

Help

```
#include "hescir1d_std.h"

#if defined(PremiaCurrentVersion) && PremiaCurrentVersion <
    (2010+2) //The "#else" part of the code will be freely av
    ailable after the (year of creation of this file + 2)
static int CHK_OPT(AP_MedvedevScaillet)(void *Opt, void *
    Mod)
{
    return NONACTIVE;
}
int CALC(AP_MedvedevScaillet)(void*Opt,void *Mod,Pricing
    Method *Met)
{
    return AVAILABLE_IN_FULL_PREMIA;
}
#else
////////////////////////////////////
    //
static void fy0(double S, double K, double t, double v,
    double * result )
{
    * result =log(K/S)/v/sqrt(t);
}
static void fb1(double sigmav, double * result)
{
    * result=sigmav/2.;
}
static void fav(double kv, double vbar, double b1, double
    v, double * result)
{
    *result=(kv*(vbar-pow(v,2))-pow(b1,2))/2/v;
}
static void fav1(double kv, double vbar, double b1, double
    v, double * result)
{
    * result=-(kv*vbar-pow(b1,2))/2/pow(v,2)-kv/2;
}
static void fav2(double kv, double vbar, double b1, double
    v, double * result)
```

```

{
    * result=(kv*vbar-pow(b1,2))/pow(v,3);
}
static void ff(double r, double d, double * result)
{
    * result=r-d;
}
static void far(double kr, double rbar, double r, double *
    result)
{
    * result=kr*(rbar-r);
}
static void far1(double kr, double * result)
{
    * result=-kr;
}
static void fbr(double sigmar, double r, double * result)
{
    * result=sigmar*sqrt(r);
}
static void fbr1(double sigmar, double r, double * result)
{
    * result=sigmar/sqrt(r)/2;
}
static void fbr2(double sigmar, double r, double * result)
{
    * result=-(1/4)*sigmar/pow(r,(3/2));
}

static void theta(double S, double K, double t, double v,
    double * result)
{
    * result=log(K/S)/v/sqrt(t);
}

//Coefficients and their derivatives
static void fC1(double N0, double n0, double K, double y,
    double v, double * result)
{
    * result=K*y*v/(N0*y+n0);
}

```

```

static void fVC1(double N0, double n0, double K, double y,
    double * result)
{
    * result= K*y/(N0*y+n0);
}
static void fC2(double N0, double n0, double C1, double K,
    double v, double f, double y, double b1, double q1, double * res
    ult)
{
    * result=1/2.*(-N0*C1*pow(v,2)+2*N0*C1*f-n0*y*b1*q1*C1-
        K*pow(y,2)*pow(v,3))/(v*(N0*pow(y,2)+N0+n0*y));
}
static void fVC2(double N0, double n0, double C1, double K,
    double y, double v, double * result)
{
    * result=- (N0*C1+K*pow(y,2)*v)/(N0*pow(y,2)+N0+n0*y);
}
static void fRC2(double N0, double n0, double C1, double y,
    double v, double * result)
{
    * result=N0*C1/(v*(N0*pow(y,2)+N0+n0*y));
}
static void fVVC2(double N0, double n0, double C1, double
    K, double v, double y, double * result)
{
    * result=- (N0*C1+K*pow(y,2)*v)/(v*(N0*pow(y,2)+N0+n0*y))
        ;
}
static void fC3(double N0, double n0, double C1, double C2,
    double VC2, double RC2, double b1, double br, double q1,
    double q2, double av, double K, double v, double y, double r,
    double f, double * result )
{
    *result=1/24.*(-3*n0*C1*pow(v,4)+48*N0*y*b1*q1*VC2*pow(
        v,2)+48*n0*b1*q1*VC2*pow(v,2)+24*n0*r*C1*pow(v,2)-4*n0*pow(
        b1,2)*pow(y,2)*C1-24*N0*y*C2*pow(v,3)+24*N0*y*r*C1*pow(v,2)
        -3*n0*pow(b1,2)*pow(q1,2)*C1*pow(y,4)+6*n0*b1*pow(y,2)*q1*
        C1*pow(v,2)-24*n0*C2*pow(v,3)-96*N0*y*b1*q1*C2*v-72*n0*b1*
        q1*C2*v-12*n0*av*v*C1-12*n0*b1*pow(y,2)*q1*C1*f+48*N0*y*C2*
        v*f+12*n0*b1*q1*C1*f+48*N0*y*br*q2*pow(v,2)*RC2+48*n0*br*q2
        *pow(v,2)*RC2-12*n0*C1*pow(f,2)-12*n0*br*q2*v*C1+12*n0*C1*

```

```

        pow(v,2)*f+48*n0*C2*v*f-2*n0*pow(b1,2)*C1+6*n0*b1*q1*C1*po
        w(v,2)-n0*pow(b1,2)*pow(q1,2)*C1+10*n0*pow(b1,2)*pow(y,2)*
        pow(q1,2)*C1+4*K*pow(y,3)*pow(v,5))/(pow(v,2)*(N0*pow(y,3)+
        3*N0*y+n0*pow(y,2)+2*n0));
    }
    static void fVC3(double N0, double n0, double C1, double C2
        , double VC2, double VVC2, double RC2, double b1, double
        br, double q1, double q2, double av, double av1, double K,
        double v, double y, double r, double f, double * result )
    {
        *result=1/24.*(-12*n0*av1*pow(v,2)*C1-9*n0*C1*pow(v,4)-9
            6*N0*y*b1*q1*VC2*pow(v,2)-72*n0*b1*q1*VC2*pow(v,2)+24*n0*
