddx = (U2[i+1][j][vk]-U2[i][j][vk])/Hx[i+1]*appu;
    break;
  case 3:
    appu = 0.5*( U1[i][j][uk] + U1[i][j-1][uk] );
    ddx = (U2[i][j][vk]-U2[i-1][j][vk])/Hx[i]*appu;
    break;
  default:
    appu = 0.5*( Hx[i+1]/(Hx[i]+Hx[i+1])*(U1[i][j][uk]+U1
       [i][j-1][uk]) +
           Hx[i]/(Hx[i]+Hx[i+1])*(U1[i+1][j][uk]+U1[i+1][
              j-1][uk]) );
    ddx = (U2[i+1][j][vk]-U2[i-1][j][vk])/(Hx[i+1]+Hx[i])
       *appu;
  }

  ddy = (U2[i][j+1][vk]-U2[i][j-1][vk])*2/(Hy[j+1]+2*Hy[j
     ]+Hy[j-1])*U1[i][j][vk];

  appw = 0.5*( Hz[k]/(Hz[k]+Hz[k+1])*(U1[i][j][wk+1]+U1[i
     ][j-1][wk+1]) +
         Hz[k+1]/(Hz[k]+Hz[k+1])*(U1[i][j][wk]+U1[i][j
            -1][wk]) );
  ddz = (U2[i][j][vk+1]-U2[i][j][vk-1])/(Hz[k+1]+Hz[k])*
     appw;
}
else if(k==wk)
{
  long double appu, appv;
  switch(g)
  {
  case 2:
    appu = 0.5*( U1[i+1][j][uk] + U1[i+1][j][uk-1] );
    ddx = (U2[i+1][j][wk]-U2[i][j][wk])/Hx[i+1]*appu;
    break;
  case 3:
    appu = 0.5*( U1[i][j][uk] + U1[i][j][uk-1] );
    ddx = (U2[i][j][wk]-U2[i-1][j][wk])/Hx[i]*appu;
    break;
  default:
    appu = 0.5*( Hx[i+1]/(Hx[i]+Hx[i+1])*(U1[i][j][uk]+U1
       [i][j][uk-1]) +
           Hx[i]/(Hx[i]+Hx[i+1])*(U1[i+1][j][uk]+U1[i+1][
              j][uk-1]) );
    ddx = (U2[i+1][j][wk]-U2[i-1][j][wk])/(Hx[i+1]+Hx[i])
       *appu;
  }

  appv = 0.5*( Hy[j]/(Hy[j]+Hy[j+1])*(U1[i][j+1][vk]+U1[i
     ][j+1][vk-1]) +
         Hy[j+1]/(Hy[j]+Hy[j+1])*(U1[i][j][vk]+U1[i][j][
            vk-1]) );
    ddy = (U2[i][j+1][wk]-U2[i][j-1][wk])/(Hy[j+1]+Hy[j])*
       appv;

    ddz = (U2[i][j][wk+1]-U2[i][j][wk-1])*2/(Hz[k+1]+2*Hz[k
       ]+Hz[k-1])*U1[i][j][wk];
  }

  return (ddx+ddy+ddz);
}
```

16 Введение

Текст Текст Текст

Listing 16: Текст

```
long double A2(long double ***U, int i, int j, int k)
{
  int g=G[i][j][k];
  if(!g) return 0;

  int uk = k%dNz;
  int vk = uk + dNz;
  int wk = vk + dNz;
  int pk = wk + dNz;

  if(k>=3*dNz)
  {
    switch(g)
    {
    case 5:
      return (U[i][j][pk+1]-U[i][j][pk])/Hz[k+1];
    case 6:
      return (U[i][j][pk]-U[i][j][pk-1])/Hz[k];
    case 7:
      return (U[i][j+1][pk]-U[i][j][pk])/Hy[j+1];
    case 8:
      return (U[i][j][pk]-U[i][j-1][pk])/Hy[j];
    default:
      return (U[i+1][j][uk]-U[i][j][uk])*2/(Hx[i+1]+Hx[i])
         +
           (U[i][j+1][vk]-U[i][j][vk])*2/(Hy[j+1]+Hy[j]) +
           (U[i][j][wk+1]-U[i][j][wk])*2/(Hz[k+1]+Hz[k]);
    }
  }

  long double lap_u;
  int i1=i;
  switch(g)
  {
  case 2:
    i1+=1;
    break;
  case 3:
    i1-=1;
  }
  if(k==uk)
  {
    lap_u=nu*(
          ((U[i1+1][j][k]-U[i1][j][k])*2/(Hx[i1+1]+Hx[i1])-
           (U[i1][j][k]-U[i1-1][j][k])*2/(Hx[i1]+Hx[i1-1]))
              *2/(0.5*Hx[i1+1]+Hx[i1]+0.5*Hx[i1-1]) +
          ((U[i][j+1][k]-U[i][j][k])/Hy[j+1]-
           (U[i][j][k]-U[i][j-1][k])/Hy[j])/(Hy[j+1]+Hy[j])
              *2.0 +
          ((U[i][j][k+1]-U[i][j][k])/Hz[k+1]-
           (U[i][j][k]-U[i][j][k-1])/Hz[k])/(Hz[k+1]+Hz[k])
              *2.0);
  }
  else if(k==vk)
  {
    lap_u=nu*(
          ((U[i1+1][j][k]-U[i1][j][k])/Hx[i1+1]-
           (U[i1][j][k]-U[i1-1][j][k])/Hx[i1])/(Hx[i1+1]+Hx
              [i1])*2.0+
          ((U[i][j+1][k]-U[i][j][k])*2/(Hy[j+1]+Hy[j])-
           (U[i][j][k]-U[i][j-1][k])*2/(Hy[j]+Hy[j-1]))
              *2/(0.5*Hy[j+1]+Hy[j]+0.5*Hy[j-1])+
          ((U[i][j][k+1]-U[i][j][k])/Hz[k+1]-
           (U[i][j][k]-U[i][j][k-1])/Hz[k])/(Hz[k+1]+Hz[k])
              *2.0);
  }
  else if(k==wk)
  {
    lap_u=nu*(
          ((U[i1+1][j][k]-U[i1][j][k])/Hx[i1+1]-
          (U[i1][j][k]-U[i1-1][j][k])/Hx[i1])/(Hx[i1+1]+Hx
             [i1])*2.0+
         ((U[i][j+1][k]-U[i][j][k])/Hy[j+1]-
          (U[i][j][k]-U[i][j-1][k])/Hy[j])/(Hy[j+1]+Hy[j])
             *2.0 +
         ((U[i][j][k+1]-U[i][j][k])*2/(Hz[k+1]+Hz[k])-
          (U[i][j][k]-U[i][j][k-1])*2/(Hz[k]+Hz[k-1]))
             *2/(0.5*Hz[k+1]+Hz[k]+0.5*Hz[k-1]));
  }

  long double grad_p = 0;
  if(k==uk)
    grad_p=(U[i][j][pk]-U[i-1][j][pk])/Hx[i];
  else if(k==vk)
    grad_p=(U[i][j][pk]-U[i][j-1][pk])/Hy[j];
  else if(k==wk)
    grad_p=(U[i][j][pk]-U[i][j][pk-1])/Hz[k];

  return grad_p/rho - lap_u;
}
```

17 Введение

Текст Текст Текст

Listing 17: Текст

```
long double A(long double ***U1, long double ***U2, int i,
   int j, int k)
{
  return A1(U1,U2,i,j,k)+A2(U2,i,j,k);
}
```

18 Введение

Текст Текст Текст

Listing 18: Текст

```
void set_gr()
{
  int ***field;
  alloc(field,Nx,Ny,Nz);
  bool cells;
  do
  {
    for(int i=0; i<Nx; ++i)
      for(int j=0; j<Ny; ++j)
        for(int k=0; k<Nz; ++k)
          field[i][j][k] = 0;

    cells = false;
    for(int i=0; i<Nx; ++i)
      for(int j=0; j<Ny; ++j)
        for(int k=0; k<Nz; ++k)
          if(groups[i][j][k]<0)
          {
            if(!cells)
            {
              cells = true;
              ++num;
            }
            int count = carg[i][j][k];
            indexes *t = arg[i][j][k];
            bool push = true;
            for(int c=0; (c<count)&&push; ++c)
            {
              if( field[t[c].i][t[c].j][t[c].k] ) push =
                 false;
            }
            if(push)
            {
              groups[i][j][k] = num;
              for(int c=0; c<count; ++c)
                field[t[c].i][t[c].j][t[c].k] = 1;
            }
          };
  }
  while(cells);
  ++num;
  del(field,Nx,Ny,Nz);
}
```

19 Введение

Текст Текст Текст

Listing 19: Текст

```
void print_gr()
{
  FILE *f = fopen("oper_arg.txt","w");
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
      {
        fprintf(f, "%3d, %3d, %3d: ", i, j, k);
        for(int c = 0; c<carg[i][j][k]; ++c)
          fprintf(f, "| %4d %4d %4d |", arg[i][j][k][c].i,
             arg[i][j][k][c].j, arg[i][j][k][c].k);
        fprintf(f,"\n");
      };
  fclose(f);
  f = fopen("oper_func.txt","w");
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
      {
        fprintf(f, "%3d, %3d, %3d: ", i, j, k);
        for(int c = 0; c<cfunc[i][j][k]; ++c)
          fprintf(f, "| %4d %4d %4d |", func[i][j][k][c].i,
              func[i][j][k][c].j, func[i][j][k][c].k);
        fprintf(f,"\n");
      };
  fclose(f);
}
```

20 Введение

Текст Текст Текст

Listing 20: Текст

