

Príloha

```
#include<stdio.h> /*hlavičkový súbor*/
#include<stdlib.h> /*hlavičkový súbor potrebný pre exit*/
#include <math.h> /*hlavickový súbor obsahujúci matematické operácie*/

float gam,aL,aR,AL,AR,BL,BR,uL,pL,pR,uR,roL,roR,delu,pmin,pmax,fa,fb,p,np,
nroL,nroR,nu,x,t,vro,vp,vu;
/*
-gam:= \gamma (Adiabatická konštanta)
-a_L, a_R:= a_L, a_R (rýchlosť zrýchlenia)
-Hodnoty rýchlosti, hustoty a tlaku stavu w_L:
uL, roL, pL:= u_L, \rho_L, p_L
-Hodnoty rýchlosti, hustoty a tlaku stavu w_R:
uR, roR, pR:= u_R, \rho_R, p_R
-delu:= u_R-u_L
-pmin:= p_{min} (minimálna hodnota z hodnôt p_L a p_R)
-pmax:= p_{max} (maximálna hodnota z hodnôt p_L a p_R)
-Funkčná hodnota funkcie danej vzorcom (2.116) z vety 2.19 v bode pmin:
fa:= f(pmin)
-Funkčná hodnota funkcie danej vzorcom (2.116) z vety 2.19 v bode pmax:
fb:= f(pmax)
-p:= do tejto premennej ukladáme nájdene p*
-Hodnoty tlaku p*, hustoty ro_{*L}, hustoty ro_{*R}, rýchlosti u_* v
hviezdovitej oblasti: np, nroL, nroR, nu
-(x,t):= priestorová a časová súradnica, kde užívateľ potrebuje
nájsť riešenie
-vro,vp,vu:= hodnoty hustoty, tlaku a rýchlosti v bode (x,t)
*/

float fL(float p)
/*Funkcia vypočíta v bode p funkčnú hodnotu f_L(p), (f_L veta(2.116)).*/
{
    float f,m,n,x;
    m=sqrt(AL/(p+BL));
    x=(gam-1)/(2*gam);
    n=pow((p/pL),x);
    if (p>pL) f=(p-pL)*m;
        else f=2*aL*(n-1)/(gam-1);
    return (f);
}

float fR(float p)
/*Funkcia vypočíta v bode p funkčnú hodnotu f_R(p), (f_R veta (2.116)).*/
{
    float f,m,n,x;
```

```

    m=sqrt(AR/(p+BR));
    x=(gam-1)/(2*gam);
    n=pow((p/pR),x);
    if (p>pR) f=(p-pR)*m;
        else f=2*aR*(n-1)/(gam-1);
    return (f);
}

float prva(float p)
/*Funkcia vypočíta v bode p funkčnú hodnotu f'(p)
= prvú deriváciu funkcie f = pre deriváciu platí vzorec (2.162).*/
{
    float h,L,R,g,x;
    h=sqrt(AL/(BL+p));
    g=-(gam+1)/(2*gam);
    x=1-(p-pL)/(2*(BL+p));
    if (p>pL) {L=h*x;} else {L=pow((p/pL),g)/(roL*aL);}
    h=pow(AR/(BR+p),1/2);
    x=1-(p-pR)/(2*(BR+p));
    if (p>pR) {R=h*x;} else {R=pow((p/pR),g)/(roR*aR);}
    return (L+R);
}

float u(float p)
/*Funkcia počíta pre p hodnotu u pomocou vzorca (2.122)*/
{
    return ((uL+uR)/2+(fR(p)-fL(p))/2);
}

float ro_L(float p)
/*Funkcia vypočíta v bode p funkčnú hodnotu ro_L(p)((2.160)).*/
{
    float x,a,b;
    if (p>pL) {
        a=p/pL+(gam-1)/(gam+1);
        b=(gam-1)*p/((gam+1)*pL)+1;
        x=roL*a/b;
    } else {
        a=1/gam;
        b=p/pL;
        x=roL*pow(b,a);
    }
    return (x);
}

float ro_R(float p)
/*Funkcia vypočíta v bode p funkčnú hodnotu ro_R(p)((2.161)).*/
{

```

```

float a,b,x;
if (p>pR) {
    a=p/pR+(gam-1)/(gam+1);
    b=(gam-1)*p/((gam+1)*pR)+1;
    x=roR*a/b;
} else {
    a=1/gam;
    b=p/pR;
    x=roR*pow(b,a);
}
return (x);
}

float newton(float p)
/*Funkcia popisuje Newtonovú metódu, vstup je nultá iterácia.
Pokiaľ k-tá iterácia bude menšia ako TOL, iterovanie sa ukončí.*/
{
    float TOL,CHA,p0,p1,x,y;
    int i;
    TOL=0.0000001;
    p0=fmax(p,TOL);
    p1=fmax(p,TOL);
    CHA=p0;
    while (CHA>=TOL) {
        x=fR(p0)+fL(p0)+delu;
        y=prva(p0);
        p1=p0-x/y;
        CHA=fabs(p1-p0)/((p1+p0)/2);
        p0=p1;
    }
    return (p1);
}

float hladaniep(void)
/*Funkcia hľadá riešenie rýchlosti u_* splňujúcu rovnosť f(p)=0.
Pokiaľ je treba zvolí 0.iteráciu a aplikuje na ňu Newtonovú metódu. */
{
    float sa,sro,sp,ppv;
    sa=aR+aL;
    sro=roR+roL;
    sp=pR+pL;
    ppv=sp/2-delu*sro*sa/8;
    if (fa>0 && fb>0) {
        float h,n;
        h=(gam-1)/(2*gam);
        n=(-h*gam*delu+sa)/(aL/(pow(pL,h))+aR/(pow(pR,h)));
        p=pow(n,1/h);
    } else if (fa<=0 && fb>=0) p=newton(ppv);
}