            r*C1*pow(v,2)+4*n0*pow(b1,2)*pow(y,2)*C1-24*N0*y*C2*pow(v,3
            )+24*N0*y*r*C1*pow(v,2)+3*n0*pow(b1,2)*pow(q1,2)*C1*pow(y,
            4)+6*n0*b1*pow(y,2)*q1*C1*pow(v,2)-24*n0*C2*pow(v,3)+96*N0
            *y*b1*q1*C2*v+72*n0*b1*q1*C2*v+12*n0*b1*pow(y,2)*q1*C1*f-4
            8*N0*y*C2*v*f-12*n0*b1*q1*C1*f+48*n0*VC2*pow(v,2)*f+12*n0*
            C1*pow(f,2)+12*n0*C1*pow(v,2)*f-48*n0*C2*v*f+48*N0*y*VC2*po
            w(v,2)*f-24*n0*VC2*pow(v,4)-24*N0*y*VC2*pow(v,4)+2*n0*pow(
            b1,2)*C1+6*n0*b1*q1*C1*pow(v,2)+n0*pow(b1,2)*pow(q1,2)*C1-1
            0*n0*pow(b1,2)*pow(y,2)*pow(q1,2)*C1+48*N0*y*b1*q1*VVC2*po
            w(v,3)+48*n0*b1*q1*VVC2*pow(v,3)+12*K*pow(y,3)*pow(v,5))/(
            pow(v,3)*(N0*pow(y,3)+3*N0*y+n0*pow(y,2)+2*n0));
    }
    static void fRC3(double N0, double n0, double C1, double C2
        , double RC2, double b1, double br1, double q1, double q2,
        double K, double v, double y, double r, double f, double
        * result )
    {
        *result=-1/2.*(-4*n0*br1*q2*pow(v,2)*RC2+8*N0*y*b1*q1*RC
            2*v+6*n0*b1*q1*RC2*v-4*n0*C2*v+2*n0*C1*f-4*N0*y*C2*v-4*N0*
            y*RC2*v*f-2*N0*y*C1*pow(v,2)-4*N0*y*br1*q2*pow(v,2)*RC2+n0*
            br1*q2*v*C1-n0*b1*q1*C1-3*n0*C1*pow(v,2)+2*n0*RC2*pow(v,3)+
            n0*b1*pow(y,2)*q1*C1+2*N0*y*RC2*pow(v,3)-4*n0*RC2*v*f)/(po
            w(v,2)*(N0*pow(y,3)+3*N0*y+n0*pow(y,2)+2*n0));
    }
    static void fVVC3(double N0, double n0, double C1, double
        C2, double VC2, double VVC2, double b1, double q1, double
        av2, double K, double v, double y, double r, double f,
        double * result )
    {

```

```

*result=1/12.*(-9*n0*C1*pow(v,4)+96*N0*y*b1*q1*VC2*pow(
    v,2)+72*n0*b1*q1*VC2*pow(v,2)-12*n0*VVC2*pow(v,5)-4*n0*pow(
    b1,2)*pow(y,2)*C1-3*n0*pow(b1,2)*pow(q1,2)*C1*pow(y,4)-96*
    N0*y*b1*q1*C2*v-72*n0*b1*q1*C2*v-12*n0*b1*pow(y,2)*q1*C1*f+
    48*N0*y*C2*v*f+12*n0*b1*q1*C1*f+24*N0*y*VVC2*pow(v,3)*f-12
    *N0*y*VVC2*pow(v,5)-6*n0*av2*pow(v,3)*C1+24*n0*VVC2*pow(v,
    3)*f-48*n0*VC2*pow(v,2)*f-12*n0*C1*pow(f,2)+48*n0*C2*v*f-4
    8*N0*y*VC2*pow(v,2)*f-24*n0*VC2*pow(v,4)-24*N0*y*VC2*pow(
    v,4)-2*n0*pow(b1,2)*C1-n0*pow(b1,2)*pow(q1,2)*C1+10*n0*pow(
    b1,2)*pow(y,2)*pow(q1,2)*C1-48*N0*y*b1*q1*VVC2*pow(v,3)-36*
    n0*b1*q1*VVC2*pow(v,3)+12*K*pow(y,3)*pow(v,5))/(pow(v,4)*(
    N0*pow(y,3)+3*N0*y+n0*pow(y,2)+2*n0));
}

static void fVRC3(double N0, double n0, double C1, double
    C2, double RC2, double b1, double br1, double q1, double q2
    , double K, double v, double y, double r, double f,
    double * result )
{
    *result=-1/2.*(-4*n0*br1*q2*pow(v,2)*RC2+8*N0*y*b1*q1*RC
        2*v+6*n0*b1*q1*RC2*v-4*n0*C2*v+2*n0*C1*f-4*N0*y*C2*v-4*N0*
        y*RC2*v*f-2*N0*y*C1*pow(v,2)-4*N0*y*br1*q2*pow(v,2)*RC2+n0*
        br1*q2*v*C1-n0*b1*q1*C1-3*n0*C1*pow(v,2)+2*n0*RC2*pow(v,3)+
        n0*b1*pow(y,2)*q1*C1+2*N0*y*RC2*pow(v,3)-4*n0*RC2*v*f)/(po
        w(v,2)*(N0*pow(y,3)+3*N0*y+n0*pow(y,2)+2*n0));
}

static void fRRC3(double N0, double n0, double C1, double
    RC2, double br2, double q2, double K, double v, double y,
    double r, double f, double * result )
{
    *result=1/2.*(-2*n0*C1+8*N0*y*RC2*v+4*N0*y*br2*q2*pow(v,
        2)*RC2-n0*br2*q2*v*C1+4*n0*br2*q2*pow(v,2)*RC2+8*n0*RC2*v)
        /(pow(v,2)*(N0*pow(y,3)+3*N0*y+n0*pow(y,2)+2*n0));
}

static void fC4(double N0, double n0, double C1, double C2,
    double C3, double VC2, double VC3, double VVC2, double RC
    2, double RC3, double b1, double br, double br1, double q1
    , double q2, double av, double av1, double ar, double K,
    double v, double y, double r, double f, double * result )
{
    *result=-1/48.*(6*n0*pow(y,5)*pow(b1,2)*pow(q1,2)*C1*f+1
        2*N0*C2*pow(v,5)+2*K*pow(y,4)*pow(v,7)+12*n0*y*b1*q1*C1*po

```

$$\begin{aligned}
& w(v, 2) * f - 36 * n0 * y * b1 * q1 * C1 * \text{pow}(f, 2) + 24 * n0 * y * br * q2 * v * C1 * f + 96 * \\
& N0 * br * q2 * \text{pow}(v, 2) * RC2 * f - 144 * N0 * \text{pow}(y, 2) * \text{pow}(v, 2) * C3 * f + 42 * n0 \\
& * y * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * f - 10 * n0 * y * \text{pow}(b1, 2) * C1 * \text{pow}(v, 2) - \\