```
double random_gr()
{
  int full = Nx*Ny*Nz;
  int number = rand();
  double z = (double)((unsigned int) ((double)number /
                     (double) RAND_MAX * (double)full) + 1)
                        /((double)full);
  return z;
}
```

21 Введение

Текст Текст Текст

Listing 21: Текст

```
  void add_gr(matrix_ind ***&m, int &n, int i, int j, int k,
     int l, int s, int t)
  {
    m[i][j][k][n].i = l;
    m[i][j][k][n].j = s;
    m[i][j][k][n].k = t;
    ++n;
  }
```

22 Введение

Текст Текст Текст

Listing 22: Текст

```
  void init_gr()
  {
    num = -1;
    srand( (unsigned)time( NULL ) );

    long double ***e, ***y;
    alloc(e,Nx,Ny,Nz);
    alloc(y,Nx,Ny,Nz);

    for(int i=0; i<Nx; ++i)
      for(int j=0; j<Ny; ++j)
        for(int k=0; k<Nz; ++k)
        {
          e[i][j][k] = 0;
          carg[i][j][k] = 0;
          cfunc[i][j][k] = 0;
          arg[i][j][k][0].i = 0;
          arg[i][j][k][0].j = 0;
          arg[i][j][k][0].k = 0;
          func[i][j][k][0].i = 0;
          func[i][j][k][0].j = 0;
          func[i][j][k][0].k = 0;
          groups[i][j][k] = -1;
        };

    int repeat = 1;
    for(int il=0; il<repeat; ++il)
    {
      printf("%4d\n",il);
      for(int a=0; a<Nx; ++a)
        for(int b=0; b<Ny; ++b)
          for(int c=0; c<Nz; ++c)
            y[a][b][c] = random_gr();

      for(int i=0; i<Nx; ++i)
      {
        printf("%8d\n",i);
        for(int j=0; j<Ny; ++j)
        {
          printf("%5d %5d %5d\n",il, i, j);
          for(int k=0; k<Nz; ++k)
          {
            e[i][j][k] = 1;
            for(int l=0; l<Nx; ++l)
            {
              for(int s=0; s<Ny; ++s)
              {
                for(int t=0; t<Nz; ++t)
                {
                  if ((fabs(A1(e,y,l,s,t)) > 1e-20)||
                       (fabs(A1(y,e,l,s,t)) > 1e-20)||
                       (fabs(A2(e,l,s,t)) > 1e-20))
                  {
                    bool d = true;
                    for(int c=0; (c<carg[i][j][k])&&d; ++c)
```

```
                    if((arg[i][j][k][c].i==l)&&
                        (arg[i][j][k][c].j==s)&&
                        (arg[i][j][k][c].k==t)) d=false;
                  if(d)
                  {
                    add_gr(arg,carg[i][j][k],i,j,k,l,s,t);
                    add_gr(func,cfunc[l][s][t],l,s,t,i,j,k)
                       ;
                  }
                }
              }
            }
          }
          e[i][j][k] = 0;
        }
      }
    }
  }
  del(e,Nx,Ny,Nz);
  del(y,Nx,Ny,Nz);
}
```

23 Введение

Текст Текст Текст

Listing 23: Текст

```
void my_init_gr()
{

  matrix_ind napr[4][5][5];
  int cnt[5][5];

  cnt[1][1]=7;
  cnt[1][2]=4;
  cnt[1][3]=4;
  cnt[1][4]=2;

  cnt[2][1]=4;
  cnt[2][2]=7;
  cnt[2][3]=4;
```

```
cnt[2][4]=2;

cnt[3][1]=4;
cnt[3][2]=4;
cnt[3][3]=7;
cnt[3][4]=2;

cnt[4][1]=2;
cnt[4][2]=2;
cnt[4][3]=2;
cnt[4][4]=0;

napr[1][1][1][1].i=0;
napr[1][1][1][1].j=0;
napr[1][1][1][1].k=0;
napr[1][1][1][2].i=1;
napr[1][1][1][2].j=0;
napr[1][1][1][2].k=0;
napr[1][1][1][3].i=-1;
napr[1][1][1][3].j=0;
napr[1][1][1][3].k=0;
napr[1][1][1][4].i=0;
napr[1][1][1][4].j=1;
napr[1][1][1][4].k=0;
napr[1][1][1][5].i=0;
napr[1][1][1][5].j=-1;
napr[1][1][1][5].k=0;
napr[1][1][1][6].i=0;
napr[1][1][1][6].j=0;
napr[1][1][1][6].k=1;
napr[1][1][1][7].i=0;
napr[1][1][1][7].j=0;
napr[1][1][1][7].k=-1;
napr[1][1][2][1].i=0;
napr[1][1][2][1].j=0;
napr[1][1][2][1].k=0;
napr[1][1][2][2].i=-1;
napr[1][1][2][2].j=0;
napr[1][1][2][2].k=0;
napr[1][1][2][3].i=0;
napr[1][1][2][3].j=1;
napr[1][1][2][3].k=0;
napr[1][1][2][4].i=-1;
napr[1][1][2][4].j=1;
napr[1][1][2][4].k=0;
```

```
napr[1][1][3][1].i=0;
napr[1][1][3][1].j=0;
napr[1][1][3][1].k=0;
napr[1][1][3][2].i=-1;
napr[1][1][3][2].j=0;
napr[1][1][3][2].k=0;
napr[1][1][3][3].i=0;
napr[1][1][3][3].j=0;
napr[1][1][3][3].k=1;
napr[1][1][3][4].i=-1;
napr[1][1][3][4].j=0;
napr[1][1][3][4].k=1;
napr[1][1][4][1].i=0;
napr[1][1][4][1].j=0;
napr[1][1][4][1].k=0;
napr[1][1][4][2].i=-1;
napr[1][1][4][2].j=0;
napr[1][1][4][2].k=0;

napr[2][1][1][1].i=0;
napr[2][1][1][1].j=0;
napr[2][1][1][1].k=0;
napr[2][1][1][2].i=1;
napr[2][1][1][2].j=0;
napr[2][1][1][2].k=0;
napr[2][1][1][3].i=2;
napr[2][1][1][3].j=0;
napr[2][1][1][3].k=0;
napr[2][1][1][4].i=0;
napr[2][1][1][4].j=1;
napr[2][1][1][4].k=0;
napr[2][1][1][5].i=0;
napr[2][1][1][5].j=-1;
napr[2][1][1][5].k=0;
napr[2][1][1][6].i=0;
napr[2][1][1][6].j=0;
napr[2][1][1][6].k=1;
napr[2][1][1][7].i=0;
napr[2][1][1][7].j=0;
napr[2][1][1][7].k=-1;
napr[2][1][2][1].i=0;
napr[2][1][2][1].j=0;
napr[2][1][2][1].k=0;
napr[2][1][2][2].i=-1;
napr[2][1][2][2].j=0;
```

```
napr[2][1][2][2].k=0;
napr[2][1][2][3].i=0;
napr[2][1][2][3].j=1;
napr[2][1][2][3].k=0;
napr[2][1][2][4].i=-1;
napr[2][1][2][4].j=1;
napr[2][1][2][4].k=0;
napr[2][1][3][1].i=0;
napr[2][1][3][1].j=0;
napr[2][1][3][1].k=0;
napr[2][1][3][2].i=-1;
napr[2][1][3][2].j=0;
napr[2][1][3][2].k=0;
napr[2][1][3][3].i=0;
napr[2][1][3][3].j=0;
napr[2][1][3][3].k=1;
napr[2][1][3][4].i=-1;
napr[2][1][3][4].j=0;
napr[2][1][3][4].k=1;
napr[2][1][4][1].i=0;
napr[2][1][4][1].j=0;
napr[2][1][4][1].k=0;
napr[2][1][4][2].i=-1;
napr[2][1][4][2].j=0;
napr[2][1][4][2].k=0;

napr[3][1][1][1].i=0;
napr[3][1][1][1].j=0;
napr[3][1][1][1].k=0;
napr[3][1][1][2].i=-2;
napr[3][1][1][2].j=0;
napr[3][1][1][2].k=0;
napr[3][1][1][3].i=-1;
napr[3][1][1][3].j=0;
napr[3][1][1][3].k=0;
napr[3][1][1][4].i=0;
napr[3][1][1][4].j=1;
napr[3][1][1][4].k=0;
napr[3][1][1][5].i=0;
napr[3][1][1][5].j=-1;
napr[3][1][1][5].k=0;
napr[3][1][1][6].i=0;
napr[3][1][1][6].j=0;
napr[3][1][1][6].k=1;
napr[3][1][1][7].i=0;
```