```

```

        else {
            float gL,gR;
            gL=sqrt(AL/(ppv+BL));
            gR=sqrt(AR/(ppv+BR));
            p=(gL*pL+gR*pR-delu)/(gR+gL);
            p=newton(p);
        }
    return(p);
}

float w(float S)
/*Funkcia urči riešenie v bode (x,t).*/
{
    float naL,naR,SHL,STL,SHR,STR,SL,SR,y1,y2,z,i,j,k;
    y1=(gam+1)/(2*gam);
    y2=(gam-1)/(2*gam);
    z=(gam-1)/2;
    i=2/(gam+1);
    j=(gam-1)/(gam+1);
    k=2/(gam-1);
    naL=aL*pow(np/pL,y2);
    naR=aR*pow(np/pR,y2);
    SHL=uL-aL;
    STL=nu-naL;
    SHR=uR+aR;
    STR=nu+naR;
    SL=uL-aL*sqrt(y1*np/pL+y2);
    SR=uR+aR*sqrt(y1*np/pR+y2);
    if (S<=nu) {
        if (np<=pL) {if (S<SHL)
            {
                vro=roL;
                vu=uL;
                vp=pL;
            }
            else {if (STL<S)
                {
                    vu=nu;
                    vro=nroL;
                    vp=np;
                } else
                {
                    vro=roL*pow(i+j*(uL-S)/aL,k);
                    vu=i*(aL+z*uL+S);
                    vp=pL*pow(i+(j/aL)*(uL-S),gam*k);
                }
            }
        }
    }
}

```

```

else {if (S<SL) {
    vro=roL;
    vp=pL;
    vu=uL;
    } else {
        vro=nroL;
        vp=np;
        vu=nu;
    }
    }
else {
    if (np<=pR) {if (S<=STR)
    {
        vro=nroR;
        vu=nu;
        vp=np;
    }
    else {if (S<=SHR)
    {
        vro=roR*pow(i-j*(uR-S)/aR,k);
        vu=i*(-aR+z*uR+S);
        vp=pR*pow(i-(j/aR)*(uR-S),gam*k);
    }
    else {
        vu=uR;
        vro=roR;
        vp=pR;
    }
    }
    }
else {if (S>=SR) {
    vro=roR;
    vp=pR;
    vu=uR;
    } else {
        vro=nroR;
        vp=np;
        vu=nu;
    }
    }
}
}
}
main()
{
/*Vstup: program v tomto kroku čaká od uživateľa vstupné hodnoty tlaku,

```

```

rýchlosti a hustoty a adiabatickej konštanty. */
printf("roL: ");
scanf("%f", &roL);
printf("uL: ");
scanf("%f", &uL);
printf("pL: ");
scanf("%f", &pL);
printf("roR: ");
scanf("%f", &roR);
printf("uR: ");
scanf("%f", &uR);
printf("pR: ");
scanf("%f", &pR);
printf("Gamma: ");
scanf("%f", &gam);
printf("x: ");
scanf("%f", &x);
printf("t: ");
scanf("%f", &t);

/*Overenie vstupných údajov:*/
if ((t<0)|| (gam<=1)|| (roL<=0)|| (roR<=0)|| (pL<=0)|| (pR<=0)) {
    printf("Zle zadany vstup.");
    exit(0);}

else {
    if (t==0) {
        if (x==0) printf("V bode (0,0) je nespojitost. ");
        else {
            if (x>0) printf("p : %f\nu: %f\nro: %f\n",pR,uR,roR);
            else printf("p : %f\nu: %f\nro: %f\n",pL,uL,roL);
        }
        exit(0);}
    }

/*Výpočet aR,aL a delta u:*/
aL=sqrt(gam*pL/roL);
aR=sqrt(gam*pR/roR);
delu=uR-uL;

/*Overenie splnenia podmienky kladného tlaku.
Ak neplatí, program sa ukončí a oznámi, čo sa stalo.*/
if ((2*aL+2*aR)/(gam-1)<=delu) {
printf ("Neplati pressure positivity condition \n");
return;
exit(0);}

/*Výpočet AL,AR,BL,BR používaných vo vete (4.2.1).*/
AL = 2/((gam+1)*roL);

```

```

AR = 2/((gam+1)*roR);
BL = (gam-1)*pL/(gam+1);
BR = (gam-1)*pR/(gam+1);

pmax=(pL>pR?pL:pR); /*= maximalná hodnota pre tlak z hodnot pL a pR*/
pmin=(pL>pR?pR:pL); /*= minimalná hodnota pre tlak z hodnot pL a pR*/
fa=fR(pmin)+fL(pmin)+delu; /*= funkčná hodnota fcie f v bode pmin*/
fb=fR(pmax)+fL(pmax)+delu; /*= funkčná hodnota fcie f v bode pmax*/
hladaniep();

/*Do nasledujúcich premenných sa uložia vypočítané hodnoty tlaku,
rýchlosti a hustôt v hviezdovitej oblasti.*/
np=p;
nroL=ro_L(p);
nroR=ro_R(p);
nu=u(p);

w(x/t);
printf("p : %f\nu: %f\nro: %f\n",vp,vu,vro);
return;
}

```