& 13 * n0 * \text{pow}(y, 5) * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 - 12 * n0 * y * C1 * \text{pow}(v, 2) * \\
& \text{pow}(f, 2) + 6 * n0 * y * C1 * \text{pow}(v, 4) * f - 44 * n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * \text{pow}(\\
& q1, 2) * C1 * f + 24 * n0 * y * av * v * C1 * f + 48 * N0 * ar * \text{pow}(v, 3) * RC2 - 24 * N0 * ar \\
& * \text{pow}(v, 2) * C1 + 8 * n0 * y * C1 * \text{pow}(f, 3) + 72 * N0 * C3 * \text{pow}(v, 4) - 28 * n0 * y * \\
& b1 * q1 * br * q2 * v * C1 + 24 * N0 * br * q2 * \text{pow}(v, 3) * C1 + 12 * n0 * \text{pow}(y, 3) * b1 * \\
& q1 * br * q2 * v * C1 - 12 * n0 * y * br * q2 * \text{pow}(v, 3) * C1 + 48 * N0 * br * q2 * \text{pow}(v, 2) \\
&) * C2 + 48 * N0 * br * \text{pow}(q2, 2) * \text{pow}(v, 3) * br1 * RC2 - 48 * n0 * y * b1 * q1 * br * \\
& q2 * \text{pow}(v, 2) * RC2 - 48 * N0 * br * q2 * \text{pow}(v, 4) * RC2 - 144 * N0 * b1 * q1 * br * q2 \\
& * \text{pow}(v, 2) * RC2 - 144 * N0 * br * q2 * \text{pow}(v, 3) * RC3 - 144 * N0 * \text{pow}(y, 2) * po \\
& w(v, 3) * br * q2 * RC3 - 144 * n0 * y * br * q2 * \text{pow}(v, 3) * RC3 + 8 * n0 * y * br * \text{pow}(\\
& q2, 2) * \text{pow}(v, 2) * br1 * C1 - 144 * N0 * b1 * q1 * VC3 * \text{pow}(v, 3) - 144 * N0 * \text{pow}(\\
& y, 2) * \text{pow}(v, 3) * b1 * q1 * VC3 - 144 * n0 * y * b1 * q1 * VC3 * \text{pow}(v, 3) + 48 * N0 * \\
& C2 * v * \text{pow}(f, 2) - 48 * N0 * C2 * \text{pow}(v, 3) * f - 144 * N0 * C3 * \text{pow}(v, 2) * f + 2 * n0 \\
& * y * \text{pow}(b1, 3) * q1 * C1 - 18 * n0 * \text{pow}(y, 3) * \text{pow}(b1, 3) * q1 * C1 - 12 * n0 * po \\
& w(y, 3) * b1 * q1 * C1 * \text{pow}(v, 2) * f + 48 * N0 * r * C1 * \text{pow}(v, 2) * f - 12 * n0 * y * po \\
& w(b1, 2) * C1 * f + 8 * n0 * y * b1 * q1 * \text{pow}(v, 2) * av1 * C1 + 432 * N0 * \text{pow}(y, 2) * \\
& \text{pow}(v, 2) * b1 * q1 * C3 + 288 * N0 * b1 * q1 * C3 * \text{pow}(v, 2) + 432 * n0 * y * b1 * q1 * \\
& C3 * \text{pow}(v, 2) + 48 * n0 * y * ar * \text{pow}(v, 3) * RC2 + 48 * N0 * \text{pow}(y, 2) * \text{pow}(v, 3) \\
& * ar * RC2 + 12 * n0 * \text{pow}(y, 3) * b1 * q1 * C1 * \text{pow}(f, 2) + 96 * N0 * b1 * q1 * VC2 * \\
& \text{pow}(v, 2) * f + 8 * n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * C1 * f - 24 * N0 * r * C1 * \text{pow}(v, 4) \\
& + 12 * N0 * \text{pow}(b1, 2) * C1 * \text{pow}(v, 2) - n0 * y * C1 * \text{pow}(v, 6) - 48 * N0 * C2 * po \\
& w(v, 3) * r + 72 * N0 * \text{pow}(b1, 2) * C2 * v - 48 * N0 * av * \text{pow}(v, 2) * C2 + 24 * N0 * av \\
& * \text{pow}(v, 3) * C1 + 48 * N0 * av * \text{pow}(v, 3) * VC2 + 24 * n0 * y * \text{pow}(b1, 2) * VVC2 * \\
& \text{pow}(v, 3) - 96 * n0 * y * \text{pow}(b1, 2) * VC2 * \text{pow}(v, 2) + 33 * n0 * \text{pow}(y, 3) * \text{pow}(\\
& b1, 3) * \text{pow}(q1, 3) * C1 + 10 * n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * po \\
& w(v, 2) + 48 * N0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * \text{pow}(v, 3) * VVC2 - 96 * n0 * y * av * \\
& \text{pow}(v, 2) * C2 - 12 * n0 * y * av * \text{pow}(v, 3) * C1 - 12 * n0 * y * b1 * q1 * av * v * C1 + 48 \\
& * n0 * y * av * \text{pow}(v, 3) * VC2 + 12 * n0 * \text{pow}(y, 3) * b1 * q1 * av * v * C1 + 48 * N0 * \\
& \text{pow}(y, 2) * \text{pow}(v, 3) * av * VC2 - 96 * N0 * \text{pow}(y, 2) * \text{pow}(v, 2) * av * C2 - 12 * \\
& n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C2 * v - 48 * n0 * y * C2 * \text{pow}(v, 3) * r - \\
& 9 * n0 * y * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 - n0 * y * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * \\
& \text{pow}(v, 2) + 3 * n0 * y * b1 * q1 * C1 * \text{pow}(v, 4) - 24 * n0 * y * b1 * q1 * \text{pow}(v, 2) * r * \\
& C1 - 192 * N0 * b1 * q1 * C2 * v * f - 48 * n0 * y * b1 * q1 * C2 * v * f - 144 * n0 * y * C3 * po \\
& w(v, 2) * f - 144 * N0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * VC2 * \text{pow}(v, 2) + 24 * N0 * \text{pow}(\\
& b1, 2) * VVC2 * \text{pow}(v, 3) - 48 * N0 * \text{pow}(b1, 2) * VC2 * \text{pow}(v, 2) - 3 * n0 * \text{pow}(\\
& y, 5) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * \text{pow}(v, 2) + n0 * \text{pow}(y, 7) * \text{pow}(b1, 3) * \\
& \text{pow}(q1, 3) * C1 - 4 * n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * C1 * \text{pow}(v, 2) + 4 * n0 * \text{pow}(\\