```
napr[3][1][1][7].j=0;
napr[3][1][1][7].k=-1;
napr[3][1][2][1].i=0;
napr[3][1][2][1].j=0;
napr[3][1][2][1].k=0;
napr[3][1][2][2].i=-1;
napr[3][1][2][2].j=0;
napr[3][1][2][2].k=0;
napr[3][1][2][3].i=0;
napr[3][1][2][3].j=1;
napr[3][1][2][3].k=0;
napr[3][1][2][4].i=-1;
napr[3][1][2][4].j=1;
napr[3][1][2][4].k=0;
napr[3][1][3][1].i=0;
napr[3][1][3][1].j=0;
napr[3][1][3][1].k=0;
napr[3][1][3][2].i=-1;
napr[3][1][3][2].j=0;
napr[3][1][3][2].k=0;
napr[3][1][3][3].i=0;
napr[3][1][3][3].j=0;
napr[3][1][3][3].k=1;
napr[3][1][3][4].i=-1;
napr[3][1][3][4].j=0;
napr[3][1][3][4].k=1;
napr[3][1][4][1].i=0;
napr[3][1][4][1].j=0;
napr[3][1][4][1].k=0;
napr[3][1][4][2].i=-1;
napr[3][1][4][2].j=0;
napr[3][1][4][2].k=0;

napr[1][2][1][1].i=0;
napr[1][2][1][1].j=0;
napr[1][2][1][1].k=0;
napr[1][2][1][2].i=0;
napr[1][2][1][2].j=-1;
napr[1][2][1][2].k=0;
napr[1][2][1][3].i=1;
napr[1][2][1][3].j=0;
napr[1][2][1][3].k=0;
napr[1][2][1][4].i=1;
napr[1][2][1][4].j=-1;
napr[1][2][1][4].k=0;
```

```
napr[1][2][2][1].i=0;
napr[1][2][2][1].j=0;
napr[1][2][2][1].k=0;
napr[1][2][2][2].i=0;
napr[1][2][2][2].j=-1;
napr[1][2][2][2].k=0;
napr[1][2][2][3].i=0;
napr[1][2][2][3].j=1;
napr[1][2][2][3].k=0;
napr[1][2][2][4].i=1;
napr[1][2][2][4].j=0;
napr[1][2][2][4].k=0;
napr[1][2][2][5].i=-1;
napr[1][2][2][5].j=0;
napr[1][2][2][5].k=0;
napr[1][2][2][6].i=0;
napr[1][2][2][6].j=0;
napr[1][2][2][6].k=1;
napr[1][2][2][7].i=0;
napr[1][2][2][7].j=0;
napr[1][2][2][7].k=-1;
napr[1][2][3][1].i=0;
napr[1][2][3][1].j=0;
napr[1][2][3][1].k=0;
napr[1][2][3][2].i=0;
napr[1][2][3][2].j=-1;
napr[1][2][3][2].k=0;
napr[1][2][3][3].i=0;
napr[1][2][3][3].j=0;
napr[1][2][3][3].k=1;
napr[1][2][3][4].i=0;
napr[1][2][3][4].j=-1;
napr[1][2][3][4].k=1;
napr[1][2][4][1].i=0;
napr[1][2][4][1].j=0;
napr[1][2][4][1].k=0;
napr[1][2][4][2].i=0;
napr[1][2][4][2].j=-1;
napr[1][2][4][2].k=0;

napr[2][2][1][1].i=1;
napr[2][2][1][1].j=0;
napr[2][2][1][1].k=0;
napr[2][2][1][2].i=1;
napr[2][2][1][2].j=0;
```

```
napr[2][2][1][2].k=0;
napr[2][2][1][3].i=1;
napr[2][2][1][3].j=0;
napr[2][2][1][3].k=0;
napr[2][2][1][4].i=1;
napr[2][2][1][4].j=-1;
napr[2][2][1][4].k=0;
napr[2][2][2][1].i=0;
napr[2][2][2][1].j=0;
napr[2][2][2][1].k=0;
napr[2][2][2][2].i=0;
napr[2][2][2][2].j=-1;
napr[2][2][2][2].k=0;
napr[2][2][2][3].i=0;
napr[2][2][2][3].j=1;
napr[2][2][2][3].k=0;
napr[2][2][2][4].i=1;
napr[2][2][2][4].j=0;
napr[2][2][2][4].k=0;
napr[2][2][2][5].i=2;
napr[2][2][2][5].j=0;
napr[2][2][2][5].k=0;
napr[2][2][2][6].i=0;
napr[2][2][2][6].j=0;
napr[2][2][2][6].k=1;
napr[2][2][2][7].i=0;
napr[2][2][2][7].j=0;
napr[2][2][2][7].k=-1;
napr[2][2][3][1].i=0;
napr[2][2][3][1].j=0;
napr[2][2][3][1].k=0;
napr[2][2][3][2].i=0;
napr[2][2][3][2].j=-1;
napr[2][2][3][2].k=0;
napr[2][2][3][3].i=0;
napr[2][2][3][3].j=0;
napr[2][2][3][3].k=1;
napr[2][2][3][4].i=0;
napr[2][2][3][4].j=-1;
napr[2][2][3][4].k=1;
napr[2][2][4][1].i=0;
napr[2][2][4][1].j=0;
napr[2][2][4][1].k=0;
napr[2][2][4][2].i=0;
napr[2][2][4][2].j=-1;
```

```
napr[2][2][4][2].k=0;

napr[3][2][1][1].i=0;
napr[3][2][1][1].j=0;
napr[3][2][1][1].k=0;
napr[3][2][1][2].i=0;
napr[3][2][1][2].j=-1;
napr[3][2][1][2].k=0;
napr[3][2][1][3].i=0;
napr[3][2][1][3].j=0;
napr[3][2][1][3].k=0;
napr[3][2][1][4].i=0;
napr[3][2][1][4].j=0;
napr[3][2][1][4].k=0;
napr[3][2][2][1].i=0;
napr[3][2][2][1].j=0;
napr[3][2][2][1].k=0;
napr[3][2][2][2].i=0;
napr[3][2][2][2].j=-1;
napr[3][2][2][2].k=0;
napr[3][2][2][3].i=0;
napr[3][2][2][3].j=1;
napr[3][2][2][3].k=0;
napr[3][2][2][4].i=-2;
napr[3][2][2][4].j=0;
napr[3][2][2][4].k=0;
napr[3][2][2][5].i=-1;
napr[3][2][2][5].j=0;
napr[3][2][2][5].k=0;
napr[3][2][2][6].i=0;
napr[3][2][2][6].j=0;
napr[3][2][2][6].k=1;
napr[3][2][2][7].i=0;
napr[3][2][2][7].j=0;
napr[3][2][2][7].k=-1;
napr[3][2][3][1].i=0;
napr[3][2][3][1].j=0;
napr[3][2][3][1].k=0;
napr[3][2][3][2].i=0;
napr[3][2][3][2].j=-1;
napr[3][2][3][2].k=0;
napr[3][2][3][3].i=0;
napr[3][2][3][3].j=0;
napr[3][2][3][3].k=1;
napr[3][2][3][4].i=0;
```

```
napr[3][2][3][4].j=-1;
napr[3][2][3][4].k=1;
napr[3][2][4][1].i=0;
napr[3][2][4][1].j=0;
napr[3][2][4][1].k=0;
napr[3][2][4][2].i=0;
napr[3][2][4][2].j=-1;
napr[3][2][4][2].k=0;

napr[1][3][1][1].i=0;
napr[1][3][1][1].j=0;
napr[1][3][1][1].k=0;
napr[1][3][1][2].i=0;
napr[1][3][1][2].j=0;
napr[1][3][1][2].k=-1;
napr[1][3][1][3].i=1;
napr[1][3][1][3].j=0;
napr[1][3][1][3].k=0;
napr[1][3][1][4].i=1;
napr[1][3][1][4].j=0;
napr[1][3][1][4].k=-1;
napr[1][3][2][1].i=0;
napr[1][3][2][1].j=0;
napr[1][3][2][1].k=0;
napr[1][3][2][2].i=0;
napr[1][3][2][2].j=0;
napr[1][3][2][2].k=-1;
napr[1][3][2][3].i=0;
napr[1][3][2][3].j=1;
napr[1][3][2][3].k=0;
napr[1][3][2][4].i=0;
napr[1][3][2][4].j=1;
napr[1][3][2][4].k=-1;
napr[1][3][3][1].i=0;
napr[1][3][3][1].j=0;
napr[1][3][3][1].k=0;
napr[1][3][3][2].i=1;
napr[1][3][3][2].j=0;
napr[1][3][3][2].k=0;
napr[1][3][3][3].i=-1;
napr[1][3][3][3].j=0;
napr[1][3][3][3].k=0;
napr[1][3][3][4].i=0;
napr[1][3][3][4].j=1;
napr[1][3][3][4].k=0;
```