& y, 5) * \text{pow}(b1, 3) * q1 * C1 + 3 * n0 * \text{pow}(y, 3) * b1 * q1 * C1 * \text{pow}(v, 4) + 24 * N0 *
\end{aligned}$$

```

    pow(y,2)*pow(v,3)*pow(b1,2)*VVC2-96*N0*pow(y,2)*pow(v,2)*po
    w(b1,2)*VC2-48*N0*b1*q1*VC2*pow(v,4)-48*n0*y*pow(b1,2)*pow(
    q1,2)*VC2*pow(v,2)+128*n0*y*pow(b1,2)*C2*v-48*N0*pow(y,2)*
    pow(v,3)*C2*r+144*N0*pow(y,2)*v*pow(b1,2)*C2+24*n0*y*b1*q1*
    C2*pow(v,3)+100*n0*y*pow(b1,2)*pow(q1,2)*C2*v+72*n0*y*C3*po
    w(v,4)+48*N0*b1*q1*C2*pow(v,3)+192*N0*pow(b1,2)*pow(q1,2)*
    C2*v+72*N0*pow(y,2)*pow(v,4)*C3)/(pow(v,3)*(3*N0+N0*pow(y,4
    )+6*N0*pow(y,2)+5*n0*y+n0*pow(y,3)));
}
static void fRC4(double N0, double n0, double C1, double C2
, double C3, double VC2, double VC3, double VVC2, double
RC2, double RC3, double b1, double br, double br1, double
q1, double q2, double av, double av1, double ar, double K,
double v, double y, double r, double f, double * result )
{
    *result=-1/48.*(6*n0*pow(y,5)*pow(b1,2)*pow(q1,2)*C1*f+1
    2*N0*C2*pow(v,5)+2*K*pow(y,4)*pow(v,7)+12*n0*y*b1*q1*C1*po
    w(v,2)*f-36*n0*y*b1*q1*C1*pow(f,2)+24*n0*y*br*q2*v*C1*f+96*
    N0*br*q2*pow(v,2)*RC2*f-144*N0*pow(y,2)*pow(v,2)*C3*f+42*n0
    *y*pow(b1,2)*pow(q1,2)*C1*f-10*n0*y*pow(b1,2)*C1*pow(v,2)-
    13*n0*pow(y,5)*pow(b1,3)*pow(q1,3)*C1-12*n0*y*C1*pow(v,2)*
    pow(f,2)+6*n0*y*C1*pow(v,4)*f-44*n0*pow(y,3)*pow(b1,2)*pow(
    q1,2)*C1*f+24*n0*y*av*v*C1*f+48*N0*ar*pow(v,3)*RC2-24*N0*ar
    *pow(v,2)*C1+8*n0*y*C1*pow(f,3)+72*N0*C3*pow(v,4)-28*n0*y*
    b1*q1*br*q2*v*C1+24*N0*br*q2*pow(v,3)*C1+12*n0*pow(y,3)*b1*
    q1*br*q2*v*C1-12*n0*y*br*q2*pow(v,3)*C1+48*N0*br*q2*pow(v,2
    )*C2+48*N0*br*pow(q2,2)*pow(v,3)*br1*RC2-48*n0*y*b1*q1*br*
    q2*pow(v,2)*RC2-48*N0*br*q2*pow(v,4)*RC2-144*N0*b1*q1*br*q2
    *pow(v,2)*RC2-144*N0*br*q2*pow(v,3)*RC3-144*N0*pow(y,2)*po
    w(v,3)*br*q2*RC3-144*n0*y*br*q2*pow(v,3)*RC3+8*n0*y*br*pow(
    q2,2)*pow(v,2)*br1*C1-144*N0*b1*q1*VC3*pow(v,3)-144*N0*pow(
    y,2)*pow(v,3)*b1*q1*VC3-144*n0*y*b1*q1*VC3*pow(v,3)+48*N0*
    C2*v*pow(f,2)-48*N0*C2*pow(v,3)*f-144*N0*C3*pow(v,2)*f+2*n0
    *y*pow(b1,3)*q1*C1-18*n0*pow(y,3)*pow(b1,3)*q1*C1-12*n0*po
    w(y,3)*b1*q1*C1*pow(v,2)*f+48*N0*r*C1*pow(v,2)*f-12*n0*y*po
    w(b1,2)*C1*f+8*n0*y*b1*q1*pow(v,2)*av1*C1+432*N0*pow(y,2)*
    pow(v,2)*b1*q1*C3+288*N0*b1*q1*C3*pow(v,2)+432*n0*y*b1*q1*
    C3*pow(v,2)+48*n0*y*ar*pow(v,3)*RC2+48*N0*pow(y,2)*pow(v,3)
    *ar*RC2+12*n0*pow(y,3)*b1*q1*C1*pow(f,2)+96*N0*b1*q1*VC2*
    pow(v,2)*f+8*n0*pow(y,3)*pow(b1,2)*C1*f-24*N0*r*C1*pow(v,4)
    +12*N0*pow(b1,2)*C1*pow(v,2)-n0*y*C1*pow(v,6)-48*N0*C2*po

```

```

w(v,3)*r+72*N0*pow(b1,2)*C2*v-48*N0*av*pow(v,2)*C2+24*N0*av
*pow(v,3)*C1+48*N0*av*pow(v,3)*VC2+24*n0*y*pow(b1,2)*VVC2*
pow(v,3)-96*n0*y*pow(b1,2)*VC2*pow(v,2)+33*n0*pow(y,3)*pow(
b1,3)*pow(q1,3)*C1+10*n0*pow(y,3)*pow(b1,2)*pow(q1,2)*C1*po
w(v,2)+48*N0*pow(b1,2)*pow(q1,2)*pow(v,3)*VVC2-96*n0*y*av*
pow(v,2)*C2-12*n0*y*av*pow(v,3)*C1-12*n0*y*b1*q1*av*v*C1+48
*n0*y*av*pow(v,3)*VC2+12*n0*pow(y,3)*b1*q1*av*v*C1+48*N0*
pow(y,2)*pow(v,3)*av*VC2-96*N0*pow(y,2)*pow(v,2)*av*C2-12*
n0*pow(y,3)*pow(b1,2)*pow(q1,2)*C2*v-48*n0*y*C2*pow(v,3)*r-
9*n0*y*pow(b1,3)*pow(q1,3)*C1-n0*y*pow(b1,2)*pow(q1,2)*C1*
pow(v,2)+3*n0*y*b1*q1*C1*pow(v,4)-24*n0*y*b1*q1*pow(v,2)*r*
C1-192*N0*b1*q1*C2*v*f-48*n0*y*b1*q1*C2*v*f-144*n0*y*C3*po
w(v,2)*f-144*N0*pow(b1,2)*pow(q1,2)*VC2*pow(v,2)+24*N0*pow(
b1,2)*VVC2*pow(v,3)-48*N0*pow(b1,2)*VC2*pow(v,2)-3*n0*pow(
y,5)*pow(b1,2)*pow(q1,2)*C1*pow(v,2)+n0*pow(y,7)*pow(b1,3)*
pow(q1,3)*C1-4*n0*pow(y,3)*pow(b1,2)*C1*pow(v,2)+4*n0*pow(
y,5)*pow(b1,3)*q1*C1+3*n0*pow(y,3)*b1*q1*C1*pow(v,4)+24*N0*
pow(y,2)*pow(v,3)*pow(b1,2)*VVC2-96*N0*pow(y,2)*pow(v,2)*po
w(b1,2)*VC2-48*N0*b1*q1*VC2*pow(v,4)-48*n0*y*pow(b1,2)*pow(
q1,2)*VC2*pow(v,2)+128*n0*y*pow(b1,2)*C2*v-48*N0*pow(y,2)*
pow(v,3)*C2*r+144*N0*pow(y,2)*v*pow(b1,2)*C2+24*n0*y*b1*q1*
C2*pow(v,3)+100*n0*y*pow(b1,2)*pow(q1,2)*C2*v+72*n0*y*C3*po
w(v,4)+48*N0*b1*q1*C2*pow(v,3)+192*N0*pow(b1,2)*pow(q1,2)*
C2*v+72*N0*pow(y,2)*pow(v,4)*C3)/(pow(v,3)*(3*N0+N0*pow(y,4)
)+6*N0*pow(y,2)+5*n0*y+n0*pow(y,3)));
}
static void fVC4(double N0, double n0, double C1, double C2
, double C3, double VC2, double VC3, double VVC2, double
VVC3, double VRC3, double RC2, double RC3, double b1,
double br, double br1, double q1, double q2, double av, double
av1, double av2, double ar, double K, double v, double y,
double r, double f, double * result )
{
*result=1/24.*(-12*n0*pow(y,3)*pow(b1,2)*pow(q1,2)*C2*v-
216*n0*y*b1*q1*VC3*pow(v,3)-72*N0*C3*pow(v,2)*f+8*n0*y*C1*
pow(f,3)-24*N0*VC2*pow(v,2)*pow(f,2)+24*N0*VC2*pow(v,4)*f+7
2*N0*VC3*pow(v,3)*f-48*N0*b1*q1*VVC2*pow(v,3)*f+72*n0*y*VC
3*pow(v,3)*f-74*n0*y*pow(b1,2)*pow(q1,2)*VC2*pow(v,2)-60*
N0*pow(b1,2)*VC2*pow(v,2)+6*n0*pow(y,3)*b1*q1*br*q2*v*C1-24
*n0*y*b1*q1*br*q2*pow(v,2)*RC2+24*N0*br*q2*pow(v,2)*C2-192
*N0*b1*q1*C2*v*f+12*n0*y*av*v*C1*f-48*n0*y*b1*q1*C2*v*f+8*

```


$$\begin{aligned}
& n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * C1 * f - 72 * N0 * \text{pow}(y, 2) * \text{pow}(v, 2) * C3 * f - 36 * \\
& n0 * y * b1 * q1 * C1 * \text{pow}(f, 2) + 42 * n0 * y * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * f - 72 * \\
& n0 * y * C3 * \text{pow}(v, 2) * f - 36 * N0 * VC3 * \text{pow}(v, 5) - 48 * N0 * \text{pow}(y, 2) * \text{pow}(v, \\
& 2) * av * C2 + 48 * n0 * y * av * \text{pow}(v, 3) * VC2 - 12 * N0 * C2 * \text{pow}(v, 5) - 18 * n0 * \\
& \text{pow}(y, 3) * \text{pow}(b1, 3) * q1 * C1 + 6 * n0 * y * av * \text{pow}(v, 3) * C1 + 6 * n0 * \text{pow}(y, 3) \\
&) * b1 * q1 * av * v * C1 + 144 * N0 * \text{pow}(y, 2) * v * \text{pow}(b1, 2) * C2 - 144 * N0 * b1 * \\
& q1 * VC3 * \text{pow}(v, 3) - 13 * n0 * \text{pow}(y, 5) * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 + 33 * n0 \\
& * \text{pow}(y, 3) * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 + 144 * N0 * b1 * q1 * VC2 * \text{pow}(v, 2) \\
& * f + n0 * \text{pow}(y, 7) * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 + 4 * n0 * \text{pow}(y, 5) * \text{pow}(b1 \\
& , 3) * q1 * C1 - 6 * n0 * y * b1 * q1 * av * v * C1 - 12 * n0 * y * \text{pow}(b1, 2) * C1 * f + 12 * \\
& n0 * y * br * q2 * v * C1 * f + 24 * N0 * VC2 * \text{pow}(v, 4) * r - 12 * N0 * av1 * \text{pow}(v, 4) * \\
& C1 - 36 * n0 * y * VC3 * \text{pow}(v, 5) - 36 * N0 * \text{pow}(y, 2) * \text{pow}(v, 5) * VC3 - 24 * N0 * \\
& av1 * \text{pow}(v, 4) * VC2 + 24 * N0 * av1 * \text{pow}(v, 3) * C2 - 24 * N0 * \text{pow}(y, 2) * \text{pow}(\\
& v, 4) * av * VVC2 - 24 * n0 * y * av * \text{pow}(v, 4) * VVC2 - 36 * n0 * y * C3 * \text{pow}(v, 4) + 1 \\
& 00 * n0 * y * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C2 * v + 72 * N0 * \text{pow}(b1, 2) * C2 * v + 72 * \\
& N0 * br * q2 * \text{pow}(v, 4) * VRC3 + 24 * n0 * y * \text{pow}(b1, 2) * \text{pow}(q1, 2) * VVC2 * po \\
& w(v, 3) + 72 * n0 * y * b1 * q1 * VVC3 * \text{pow}(v, 4) + 24 * N0 * b1 * q1 * VVC2 * \text{pow}(v, 5) \\
&) + 24 * n0 * y * VC2 * \text{pow}(v, 4) * r + 24 * N0 * \text{pow}(y, 2) * \text{pow}(v, 4) * VC2 * r - 4 * \\
& n0 * y * b1 * q1 * \text{pow}(v, 3) * av2 * C1 + 6 * n0 * \text{pow}(y, 3) * \text{pow}(b1, 2) * \text{pow}(q1, 2) \\
&) * VC2 * \text{pow}(v, 2) - 12 * n0 * y * b1 * q1 * VC2 * \text{pow}(v, 4) + 48 * n0 * y * av1 * \text{pow}(\\
& v, 3) * C2 + 48 * N0 * \text{pow}(y, 2) * \text{pow}(v, 3) * av1 * C2 - 6 * n0 * \text{pow}(y, 3) * b1 * q1 * \\
& av1 * \text{pow}(v, 2) * C1 + 72 * N0 * b1 * q1 * VVC3 * \text{pow}(v, 4) + 2 * n0 * y * \text{pow}(b1, 3) * \\
& q1 * C1 + 72 * N0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * \text{pow}(v, 3) * VVC2 + 6 * n0 * y * b1 * q1 * \\
& \text{pow}(v, 2) * av1 * C1 + 216 * n0 * y * b1 * q1 * C3 * \text{pow}(v, 2) - 36 * N0 * \text{pow}(y, 2) * \\
& \text{pow}(v, 4) * C3 + 6 * n0 * y * av1 * \text{pow}(v, 4) * C1 - 24 * n0 * y * av1 * \text{pow}(v, 4) * VC2 \\
& - 24 * N0 * br * q2 * \text{pow}(v, 3) * VC2 + 144 * N0 * b1 * q1 * C3 * \text{pow}(v, 2) + 216 * N0 * \\
& \text{pow}(y, 2) * \text{pow}(v, 2) * b1 * q1 * C3 - 168 * N0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * VC2 * \\
& \text{pow}(v, 2) - 48 * n0 * y * av * \text{pow}(v, 2) * C2 + 24 * N0 * av * \text{pow}(v, 3) * VC2 - 24 * N0 \\
& * av * \text{pow}(v, 2) * C2 - 4 * K * \text{pow}(y, 4) * \text{pow}(v, 7) + 24 * N0 * r * C1 * \text{pow}(v, 4) - \\