```
napr[1][3][3][5].i=0;
napr[1][3][3][5].j=-1;
napr[1][3][3][5].k=0;
napr[1][3][3][6].i=0;
napr[1][3][3][6].j=0;
napr[1][3][3][6].k=1;
napr[1][3][3][7].i=0;
napr[1][3][3][7].j=0;
napr[1][3][3][7].k=-1;
napr[1][3][4][1].i=0;
napr[1][3][4][1].j=0;
napr[1][3][4][1].k=0;
napr[1][3][4][2].i=0;
napr[1][3][4][2].j=0;
napr[1][3][4][2].k=-1;

napr[2][3][1][1].i=1;
napr[2][3][1][1].j=0;
napr[2][3][1][1].k=0;
napr[2][3][1][2].i=1;
napr[2][3][1][2].j=0;
napr[2][3][1][2].k=0;
napr[2][3][1][3].i=1;
napr[2][3][1][3].j=0;
napr[2][3][1][3].k=0;
napr[2][3][1][4].i=1;
napr[2][3][1][4].j=0;
napr[2][3][1][4].k=-1;
napr[2][3][2][1].i=0;
napr[2][3][2][1].j=0;
napr[2][3][2][1].k=0;
napr[2][3][2][2].i=0;
napr[2][3][2][2].j=0;
napr[2][3][2][2].k=-1;
napr[2][3][2][3].i=0;
napr[2][3][2][3].j=1;
napr[2][3][2][3].k=0;
napr[2][3][2][4].i=0;
napr[2][3][2][4].j=1;
napr[2][3][2][4].k=-1;
napr[2][3][3][1].i=0;
napr[2][3][3][1].j=0;
napr[2][3][3][1].k=0;
napr[2][3][3][2].i=1;
napr[2][3][3][2].j=0;
```

```
napr[2][3][3][2].k=0;
napr[2][3][3][3].i=2;
napr[2][3][3][3].j=0;
napr[2][3][3][3].k=0;
napr[2][3][3][4].i=0;
napr[2][3][3][4].j=1;
napr[2][3][3][4].k=0;
napr[2][3][3][5].i=0;
napr[2][3][3][5].j=-1;
napr[2][3][3][5].k=0;
napr[2][3][3][6].i=0;
napr[2][3][3][6].j=0;
napr[2][3][3][6].k=1;
napr[2][3][3][7].i=0;
napr[2][3][3][7].j=0;
napr[2][3][3][7].k=-1;
napr[2][3][4][1].i=0;
napr[2][3][4][1].j=0;
napr[2][3][4][1].k=0;
napr[2][3][4][2].i=0;
napr[2][3][4][2].j=0;
napr[2][3][4][2].k=-1;

napr[3][3][1][1].i=0;
napr[3][3][1][1].j=0;
napr[3][3][1][1].k=0;
napr[3][3][1][2].i=0;
napr[3][3][1][2].j=0;
napr[3][3][1][2].k=-1;
napr[3][3][1][3].i=0;
napr[3][3][1][3].j=0;
napr[3][3][1][3].k=0;
napr[3][3][1][4].i=0;
napr[3][3][1][4].j=0;
napr[3][3][1][4].k=0;
napr[3][3][2][1].i=0;
napr[3][3][2][1].j=0;
napr[3][3][2][1].k=0;
napr[3][3][2][2].i=0;
napr[3][3][2][2].j=0;
napr[3][3][2][2].k=-1;
napr[3][3][2][3].i=0;
napr[3][3][2][3].j=1;
napr[3][3][2][3].k=0;
napr[3][3][2][4].i=0;
```

```
napr[3][3][2][4].j=1;
napr[3][3][2][4].k=-1;
napr[3][3][3][1].i=0;
napr[3][3][3][1].j=0;
napr[3][3][3][1].k=0;
napr[3][3][3][2].i=-2;
napr[3][3][3][2].j=0;
napr[3][3][3][2].k=0;
napr[3][3][3][3].i=-1;
napr[3][3][3][3].j=0;
napr[3][3][3][3].k=0;
napr[3][3][3][4].i=0;
napr[3][3][3][4].j=1;
napr[3][3][3][4].k=0;
napr[3][3][3][5].i=0;
napr[3][3][3][5].j=-1;
napr[3][3][3][5].k=0;
napr[3][3][3][6].i=0;
napr[3][3][3][6].j=0;
napr[3][3][3][6].k=1;
napr[3][3][3][7].i=0;
napr[3][3][3][7].j=0;
napr[3][3][3][7].k=-1;
napr[3][3][4][1].i=0;
napr[3][3][4][1].j=0;
napr[3][3][4][1].k=0;
napr[3][3][4][2].i=0;
napr[3][3][4][2].j=0;
napr[3][3][4][2].k=-1;



napr[1][4][1][1].i=0;
napr[1][4][1][1].j=0;
napr[1][4][1][1].k=0;
napr[1][4][1][2].i=1;
napr[1][4][1][2].j=0;
napr[1][4][1][2].k=0;

napr[1][4][2][1].i=0;
napr[1][4][2][1].j=0;
napr[1][4][2][1].k=0;
napr[1][4][2][2].i=0;
napr[1][4][2][2].j=1;
```

```
napr[1][4][2][2].k=0;

napr[1][4][3][1].i=0;
napr[1][4][3][1].j=0;
napr[1][4][3][1].k=0;
napr[1][4][3][2].i=0;
napr[1][4][3][2].j=0;
napr[1][4][3][2].k=1;


for(int i=0; i<Nx; ++i)
  for(int j=0; j<Ny; ++j)
    for(int k=0; k<Nz; ++k)
    {
      carg[i][j][k] = 0;
      arg[i][j][k][0].i = 0;
      arg[i][j][k][0].j = 0;
      arg[i][j][k][0].k = 0;
    };
for(int k=0; k<Nz; ++k)
{
  int uk = k%dNz;
  int vk = uk + dNz;
  int wk = vk + dNz;
  int pk = wk + dNz;
  int var1;
  if(k==uk)
  {
    var1=1;
  }
  else if(k==vk)
  {
    var1=2;
  }
  else if(k==wk)
  {
    var1=3;
  }
  else if(k==pk)
  {
    var1=4;
  }
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
    {
```

```
  for(int var2=1; var2<5; var2++)
  {
    for(int c=1; c<=cnt[var1][var2]; c++)
    {
      int i1=i+napr[G[i][j][k]][var1][var2][c].i;
      int j1=j+napr[G[i][j][k]][var1][var2][c].j;
      int k1=(var2-1)*dNz + uk + napr[G[i][j][k]][
         var1][var2][c].k;

      if ( ((uk + napr[G[i][j][k]][var1][var2][c].k)
         >= dNz) ||
          ((uk + napr[G[i][j][k]][var1][var2][c].k) <
              0) ||
          (i1 >= Nx) ||
          (i1 < 0) ||
          (j1 >= Ny) ||
          (j1 < 0) ) {}
      else
      {
        if (G[i][j][k]!=0)
        {
          bool d = true;
          for(int c2=0; (c2<carg[i1][j1][k1])&&d; ++
             c2)
            if((arg[i1][j1][k1][c2].i==i)&&
                (arg[i1][j1][k1][c2].j==j)&&
                (arg[i1][j1][k1][c2].k==k)) d=false;
          if(d)
          {
            add_gr(arg,carg[i1][j1][k1],i1,j1,k1,i,j,
               k);
          }
        }


      }

    }
  }


}
```

```
    }

  }
```

24 Введение

Текст Текст Текст

Listing 24: Текст

```
void eval_scalars(long double ***u, long double ***R, int
    i1, int j1, int k1, long double &Rn_F1, long double &
    Rn_F2, long double &F1_F1, long double &F2_F2, long
    double &F1_F2)
{
  Rn_F1 = 0;
  Rn_F2 = 0;
  F1_F1 = 0;
  F2_F2 = 0;
  F1_F2 = 0;

  int c = carg[i1][j1][k1];
  for(int oc=0; oc<c; ++oc)
  {
    indexes &a = arg[i1][j1][k1][oc];
    int i = a.i;
    int j = a.j;
    int k = a.k;
    if(!G[i][j][k]) continue;
    long double F1=A1(u,Z,i,j,k)+A1(Z,u,i,j,k)+A2(Z,i,j,k);
    long double F2=A1(Z,Z,i,j,k);
    long double RRn=R[i][j][k];
    long double hh=Hx[i]*Hy[j]*Hz[k];

    Rn_F1+=RRn*F1*hh;
    Rn_F2+=RRn*F2*hh;
    F1_F1+=F1*F1*hh;
    F2_F2+=F2*F2*hh;
    F1_F2+=F1*F2*hh;
  }

}
```

25 Введение

Текст Текст Текст

Listing 25: Текст

```
long double calc_alpha(long double Rn_F1, long double Rn_F2
    , long double F1_F1, long double F2_F2, long double
    F1_F2)
{
  if(fabs(F2_F2)<1e-15)
  {
    if(fabsl(F1_F1)<1e-15)
      return 0;
    return -Rn_F1/F1_F1;
  }

  long double alpha=0;
  long double x[3];
  int t1=cubic(x,6*F1_F2/4.0/F2_F2,(2*F1_F1+4*Rn_F2)/4.0/
      F2_F2,2*Rn_F1/4.0/F2_F2);
  alpha=x[0];
  long double m=2*alpha*Rn_F1+(2*Rn_F2+F1_F1)*alpha*alpha
      +2*alpha*alpha*alpha*F1_F2+alpha*alpha*alpha*alpha*
      F2_F2;
  for(int i=1; i<t1; ++i)
  {
    long double t;
    t=x[i];
    long double m1=2*t*Rn_F1+2*t*t*Rn_F2+t*t*F1_F1+2*t*t*t*
        F1_F2+t*t*t*t*F2_F2;
    if(m>m1)
      alpha=t,m=m1;
  }
  return alpha;
}
```

26 Введение

Текст Текст Текст

Listing 26: Текст

```
int cubic(long double *x,long double a,long double b,long
    double c)
{
  long double q,r,r2,q3;
  q=(a*a-3*b)/9;
  r=(2*a*a*a-9*a*b+27*c)/54;
  r2=r*r;
  q3=q*q*q;
  if(r2<q3)
  {
    long double t=acos(r/sqrt(q3))/3.0;
    x[0]=-2*sqrt(q)*cos(t)-a/3.0;
    x[1]=-2*sqrt(q)*cos(t+M_2PI/3)-a/3.0;
    x[2]=-2*sqrt(q)*cos(t-M_2PI/3)-a/3.0;
    return(3);
  }
  else
  {
    long double A=(r>0?-1:1)*powl(fabs(r)+sqrtl(r2-q3)
        ,1/3.0);
    long double B=(fabsl(A)>1e-15)?(q/A):0;
    x[0]=A+B-a/3.0;
    x[1]=-0.5*(A+B)-a/3.0;
    x[2]=sqrtl(3.0)*0.5*(A-B);
    if(fabsl(x[2])<1e-10) return(2);
    return(1);
  }
}
```

27 Введение

Текст Текст Текст

Listing 27: Текст

```
void tau_iter()
```

```
{
  long double Rn_F1=0,Rn_F2=0,F1_F1=0,F2_F2=0,F1_F2=0;
  for(int i=0; i<Nx; ++i)
  {
    for(int j=0; j<Ny; ++j)
    {
      for(int k=0; k<Nz; ++k)
      {
        if(G[i][j][k])
        {
          long double F1 = A1(U,R,i,j,k)+A1(R,U,i,j,k)+A2(R
             ,i,j,k);
          long double F2 = A1(R,R,i,j,k);
          long double Rn = R[i][j][k];
          long double hh = Hx[i]*Hy[j]*Hz[k];

          Rn_F1+=Rn*F1*hh;
          Rn_F2+=Rn*F2*hh;
          F1_F1+=F1*F1*hh;
          F2_F2+=F2*F2*hh;
          F1_F2+=F1*F2*hh;
        }
      }
    }
  }
  long double tau=0;
  if(fabs(F2_F2)>1e-15)
  {
    long double x[3];
    int t1=cubic(x,6*F1_F2/4.0/F2_F2,(2*F1_F1+4*Rn_F2)/4.0/
       F2_F2,2*Rn_F1/4.0/F2_F2);
    tau=x[0];
    long double m=2*tau*Rn_F1+2*tau*tau*Rn_F2+tau*tau*F1_F1
       +2*tau*tau*tau*F1_F2+tau*tau*tau*tau*F2_F2;
    for(int i=1; i<t1; ++i)
    {
      long double t;
      t=x[i];
      long double m1=2*t*Rn_F1+2*t*t*Rn_F2+t*t*F1_F1+2*t*t*
         t*F1_F2+t*t*t*t*F2_F2;
      if(m>m1)
        tau=t,m=m1;
    }
  }
  else
```

```
    tau=-Rn_F1/F1_F1;

  for(int i=0; i<Nx; ++i)
  {
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        if(G[i][j][k])
          U[i][j][k]+=tau*R[i][j][k];
  }
}
```

28 Введение

Текст Текст Текст

Listing 28: Текст

```
void alpha_iter()
{
  for(int i1=0; i1<Nx; ++i1)
  {
    for(int j1=0; j1<Ny; ++j1)
    {
      for(int k1=0; k1<Nz; ++k1)
      {
        int g=G[i1][j1][k1];
        if( (g) )
        {
          Z[i1][j1][k1]=1.0;
          long double Rn_F1=0,Rn_F2=0,F1_F1=0,F2_F2=0,F1_F2
             =0;
          long double _Rn_F1=0,_Rn_F2=0,_F1_F1=0,_F2_F2=0,
             _F1_F2=0;
          eval_scalars(U,R,i1,j1,k1,Rn_F1,Rn_F2,F1_F1,F2_F2
             ,F1_F2);
          long double alpha=calc_alpha(Rn_F1,Rn_F2,F1_F1,
             F2_F2,F1_F2);
          U[i1][j1][k1]+=alpha;
          Z[i1][j1][k1] = 0;
          residual(i1,j1,k1);
        }
      }
    }
  }


}
```

29 Введение

Текст Текст Текст

Listing 29: Текст

```
void speed_first()
{
  for(int i=0; i<Nx; ++i)
  {
    for(int j=0; j<Ny; ++j)
    {
      for(int k=0; k<Nz; ++k)
      {
        U_2[i][j][k] = U_1[i][j][k];
        U_1[i][j][k] = U[i][j][k];
      }

    }
  }
  Rn_2=Rn_1;
  Rn_1=Rn;
}
```

30 Введение

Текст Текст Текст

Listing 30: Текст

```
void speed_work()
{
  long double Rn_F1=0,Rn_F2=0,F1_F1=0,F2_F2=0,F1_F2=0;
```

```
for(int i=0; i<Nx; ++i)
{
  for(int j=0; j<Ny; ++j)
    for(int k=0; k<Nz; ++k)
      if(G[i][j][k])
      {

        long double F1 = 2*A1(U_2,U_2,i,j,k) - A1(U,U_2,i
           ,j,k) -
                 A1(U_2,U,i,j,k) + A2(U_2,i,j,k) - A2(U,i
                    ,j,k);

        long double F2 = A1(U,U,i,j,k) - A1(U,U_2,i,j,k)
           -
                 A1(U_2,U,i,j,k) + A1(U_2,U_2,i,j,k);

        long double Rn = A(U_2,U_2,i,j,k);
        long double hh = Hx[i]*Hy[j]*Hz[k];

        Rn_F1+=Rn*F1*hh;
        Rn_F2+=Rn*F2*hh;
        F1_F1+=F1*F1*hh;
        F2_F2+=F2*F2*hh;
        F1_F2+=F1*F2*hh;
      }
}

long double tau=0;
if(fabs(F2_F2)>1e-15)
{
  long double x[3];
  int t1=cubic(x,6*F1_F2/4.0/F2_F2,(2*F1_F1+4*Rn_F2)/4.0/
     F2_F2,2*Rn_F1/4.0/F2_F2);
  tau=x[0];
  long double m= 2*tau*Rn_F1+(2*Rn_F2+F1_F1)*tau*tau+2*
     tau*tau*tau*F1_F2+tau*tau*tau*tau*F2_F2;
  for(int i=1; i<t1; ++i)
  {
    long double t;
    t=x[i];
    long double m1=2*t*Rn_F1+2*t*t*Rn_F2+t*t*F1_F1+2*t*t*
       t*F1_F2+t*t*t*t*F2_F2;
    if(m>m1)
      tau=t,m=m1;
  }
  }
  else if(F1_F2>1e-10)
  {
    long double a = 6.0*F1_F2;
    long double b = 2.0*(F1_F1 + 2.0*Rn_F2);
    long double c = 2.0*Rn_F1;

    long double D = b*b - 4*a*c;
    long double
    x1 = ( -b + sqrt(D) )/(2*a),
    x2 = ( -b - sqrt(D) )/(2*a);
    if( (a*x1*x1 + b*x1)<(a*x2*x2 + b*x2) )
      tau = x1;
    else
      tau = x2;
  }
  else
    tau=-Rn_F1/F1_F1;

  long double omega=tau;
  for(int i=0; i<Nx; ++i)
  {
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        if(G[i][j][k])
        {
          long double tmp=U[i][j][k];
          U[i][j][k] = (1+omega)*U_2[i][j][k]-omega*U[i][j
             ][k];
        }
  }
}
```

31 Введение

Текст Текст Текст

Listing 31: Текст

```
void print_texplot_matrix()
{
  char name[40];
```

```
sprintf(name, "Matrixdata%6d.dat", iters);
FILE *f = fopen(name,"w");

fprintf(f,"TITLE = Test\n");
fprintf(f,"VARIABLES = X,Y,Z,U,V,W,P\n");
fprintf(f,"ZONE T=Test,I=%d, J=%d, K=%d, F=POINT\n", Nx,
   Ny, dNz);
for(int k=0; k<dNz; ++k)
{
  for(int j=0; j<Ny; ++j)
  {
    for(int i=0; i<Nx; ++i)
    {
      fprintf(f,"%LF, %LF, %LF,", Cx[i], Cy[j], Cz[k]);

      fprintf(f,"%LF,", U[i][j][k]);

      fprintf(f,"%LF,", U[i][j][k+dNz]);

      fprintf(f,"%LF,", U[i][j][k+2*dNz]);

      fprintf(f,"%LF\n", U[i][j][k+3*dNz]);
    }
  }
}

fclose(f);

strcpy(name,"");
sprintf(name, "Res%6d.dat", iters);
f = fopen(name,"w");

fprintf(f,"TITLE = Test\n");
fprintf(f,"VARIABLES = X,Y,Z,RU,RV,RW,P\n");
fprintf(f,"ZONE T=Test,I=%d, J=%d, K=%d, F=POINT\n", Nx,
   Ny, dNz);
for(int k=0; k<dNz; ++k)
{
  for(int j=0; j<Ny; ++j)
  {
    for(int i=0; i<Nx; ++i)
    {
      fprintf(f,"%LF, %LF, %LF,", Cx[i], Cy[j], Cz[k]);

      fprintf(f,"%LF,", R[i][j][k]);
```

```
        fprintf(f,"%LF,", R[i][j][k+dNz]);

        fprintf(f,"%LF,", R[i][j][k+2*dNz]);

        fprintf(f,"%LF\n", R[i][j][k+3*dNz]);
      }
    }
  }

  fclose(f);