& 36 * N0 * C3 * \text{pow}(v, 4) + 24 * N0 * \text{pow}(b1, 2) * VVC2 * \text{pow}(v, 3) + 2 * n0 * y * C1 * \\
& \text{pow}(v, 6) + 192 * N0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C2 * v + 128 * n0 * y * \text{pow}(b1, 2) \\
& * C2 * v - 12 * N0 * av * \text{pow}(v, 3) * C1 + 48 * N0 * \text{pow}(y, 2) * \text{pow}(v, 3) * av * VC2 - \\
& 112 * n0 * y * \text{pow}(b1, 2) * VC2 * \text{pow}(v, 2) - 120 * N0 * \text{pow}(y, 2) * \text{pow}(v, 2) * \\
& \text{pow}(b1, 2) * VC2 + 24 * n0 * y * b1 * q1 * VC2 * \text{pow}(v, 2) * f + 72 * N0 * \text{pow}(y, 2) * \\
& \text{pow}(v, 3) * VC3 * f - 12 * n0 * y * av1 * \text{pow}(v, 2) * C1 * f - 3 * n0 * \text{pow}(y, 3) * b1 * \\
& q1 * C1 * \text{pow}(v, 4) + 6 * n0 * y * br * q2 * \text{pow}(v, 3) * C1 - 72 * N0 * b1 * q1 * br * q2 * \\
& \text{pow}(v, 2) * RC2 - 12 * N0 * br * q2 * \text{pow}(v, 3) * C1 + 6 * n0 * \text{pow}(y, 5) * \text{pow}(b1, 2) \\
&) * \text{pow}(q1, 2) * C1 * f - 6 * n0 * y * C1 * \text{pow}(v, 4) * f + 12 * n0 * \text{pow}(y, 3) * b1 * q1 \\
& * C1 * \text{pow}(f, 2) + 48 * N0 * br * q2 * \text{pow}(v, 2) * RC2 * f - 44 * n0 * \text{pow}(y, 3) * po \\
& w(b1, 2) * \text{pow}(q1, 2) * C1 * f - 216 * N0 * \text{pow}(y, 2) * \text{pow}(v, 3) * b1 * q1 * VC3 + 2 \\
& 4 * N0 * br * q2 * \text{pow}(v, 4) * RC2 - 24 * N0 * \text{pow}(y, 2) * \text{pow}(v, 4) * av1 * VC2 + 48
\end{aligned}$$

```

    *n0*y*pow(b1,2)*VVC2*pow(v,3)-6*N0*VC2*pow(v,6)+72*N0*pow(
    y,2)*pow(v,4)*b1*q1*VVC3+72*N0*pow(y,2)*pow(v,4)*br*q2*VRC3
    +72*n0*y*br*q2*pow(v,4)*VRC3+48*N0*C2*v*pow(f,2)-24*N0*av*
    pow(v,4)*VVC2-14*n0*y*b1*q1*br*q2*v*C1-3*n0*y*b1*q1*C1*pow(
    v,4)-9*n0*y*pow(b1,3)*pow(q1,3)*C1+48*N0*pow(y,2)*pow(v,3)*
    pow(b1,2)*VVC2)/(pow(v,4)*(N0*pow(y,4)+6*N0*pow(y,2)+3*N0+5
    *n0*y+n0*pow(y,3)));
}
static void fc5(double N0, double n0, double C1, double C2,
    double C3, double C4, double VC2, double VC3, double VC4,
    double VVC2, double VVC3, double VRC3, double RC2,
    double RC3, double RC4, double RRC3, double b1, double br,
    double br1, double br2, double q1, double q2, double q3, double
    av, double av1, double av2, double ar, double ar1, double
    K, double v, double y, double r, double f, double * result)
{
    *result=1/5760.*(-5760*N0*pow(y,3)*pow(v,4)*q3*br*b1*VR
    C3+86400*N0*y*b1*q1*pow(v,3)*br*q2*RC3-17280*n0*br*q2*pow(
    v,3)*C3-360*n0*pow(y,2)*pow(v,5)*br*q2*C1+16320*n0*br*q2*po
    w(v,2)*pow(b1,2)*RC2-11520*N0*y*br*q2*pow(v,4)*RC2*r+5760*
    N0*y*q3*br*b1*pow(v,3)*VC2-2880*n0*pow(v,5)*br*pow(q2,2)*br
    1*RC2-17280*n0*pow(br,2)*pow(q2,2)*pow(v,4)*RRC3-960*n0*br
    *q2*pow(v,3)*ar1*C1+1440*n0*pow(y,2)*br*q2*v*pow(b1,2)*C1-
    13440*n0*pow(y,2)*pow(b1,2)*pow(q1,2)*br*q2*pow(v,2)*RC2+7
    20*n0*pow(y,2)*pow(b1,2)*q1*q3*br*v*C1-11520*n0*br*q2*pow(
    v,4)*RC2*r+23040*N0*y*br*q2*pow(v,2)*pow(b1,2)*RC2+1920*n0*
    pow(y,2)*br*q2*pow(v,2)*pow(b1,2)*RC2-240*n0*pow(y,2)*pow(
    br,2)*pow(q2,3)*pow(v,3)*br2*C1+1920*n0*pow(br,2)*pow(q2,3)
    *pow(v,4)*br2*RC2+23040*N0*pow(y,3)*pow(v,4)*br*q2*RC4+691
    20*N0*y*br*q2*pow(v,4)*RC4-1080*n0*pow(v,5)*br*q2*C1+1440*
    n0*b1*q1*ar*pow(v,2)*C1-120*n0*pow(v,2)*pow(b1,2)*pow(q1,2)
    *C1*f+240*n0*pow(b1,3)*q1*C1*f-17280*N0*y*pow(v,2)*C3*pow(
    f,2)+2160*n0*pow(y,2)*pow(b1,2)*C1*pow(f,2)+5760*n0*b1*q1*
    VC2*pow(v,2)*pow(f,2)-120*n0*pow(b1,4)*pow(q1,2)*pow(y,8)*
    C1-15*n0*pow(y,2)*pow(v,8)*C1-840*n0*pow(b1,3)*pow(y,4)*po
    w(v,2)*q1*C1+390*n0*pow(b1,2)*pow(y,4)*pow(v,4)*pow(q1,2)*
    C1-3771*n0*pow(y,2)*pow(b1,4)*pow(q1,4)*C1-1440*n0*pow(y,2)
    *pow(v,4)*b1*q1*r*C1+2400*n0*pow(b1,3)*pow(y,4)*pow(v,2)*
    pow(q1,3)*C1-177*n0*pow(b1,4)*pow(q1,4)*C1+5760*N0*y*pow(v,
    3)*pow(b1,2)*VVC2*f-6720*n0*pow(y,2)*pow(b1,3)*q1*C1*f-117
    60*n0*pow(b1,3)*pow(y,4)*pow(q1,3)*C1*f+12960*n0*pow(y,2)*

```

```

pow(b1,3)*pow(q1,3)*C1*f+240*n0*C1*pow(f,4)+150*n0*pow(v,4)
*pow(b1,2)*pow(q1,2)*C1+60*n0*pow(b1,3)*pow(q1,3)*pow(y,8)
*pow(v,2)*C1+17280*n0*pow(b1,2)*VC3*pow(v,3)-480*n0*b1*po
w(y,4)*q1*C1*pow(f,3)+5760*n0*pow(v,3)*pow(b1,2)*VVC2*f-230
40*N0*y*pow(v,2)*av*C2*f-34560*n0*pow(v,3)*br*q2*RC3*f-345
60*N0*y*pow(v,3)*br*q2*RC3*f-960*n0*av*pow(v,3)*av1*C1+288