}
```

32 Введение

Текст Текст Текст

Listing 32: Текст

```
void print_texplot()
{
  char name[20];
  sprintf(name, "data%6d.dat", iters);
  FILE *f = fopen(name,"w");

  fprintf(f,"TITLE = Test\n");
  fprintf(f,"VARIABLES = X,Y,Z,U,V,W,P\n");
  fprintf(f,"ZONE T=Test,I=%d, J=%d, K=%d, F=POINT\n", Nx,
     Ny-1, dNz-1);
  for(int k=0; k<dNz-1; ++k)
  {
    for(int j=0; j<Ny-1; ++j)
    {
      for(int i=0; i<Nx; ++i)
      {
        fprintf(f,"%LF, %LF, %LF,", Cx[i], Cy[j], Cz[k]);

        if(i==0)
          fprintf(f,"%LF,",U[i+1][j][k]);
        else if(i==Nx-1)
          fprintf(f,"%LF,",U[i][j][k]);
        else
          fprintf(f,"%LF,", Hx[i]/(Hx[i+1]+Hx[i])*U[i+1][j
             ][k]+Hx[i+1]/(Hx[i+1]+Hx[i])*U[i][j][k] );

        if(j==0)
          fprintf(f,"%LF,",U[i][j][k+dNz]);
        else if(j==Ny-2)
          fprintf(f,"%LF,",U[i][j+1][k+dNz]);
        else
          fprintf(f,"%LF,", Hy[j]/(Hy[j]+Hy[j+1])*U[i][j
             +1][k+dNz]+
              Hy[j+1]/(Hy[j]+Hy[j+1])*U[i][j][k+dNz]);

        if(k==0)
          fprintf(f,"%LF,",U[i][j][k+2*dNz]);
        else if(k==dNz-2)
          fprintf(f,"%LF,",U[i][j][k+1+2*dNz]);
        else
          fprintf(f,"%LF,", Hz[k]/(Hz[k]+Hz[k+1])*U[i][j][k
             +1+2*dNz]+
              Hz[k+1]/(Hz[k]+Hz[k+1])*U[i][j][k+2*dNz]);

        fprintf(f,"%LF\n",U[i][j][k+3*dNz]);
      }
    }
  }

  fclose(f);
}
```

33 Введение

Текст Текст Текст

Listing 33: Текст

```
void load_coords(const char *file_x_name,
        const char *file_y_name,
        const char *file_z_name)
{
    FILE *f = fopen(file_x_name,"r");
```

```
 for(int i=0; i<Nx; ++i)
 {
    fscanf(f,"%LF\n", &Cx[i]);
 }

  fclose(f);

  f = fopen(file_y_name,"r");

  for(int j=0; j<Ny-1; ++j)
  {
    fscanf(f,"%LF\n", &Cy[j]);
  }

  fclose(f);

  f = fopen(file_z_name,"r");

 for(int k=0; k<dNz-1; ++k)
 {
    fscanf(f,"%LF\n", &Cz[k]);
 }

  fclose(f);

  printf("Coordinates has been loaded\n");
}
```

34 Введение

Текст Текст Текст

Listing 34: Текст

```
void load_mask(const char *file_name)
{
  FILE *f = fopen(file_name,"r");

  for(int i=0; i<Nx; ++i)
  {
    for(int k=0; k<dNz-1; ++k)
    {
          for(int j=0; j<Ny-1; ++j)
          {
              fscanf(f,"%d ", &G[i][j][k]);
          }
    }
  }

  fclose(f);
  printf("Mask has been loaded\n");
}
```

35 Введение

Текст Текст Текст

Listing 35: Текст

```
void print_area()
{
    char output_path[] = "surface.vtk";
    print_vtk_header(output_path, Nx, Ny-1, dNz-1);

    FILE *f = fopen(output_path,"a");

  for(int k=0; k<dNz-1; ++k)
  {
    for(int j=0; j<Ny-1; ++j)
    {
      for(int i=0; i<Nx; ++i)
      {
                fprintf(f,"%LF %LF %LF\n", Cx[i],Cy[j],Cz[k
                   ]);
      }
    }
  }

    fclose(f);

    print_vtk_data_header(output_path, Nx, Ny-1, dNz-1);

    f = fopen(output_path,"a");
    for(int k=0; k<dNz-1; ++k)
    {
        for(int j=0; j<Ny-1; ++j)
        {
            for(int i=0; i<Nx; ++i)
            {
                fprintf(f,"%d\n", G[i][j][k]);
            }
        }
    }

  fclose(f);
}
```

36 Введение

Текст Текст Текст

Listing 36: Текст

```
void print_area_points()
{
    char output_path[] = "surface.vtk";
    print_vtk_header_points(output_path, Nx, Ny-1, dNz-1);
    print_vtk_data_header(output_path, Nx, Ny-1, dNz-1);

    FILE *f = fopen(output_path,"a");
    for(int k=0; k<dNz-1; ++k)
    {
        for(int j=0; j<Ny-1; ++j)
        {
            for(int i=0; i<Nx; ++i)
            {
                fprintf(f,"%d\n", G[i][j][k]);
            }
        }
    }

  fclose(f);
}
```

37 Введение

Текст Текст Текст

Listing 37: Текст

```
void print_vtk_data_header(char *output_path, int sizeX,
    int sizeY, int sizeZ)
{
#pragma region HEADER
     string header;
     char line [50];

     sprintf(line,"POINT_DATA %d\n",sizeX*sizeY*sizeZ);
     header.append(line);

     sprintf(line,"SCALARS scalars int\n");
     header.append(line);

     sprintf(line,"LOOKUP_TABLE default\n");
     header.append(line);
#pragma endregion
                          vtk

#pragma region WRITE_FILE
     ofstream output_data;
     output_data.open(output_path,std::ios_base::app | std::
         ios_base::out);
     output_data << header;
     output_data.close();
#pragma endregion

}
```

38 Введение

Текст Текст Текст

Listing 38: Текст

```
void print_vtk_header(char *output_path, int sizeX, int
    sizeY, int sizeZ)
```

```
{
#pragma region HEADER
    string header;
    char line [50];

    sprintf(line,"# vtk DataFile Version 1.0\n");
    header.append(line);

    sprintf(line,"Data file for valves model\n");
    header.append(line);

    sprintf(line,"ASCII\n");
    header.append(line);

    sprintf(line,"DATASET STRUCTURED_GRID\n");
    header.append(line);

    sprintf(line,"DIMENSIONS %d %d %d\n",sizeX,sizeY,sizeZ)
       ;
    header.append(line);

    sprintf(line,"POINTS %d double\n",sizeX*sizeY*sizeZ);
    header.append(line);
#pragma endregion
                        vtk

#pragma region WRITE_FILE
    ofstream output_data;
    output_data.open(output_path);
    output_data << header;
    output_data.close();
#pragma endregion

}
```

39 Введение

Текст Текст Текст

Listing 39: Текст

```
void print_vtk_header_points(char *output_path, int sizeX,
   int sizeY, int sizeZ)
```

```
{
#pragma region HEADER
    string header;
    char line [50];

    sprintf(line,"# vtk DataFile Version 1.0\n");
    header.append(line);

    sprintf(line,"Data file for valves model\n");
    header.append(line);

    sprintf(line,"ASCII\n");
    header.append(line);

    sprintf(line,"DATASET STRUCTURED_POINTS\n");
    header.append(line);

    sprintf(line,"DIMENSIONS %d %d %d\n",sizeX,sizeY,sizeZ)
       ;
    header.append(line);

    sprintf(line,"ORIGIN %d %d %d\n",0,0,0);
    header.append(line);

    sprintf(line,"SPACING %d %d %d\n",1,1,1);
    header.append(line);
#pragma endregion
                         vtk

#pragma region WRITE_FILE
    ofstream output_data;
    output_data.open(output_path);
    output_data << header;
    output_data.close();
#pragma endregion

}
```

40 Введение

Текст Текст Текст

Listing 40: Текст

```
void print_info()
{
  print_texplot();
  print_texplot_matrix();

  char fn [20];
  sprintf(fn, "info%6d.txt", iters);
  FILE *f = fopen(fn,"w");
  fprintf(f,"iters = %d\n\n", iters);
  fprintf(f,"r0 = %LF\n", R0);
  fprintf(f,"rn = %LF\n", Rn);
  fprintf(f,"rn/ro = %LF\n", Rn/R0);

  long double max_ri = fabs(R[0][0][0]);
  int im = 0, jm = 0, km = 0;
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        if( max_ri < fabs(R[i][j][k]) )
        {
          max_ri = fabs(R[i][j][k]);
          im = i;
          jm = j;
          km = k;
        }
  fprintf(f,"max_ri = %LF at (%d,%d,%d)\n\n", max_ri, im,
     jm, km);

  long double s1=0, s2=0;
  for(int j=0; j<Ny; ++j)
    for(int k=0; k<dNz; ++k)
    {
      s1 += U[1][j][k]*Hy[j]*Hz[k];
      s2 += U[Nx-1][j][k]*Hy[j]*Hz[k];
    }

  fprintf(f,"s1 = %LF\n", s1);
  fprintf(f,"s2 = %LF\n", s2);
  fprintf(f,"ds = %lf\n", fabs(s1-s2));
  fclose(f);