0*n0*pow(y,2)*b1*q1*pow(v,2)*av*C2-720*n0*pow(v,5)*b1*q1*
C2-11520*n0*av*pow(v,4)*VC3-5760*n0*pow(v,5)*av*VC2+2880*n0
*pow(v,4)*av*C2-1080*n0*pow(v,5)*av*C1+17280*n0*av*pow(v,3
)*C3-240*n0*pow(y,2)*C1*pow(f,4)+1920*n0*C2*v*pow(f,3)+172
80*n0*pow(v,4)*C3*f-17280*n0*pow(v,2)*C3*pow(f,2)+2880*n0*
pow(y,2)*b1*q1*C1*pow(f,3)+360*n0*pow(v,4)*b1*q1*C1*f+360*
n0*pow(b1,2)*pow(y,6)*pow(v,2)*pow(q1,2)*C1*f-240*n0*pow(y,
2)*pow(b1,2)*C1*pow(v,2)*f-23040*N0*y*b1*q1*pow(v,3)*ar*RC
2+1440*n0*pow(v,3)*br*q2*C1*f-1440*n0*pow(y,2)*av*v*C1*po
w(f,2)-1440*n0*b1*pow(y,4)*q1*av*v*C1*f-23040*n0*b1*q1*br*
q2*pow(v,2)*RC2*f+5760*n0*pow(y,2)*b1*q1*br*q2*pow(v,2)*RC2
*f+7680*n0*pow(y,2)*b1*q1*br*q2*v*C1*f+24000*n0*v*pow(b1,2
)*C2*f-120*n0*pow(b1,3)*pow(q1,3)*pow(y,8)*C1*f-7200*n0*po
w(v,3)*pow(b1,2)*C2-14400*n0*b1*q1*C2*v*pow(f,2)-11520*N0*
y*pow(v,3)*C2*r*f-3360*n0*b1*q1*br*q2*v*C1*f-960*n0*pow(y,2
)*pow(v,3)*b1*q1*br*q2*C1-17280*N0*y*br*q2*pow(v,3)*C3-115
20*n0*q3*br*b1*VRC3*pow(v,4)+5760*n0*b1*q1*br*q2*pow(v,3)*
VC2+1900*n0*pow(b1,4)*pow(y,6)*pow(q1,2)*C1-3530*n0*pow(b1,
4)*pow(y,6)*pow(q1,4)*C1-840*n0*pow(b1,3)*pow(y,6)*pow(v,2
)*pow(q1,3)*C1+435*n0*pow(b1,4)*pow(q1,4)*pow(y,8)*C1+960*
n0*b1*q1*pow(v,2)*av1*C1*f-960*n0*pow(y,2)*b1*q1*pow(v,2)*
av1*C1*f+120*n0*pow(y,2)*pow(v,6)*C1*f-360*n0*pow(y,2)*pow(
v,4)*C1*pow(f,2)-207360*N0*y*b1*q1*C4*pow(v,3)-960*n0*ar*po
w(v,3)*br1*q2*C1+720*n0*b1*pow(y,4)*pow(v,3)*q1*br*q2*C1+28
80*n0*pow(v,3)*br*q2*C1*r-480*n0*pow(b1,2)*pow(y,4)*br*q2*
v*C1+480*n0*pow(y,2)*pow(v,4)*br*pow(q2,2)*br1*C1+240*n0*br
*pow(q2,3)*pow(v,3)*pow(br1,2)*C1+1920*n0*br*pow(q2,2)*po
w(v,3)*br1*C2-360*n0*pow(b1,2)*pow(y,6)*pow(q1,2)*br*q2*v*
C1+240*n0*pow(br,2)*pow(q2,3)*pow(v,3)*br2*C1-5760*n0*pow(
y,2)*pow(v,4)*q3*br*b1*VRC3-360*n0*b1*pow(y,4)*pow(v,4)*q1*
C1*f+5760*n0*pow(y,2)*b1*q1*av*v*C1*f-2880*n0*ar*pow(v,2)*
C1*f-1440*n0*b1*pow(y,4)*q1*br*q2*v*C1*f+15*n0*pow(v,8)*C1-
5760*n0*pow(v,4)*br*q2*RC2*f-1440*n0*pow(y,2)*br*q2*v*C1*
pow(f,2)+5760*n0*pow(y,2)*pow(v,4)*C3*r+720*n0*pow(v,2)*b1*
q1*C1*pow(f,2)-17280*n0*pow(v,2)*av*C2*f-1440*n0*b1*q1*av*

```

$$\begin{aligned}
& v * C1 * f + 11520 * n0 * \text{pow}(v, 3) * av * VC2 * f + 1440 * n0 * \text{pow}(y, 2) * av * \text{pow}(\\
& v, 3) * C1 * f - 240 * n0 * \text{pow}(v, 7) * C2 - 5760 * N0 * \text{pow}(y, 3) * \text{pow}(v, 4) * av * \\
& VC3 + 480 * n0 * av * v * \text{pow}(b1, 2) * C1 - 360 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 6) * \text{pow}(\\
& q1, 2) * av * v * C1 + 11520 * N0 * y * b1 * q1 * \text{pow}(v, 4) * av * VVC2 + 2880 * n0 * av * \\
& \text{pow}(v, 3) * r * C1 + 720 * n0 * \text{pow}(v, 3) * b1 * q1 * av * C1 + 5760 * N0 * y * ar * \text{pow}(\\
& v, 4) * br1 * q2 * RC2 + 480 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 4) * \text{pow}(v, 2) * C1 * f - 480 \\
& * n0 * \text{pow}(b1, 3) * \text{pow}(y, 6) * q1 * C1 * f - 120 * n0 * \text{pow}(v, 6) * C1 * f - 23040 * \\
& n0 * \text{pow}(v, 5) * C4 - 4320 * n0 * \text{pow}(v, 6) * C3 + 60 * n0 * b1 * \text{pow}(y, 4) * \text{pow}(v, \\
& 6) * q1 * C1 - 2880 * N0 * \text{pow}(y, 3) * \text{pow}(v, 4) * \text{pow}(b1, 2) * VVC3 + 240 * n0 * \\
& \text{pow}(b1, 3) * \text{pow}(y, 6) * \text{pow}(v, 2) * q1 * C1 - 90 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 6) * \\
& \text{pow}(v, 4) * \text{pow}(q1, 2) * C1 - 15 * n0 * \text{pow}(b1, 4) * \text{pow}(q1, 4) * C1 * \text{pow}(y, 10) \\
& - 2880 * n0 * \text{pow}(r, 2) * \text{pow}(v, 4) * C1 + 720 * n0 * \text{pow}(v, 6) * r * C1 + 1440 * \\
& n0 * br * q2 * v * C1 * \text{pow}(f, 2) - 5760 * n0 * q3 * br * b1 * \text{pow}(v, 2) * RC2 * f + 5760 \\
& * n0 * br * q2 * \text{pow}(v, 2) * RC2 * \text{pow}(f, 2) + 720 * n0 * b1 * \text{pow}(y, 4) * q1 * C1 * \\
& \text{pow}(v, 2) * \text{pow}(f, 2) - 92160 * N0 * \text{pow}(y, 3) * \text{pow}(v, 3) * b1 * q1 * C4 - 23040 \\
& * N0 * y * \text{pow}(v, 2) * \text{pow}(b1, 2) * VC2 * f + 69120 * N0 * y * C4 * \text{pow}(v, 3) * f + 23 \\
& 040 * n0 * \text{pow}(y, 2) * \text{pow}(v, 3) * C4 * f - 11520 * n0 * \text{pow}(v, 3) * C2 * r * f - 163 \\
& 20 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C2 * v * f - 720 * n0 * \text{pow}(b1, 2) \\