}
```

41 Введение

Текст Текст Текст

Listing 41: Текст

```
void X_init()
{
  for(int d=0; d<Rd; ++d)
  {
    long double s = sqrtl(2)/2*r_L*d/(Rd-1);
    long double t = sqrtl(r_L*r_L-s*s);
    Cz[Rd_2-1+d] = Cy[Rd_2-1+d] = s;
    Cz[Rd_2-1-d] = Cy[Rd_2-1-d] = -s;
    Cz[Rd_4-1-d] = Cy[Rd_4-1-d] = t;
    Cz[d] = Cy[d] = -t;
  }

  Cy[Ny-1] = 2*Cy[Ny-2]-Cy[Ny-3];
  Cz[dNz-1] = 2*Cz[dNz-2]-Cz[dNz-3];

  for(int i=0; i<Nx1; ++i)
    Cx[i] = x1_L*i/(Nx1-1);
  for(int i=1; i<Nx2; ++i)
    Cx[Nx1-1 + i] = Cx[Nx1-1+i-1] + abs(Cz[i]-Cz[i-1]);
  x2_L = Cx[Nx1-1 + Nx2-1] - x1_L;
  for(int i=1; i<Nx3; ++i)
    Cx[Nx1+Nx2-2 + i] = Cx[Nx1-1 + Nx2-1] + x3_L*i/(Nx3-1);
  for(int i=1; i<Nx4; ++i)
    Cx[Nx1+Nx2+Nx3-3 + i] = Cx[Nx1-1 + Nx2-1 + Nx3-1 + i-1]
        + abs(Cz[i]-Cz[i-1]);
  x4_L = Cx[Nx1-1 + Nx2-1 + Nx3-1 + Nx4-1] - Cx[Nx1-1 + Nx2
     -1 + Nx3-1];
  for(int i=1; i<Nx5; ++i)
    Cx[Nx1+Nx2+Nx3+Nx4-4 + i] = Cx[Nx1-1 + Nx2-1 + Nx3-1 +
       Nx4-1] + x5_L*i/(Nx5-1);
  x_L = Cx[Nx1+Nx2+Nx3+Nx4+Nx5-5];


  for(int i=1; i<Nx; ++i)
    Hx[i]=Cx[i]-Cx[i-1];
  Hx[0]=Hx[1];
  for(int j=1; j<Ny; ++j)
    Hy[j]=Cy[j]-Cy[j-1];
  Hy[0]=Hy[1];
  for(int k=1; k<dNz; ++k)
    Hz[k]=Cz[k]-Cz[k-1];
  Hz[0]=Hz[1];

  for(int i=0; i<dNz; ++i)
  {
    Cz[i+3*dNz] = Cz[i+2*dNz] = Cz[i+1*dNz] = Cz[i];
    Hz[i+3*dNz] = Hz[i+2*dNz] = Hz[i+1*dNz] = Hz[i];
  }

  {
    FILE *f = fopen("h.txt","w");
    for(int i=0; i<Nx; ++i)
      fprintf(f,"%LF ",Cx[i]);
    fprintf(f,"\n");

    for(int j=0; j<Ny; ++j)
      fprintf(f,"%LF ",Cy[j]);
    fprintf(f,"\n");

    for(int k=0; k<Nz; ++k)
      fprintf(f,"%LF ",Cz[k]);
    fprintf(f,"\n");
    fclose(f);
  }
}
```

42 Введение

Текст Текст Текст

Listing 42: Текст

```
void G_init()
{
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k) G[i][j][k] = 1;

  for(int j=0; j<Ny; ++j)
    for(int k=0; k<dNz; ++k) G[0][j][k] = 0;
  for(int i=0; i<Nx; ++i)
    for(int k=0; k<dNz; ++k) G[i][Ny-1][k] = 0;
  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j) G[i][j][dNz-1] = 0;

    for(int k=0; k<Nz1; ++k)
    {
        for(int j=k; j<Rd_2*2-k; ++j)
            for(int i=Nx1-1+k + 1; i<Nx1+Nx2+Nx3-3+k; ++i)
            {
                  G[i][j][k] = 0;
                  G[i][j][dNz-2-k] = 0;
            }
    }

  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Rd_2; ++j)
      for(int k=Rd_2-1-j; k<Rd_2; ++k)
      {
        G[i][Rd_2-1+j][Rd_2-1+k] = 0;
        G[i][Rd_2-1-j][Rd_2-1+k] = 0;
        G[i][Rd_2-1+j][Rd_2-1-k] = 0;
        G[i][Rd_2-1-j][Rd_2-1-k] = 0;
      };

  for(int j=0; j<Rd_2-1; ++j)
    for(int k=0; k<Rd_2-1-j; ++k)
    {
      G[1][Rd_2-1+j][Rd_2-1+k] = 2;
      G[1][Rd_2-1-j][Rd_2-1+k] = 2;
      G[1][Rd_2-1+j][Rd_2-1-k] = 2;
      G[1][Rd_2-1-j][Rd_2-1-k] = 2;
      G[Nx-1][Rd_2-1+j][Rd_2-1+k] = 3;
      G[Nx-1][Rd_2-1-j][Rd_2-1+k] = 3;
      G[Nx-1][Rd_2-1+j][Rd_2-1-k] = 3;
      G[Nx-1][Rd_2-1-j][Rd_2-1-k] = 3;
    };

  for(int k=0; k<Nz1; ++k)
  {
    for(int j=Rd_2-k; j<Rd_2+k; ++j)
      for(int i=Nx1-1+k; i<Nx1+Nx2+Nx3-3+k; ++i)
      {
        G[i][j][k +dNz] = 0;
        G[i][j][dNz-2-k +dNz] = 0;
      }
  }

  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Ny; ++j) G[i][j][dNz+dNz-1] = 0;

  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Rd_2; ++j)
      for(int k=Rd_2-1-j; k<Rd_2; ++k)
      {
        G[i][Rd_2-1+j+1][Rd_2-1+k+dNz] = 0;
        G[i][Rd_2-1-j][Rd_2-1+k+dNz] = 0;
        G[i][Rd_2-1+j+1][Rd_2-1-k+dNz] = 0;
        G[i][Rd_2-1-j][Rd_2-1-k+dNz] = 0;
      };

  for(int j=0; j<Rd_2-1; ++j)
    for(int k=0; k<Rd_2-1-j; ++k)
    {
      G[0][Rd_2-1+j+1][Rd_2-1+k+dNz] = 0;
      G[0][Rd_2-1-j][Rd_2-1+k+dNz] = 0;
      G[0][Rd_2-1+j+1][Rd_2-1-k+dNz] = 0;
      G[0][Rd_2-1-j][Rd_2-1-k+dNz] = 0;
      G[Nx-1][Rd_2-1+j+1][Rd_2-1+k+dNz] = 0;
      G[Nx-1][Rd_2-1-j][Rd_2-1+k+dNz] = 0;
      G[Nx-1][Rd_2-1+j+1][Rd_2-1-k+dNz] = 0;
      G[Nx-1][Rd_2-1-j][Rd_2-1-k+dNz] = 0;
    };


  for(int k=1; k<Nz1; ++k)
  {
    for(int j=Rd_2-1-k; j<Rd_2+k; ++j)
      for(int i=Nx1-1+k; i<Nx1+Nx2+Nx3-3+k; ++i)
      {
        G[i][j][k +2*dNz] = 0;
        G[i][j][dNz-1-k +2*dNz] = 0;
      }
  }


  for(int i=0; i<Nx; ++i)
    for(int k=0; k<dNz; ++k) G[i][Ny-1][2*dNz+k] = 0;

  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Rd_2; ++j)
```

```
    for(int k=Rd_2-1-j; k<Rd_2; ++k)
    {
      G[i][Rd_2-1+j][Rd_2-1+k+1+dNz*2] = 0;
      G[i][Rd_2-1-j][Rd_2-1+k+1+dNz*2] = 0;
      G[i][Rd_2-1+j][Rd_2-1-k+dNz*2] = 0;
      G[i][Rd_2-1-j][Rd_2-1-k+dNz*2] = 0;
    };

for(int j=0; j<Rd_2-1; ++j)
  for(int k=0; k<Rd_2-1-j; ++k)
  {
    G[0][Rd_2-1+j][Rd_2-1+k+1+dNz*2] = 0;
    G[0][Rd_2-1-j][Rd_2-1+k+1+dNz*2] = 0;
    G[0][Rd_2-1+j][Rd_2-1-k+dNz*2] = 0;
    G[0][Rd_2-1-j][Rd_2-1-k+dNz*2] = 0;
    G[Nx-1][Rd_2-1+j][Rd_2-1+k+1+dNz*2] = 0;
    G[Nx-1][Rd_2-1-j][Rd_2-1+k+1+dNz*2] = 0;
    G[Nx-1][Rd_2-1+j][Rd_2-1-k+dNz*2] = 0;
    G[Nx-1][Rd_2-1-j][Rd_2-1-k+dNz*2] = 0;
  };


for(int k=1; k<Nz1-1; ++k)
{
  for(int j=Rd_2-k; j<Rd_2-1+k; ++j)
    for(int i=Nx1-1+k+1; i<Nx1+Nx2+Nx3-3+k-1; ++i)
    {
      G[i][j][k +3*dNz] = 0;
      G[i][j][dNz-2-k +3*dNz] = 0;
    }
}

for(int i=0; i<Nx; ++i)
  for(int j=0; j<Ny; ++j) G[i][j][3*dNz+dNz-1] = 0;
for(int i=0; i<Nx; ++i)
  for(int k=0; k<dNz; ++k) G[i][Ny-1][3*dNz+k] = 0;

for(int i=0; i<Nx; ++i)
  for(int j=0; j<Rd_2; ++j)
    for(int k=Rd_2-1-j+1; k<Rd_2; ++k)
    {
      G[i][Rd_2-1+j][Rd_2-1+k+dNz*3] = 0;
      G[i][Rd_2-1-j][Rd_2-1+k+dNz*3] = 0;
      G[i][Rd_2-1+j][Rd_2-1-k+dNz*3] = 0;
      G[i][Rd_2-1-j][Rd_2-1-k+dNz*3] = 0;
      };

  for(int j=0; j<Ny; ++j)
    for(int k=0; k<dNz; ++k)
    {
      G[0][j][3*dNz+k] = 0;
      G[Nx-1][j][3*dNz+k] = 0;
    };

  {
    ofstream f("check_grid.txt");
    for(int k=dNz-1; k>=0; --k)
    {
      for(int j=Ny-1; j>=0; --j)
      {
        for(int i=0; i<Nx; ++i)
                    f<<Cx[i]<<' '<<Cy[i]<<' '<<Cz[i]<<' '<<
                       G[i][j][k]<<' ';
        f<<"\n";
      };
    };
    f.close();
  }

    load_mask("prism.mask");
    load_coords("prism.x.coord","prism.y.coord","prism.z.
       coord");
    print_area();
}
```

43 Введение

Текст Текст Текст

Listing 43: Текст

```
void U_init()
{
  int vortex_inside_only = 1;
  for(int i=0; i<Nx; ++i)
  {
    for(int j=0; j<Ny; ++j)
      for(int k=0; k<Nz; ++k)
        U[i][j][k]=0;
  }


  for(int i=0; i<Nx; ++i)
    for(int j=0; j<Rd_2; ++j)
      for(int k=0; k<Rd_2-j; ++k)
      {
        long double p = p_left - (p_left-p_right)*i/(Nx-1);
        if( (G[i][Rd_2-1+j][3*dNz+Rd_2-1+k]) ||(i==0)||(i==
           Nx-1) )
          U[i][Rd_2-1+j][3*dNz+Rd_2-1+k]= p;
        if( (G[i][Rd_2-1-j][3*dNz+Rd_2-1+k]) ||(i==0)||(i==
           Nx-1))
          U[i][Rd_2-1-j][3*dNz+Rd_2-1+k]= p;
        if( (G[i][Rd_2-1+j][3*dNz+Rd_2-1-k]) ||(i==0)||(i==
           Nx-1))
          U[i][Rd_2-1+j][3*dNz+Rd_2-1-k]= p;
        if( (G[i][Rd_2-1-j][3*dNz+Rd_2-1-k]) ||(i==0)||(i==
           Nx-1))
          U[i][Rd_2-1-j][3*dNz+Rd_2-1-k]= p;
      }

  long double um = 1;
  for(int i=1; i<Nx; ++i)
    for(int j=0; j<Rd_2-1; ++j)
      for(int k=0; k<Rd_2-1-j; ++k)
      {
        long double
        y = Cy[Rd_2-1+j],
        z = Cz[Rd_2-1+k];

        if( (G[i][Rd_2-1+j][Rd_2-1+k]))
          U[i][Rd_2-1+j][Rd_2-1+k] = ( r_L*r_L-(y*y+z*z) )*
             um*um/r_L/r_L;
        if( (G[i][Rd_2-1-j][Rd_2-1+k]))
          U[i][Rd_2-1-j][Rd_2-1+k] = ( r_L*r_L-(y*y+z*z) )*
             um*um/r_L/r_L;
        if( (G[i][Rd_2-1+j][Rd_2-1-k]))
          U[i][Rd_2-1+j][Rd_2-1-k] = ( r_L*r_L-(y*y+z*z) )*
             um*um/r_L/r_L;
        if( (G[i][Rd_2-1-j][Rd_2-1-k]))
          U[i][Rd_2-1-j][Rd_2-1-k] = ( r_L*r_L-(y*y+z*z) )*
             um*um/r_L/r_L;
        };

}
```

44 Введение

Текст Текст Текст

Listing 44: Текст

```
void vars_init()
{
  eps = 0.000001;

  x1_L = 0.06;
  x2_L = 0.06;
  x3_L = 0.06;
  x4_L = 0.06;
  x5_L = 0.12;

  x_L = x1_L + x2_L + x3_L + x4_L + x5_L;
  y_L = 0.2;
  z_L = 0.2;
  r_L = 0.1;

  Nx1 = 11;
  Nx2 = 11;
  Nx3 = 11;
  Nx4 = 11;
  Nx5 = 21;
  Nx = Nx1+Nx2+Nx3+Nx4+Nx5 - 4;

  Nz1 = 11;
  Nz2 = 5;
  Nz3 = 11;
  Rd = 7;

  nu = 1e-2;
  rho = 1;

  p_left = 1;
  p_right = 0;
}
```

45 Введение

Текст Текст Текст

Listing 45: Текст

```
void init()
{
  vars_init();

  y_L = 2*r_L;
  z_L = 2*r_L;

  Rd_2 = Rd*2-1;
  Rd_4 = Rd_2*2-1;

  Ny = dNz = Rd_4;

  ++Ny;
  ++dNz;
  Nz = 4*dNz;

  alloc(arg,Nx,Ny,Nz);
  alloc(func,Nx,Ny,Nz);
  alloc(carg,Nx,Ny,Nz);
  alloc(cfunc,Nx,Ny,Nz);
  alloc(groups,Nx,Ny,Nz);

  alloc(U,Nx,Ny,Nz);
  alloc(R,Nx,Ny,Nz);
  alloc(G,Nx,Ny,Nz);
  alloc(Z,Nx,Ny,Nz);

  alloc(U_1,Nx,Ny,Nz);
  alloc(U_2,Nx,Ny,Nz);

  Rn_1 = Rn_2 = 0;

  for(int i=0; i<Nx; ++i)
```

```
{
  for(int j=0; j<Ny; ++j)
    for(int k=0; k<Nz; ++k)
      U_2[i][j][k] = U_1[i][j][k] = 0;
}


Hx = new long double [Nx];
Hy = new long double [Ny];
Hz = new long double [Nz];
Cx = new long double [Nx];
Cy = new long double [Ny];
Cz = new long double [Nz];

X_init();
G_init();
U_init();

residual();
Rn = norm(R);
R0 = Rn;

for(int i=0; i<Nx; ++i)
{
  for(int j=0; j<Ny; ++j)
  {
    for(int k=0; k<Nz; ++k)
    {
      Z[i][j][k]=0;
    }
  }
}
init_gr();
print_gr();
set_gr();
FILE *fgr = fopen("gr.txt","w");
for(int i=0; i<Nx; ++i)
{
  for(int j=0; j<Ny; ++j)
  {
    for(int k=0; k<Nz; ++k)
    {
      fprintf(fgr,"%3d ",groups[i][j][k]);
    }
  }
```

```
    fprintf(fgr,"\n");
  }
  fclose(fgr);
}
```

46 Введение

Текст Текст Текст

Listing 46: Текст

```
void run()
{

  residual();
  iters = 0;
  print_info();
  do
  {

    ++iters;
    speed_first();
    long double R1 = norm(R);
    tau_iter();
    residual();
    long double R2 = norm(R);
    alpha_iter();
    residual();
    long double R3 = norm(R);

    if( (iters>5)&&( (iters%100==0)||(iters%100==1)||(iters
       %100==2)||(iters%100==3)||(iters%100==4) ) )
    {
      speed_work();
      residual();
    }
    long double R4 = norm(R);

    if( (R4>R3) )
    {
      printf("Error on speed\n");
    }
    else if ( (R3>R2) )
    {
      printf("Error on alpha\n");
    }
    else if ( (R2>R1) )
    {
    }

    Rn = norm(R);

    if(iters%100==5)
    {
      printf("%5d: %3.8LF %3.8LF\n", iters, Rn, Rn/R0);
    }

    if(iters%1000==5)
    {
      print_info();
    }
  }
  while (Rn/R0>eps);
  print_texplot();
}
```

47 Введение

Текст Текст Текст

Listing 47: Текст

```
void down()
{
  del(arg,Nx,Ny,Nz);
  del(func,Nx,Ny,Nz);
  del(carg,Nx,Ny,Nz);
  del(cfunc,Nx,Ny,Nz);
  del(groups,Nx,Ny,Nz);

  del(U,Nx,Ny,Nz);
  del(R,Nx,Ny,Nz);
  del(G,Nx,Ny,Nz);
  del(Z,Nx,Ny,Nz);
  delete [] Hx;
  delete [] Hy;


}
```

48 Введение

Текст Текст Текст

Listing 48: Текст

```
int main()
{
  init();
  run();
  down();
  return 0;
}
```

49 Введение

Текст Текст Текст