& * C1 * \text{pow}(f, 2) + 1440 * n0 * C2 * \text{pow}(v, 5) * f - 2880 * n0 * \text{pow}(v, 3) * C2 * po \\
& w(f, 2) + 360 * n0 * \text{pow}(v, 4) * C1 * \text{pow}(f, 2) - 480 * n0 * C1 * \text{pow}(v, 2) * \text{pow}(\\
& f, 3) + 46080 * n0 * C4 * \text{pow}(v, 3) * f - 480 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 4) * C1 * po \\
& w(f, 2) + 2520 * n0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * \text{pow}(f, 2) + 240 * n0 * \text{pow}(\\
& b1, 2) * \text{pow}(q1, 2) * \text{pow}(v, 3) * av2 * C1 - 4320 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 2) * \\
& \text{pow}(q1, 2) * \text{pow}(v, 2) * C3 - 720 * n0 * \text{pow}(b1, 2) * q1 * q3 * br * v * C1 - 17280 * \\
& N0 * y * br * q2 * \text{pow}(v, 3) * av * RC2 - 2880 * n0 * q3 * br * b1 * \text{pow}(v, 4) * RC2 + 57 \\
& 60 * N0 * y * \text{pow}(br, 2) * \text{pow}(v, 3) * RC2 - 17280 * N0 * y * q3 * br * b1 * VRC3 * po \\
& w(v, 4) + 5760 * n0 * q3 * br * b1 * \text{pow}(v, 3) * VC2 + 2880 * n0 * \text{pow}(y, 2) * b1 * q1 \\
& * br * q2 * \text{pow}(v, 2) * C2 - 12480 * n0 * b1 * q1 * br * q2 * \text{pow}(v, 2) * C2 + 2880 * \\
& n0 * \text{pow}(y, 2) * b1 * q1 * br * \text{pow}(q2, 2) * \text{pow}(v, 3) * br1 * RC2 - 17280 * N0 * y * \\
& \text{pow}(br, 2) * \text{pow}(q2, 2) * \text{pow}(v, 4) * RRC3 + 5760 * N0 * y * br * q2 * \text{pow}(v, 4) * \\
& ar1 * RC2 - 2880 * N0 * y * q3 * br * b1 * \text{pow}(v, 4) * RC2 + 11520 * N0 * y * q3 * br * po \\
& w(b1, 2) * \text{pow}(v, 2) * q1 * RC2 + 1320 * n0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * br * q2 * \\
& v * C1 + 48 * K * \text{pow}(y, 5) * \text{pow}(v, 9) - 120 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 4) * \text{pow}(\\
& v, 4) * C1 + 17280 * n0 * \text{pow}(v, 5) * b1 * q1 * VC3 + 5760 * N0 * \text{pow}(y, 3) * \text{pow}(v, \\
& 4) * C3 * r + 11520 * N0 * y * \text{pow}(v, 3) * ar * RC2 * f + 4440 * n0 * \text{pow}(b1, 2) * po \\
& w(y, 4) * \text{pow}(q1, 2) * C1 * \text{pow}(f, 2) - 34560 * N0 * y * \text{pow}(v, 3) * b1 * q1 * VC3 * \\
& f - 1080 * n0 * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 * f + 34560 * N0 * y * v * \text{pow}(b1, 2) * \\
& C2 * f - 28800 * N0 * y * \text{pow}(b1, 3) * q1 * \text{pow}(v, 3) * VVC2 - 8640 * n0 * \text{pow}(b1, 3) \\
&) * \text{pow}(q1, 3) * \text{pow}(v, 3) * VVC2 + 5760 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 2) * \text{pow}(\\
& q1, 2) * VC2 * \text{pow}(v, 2) * f + 23040 * N0 * \text{pow}(y, 3) * \text{pow}(v, 3) * C4 * f + 86400 * \\
& n0 * \text{pow}(v, 2) * b1 * q1 * C3 * f + 1440 * n0 * \text{pow}(y, 2) * \text{pow}(v, 3) * br * q2 * C1 *
\end{aligned}$$

$$\begin{aligned}
& f-2400*n0*pow(y,2)*pow(v,2)*pow(b1,2)*pow(q1,2)*C1*r-720*n0 \\
& *pow(y,2)*pow(av,2)*pow(v,2)*C1-120*n0*pow(b1,2)*pow(q1,2) \\
& *av*v*C1+720*n0*pow(y,2)*pow(v,5)*b1*q1*C2+240*n0*pow(b1,3) \\
&)*pow(y,6)*pow(q1,3)*C2*v+17280*N0*y*pow(v,4)*C3*r-155520* \\
& N0*y*pow(b1,2)*pow(q1,2)*pow(v,2)*C3-8640*N0*y*pow(b1,2)*VV \\
& C3*pow(v,4)+23040*n0*pow(y,2)*pow(v,4)*b1*q1*VC4+720*n0*po \\
& w(y,2)*pow(v,4)*b1*q1*C1*f+2880*n0*pow(y,2)*pow(v,2)*b1*q1* \\
& C1*r*f-2880*n0*pow(y,2)*pow(v,2)*b1*q1*C1*pow(f,2)+3720*n0* \\
& pow(y,2)*pow(v,2)*pow(b1,2)*pow(q1,2)*C1*f-420*n0*pow(y,2)* \\
& pow(v,4)*pow(b1,2)*C1+2880*n0*pow(y,2)*pow(b1,3)*pow(q1,3)* \\
& pow(v,3)*VVC2-360*n0*pow(b1,2)*pow(y,6)*pow(q1,2)*C1*pow(f, \\
& 2)+11520*N0*y*pow(v,3)*av*VC2*f+1440*n0*pow(b1,2)*pow(y,4) \\
& *pow(q1,2)*C2*v*f+8640*n0*pow(v,3)*b1*q1*C2*f-34560*N0*y* \\
& pow(v,4)*b1*q1*C3-60480*N0*y*pow(b1,2)*pow(v,2)*C3-25920*n0 \\
& *pow(v,4)*b1*q1*C3+1440*n0*pow(v,6)*b1*q1*VC2-2880*n0*pow(\\
& y,2)*pow(v,3)*b1*q1*C2*f+5760*n0*br*q2*pow(v,2)*C2*f+1920* \\
& n0*pow(y,2)*v*pow(b1,2)*C2*f-5760*N0*y*pow(v,5)*ar*RC2+5760 \\
& *n0*ar*pow(v,3)*C2+480*n0*pow(b1,2)*pow(v,2)*r*C1-900*n0* \\
& pow(v,4)*pow(b1,2)*C1-13440*n0*pow(y,2)*pow(b1,3)*pow(q1,3) \\
& *VC2*pow(v,2)+5760*n0*pow(b1,2)*pow(q1,2)*pow(v,3)*VVC2*f- \\
& 2880*n0*pow(v,4)*r*C1*f-180*n0*pow(v,6)*b1*q1*C1-840*n0*po \\
& w(y,2)*pow(v,2)*pow(b1,3)*pow(q1,3)*C1-5760*N0*y*pow(v,5)* \\
& av*VC2-17280*N0*y*pow(b1,2)*pow(q1,2)*pow(v,4)*VVC3-17280* \\
& N0*y*av*pow(v,4)*VC3+960*n0*pow(y,2)*av*v*pow(b1,2)*C1+3456 \\
& 0*N0*y*av*pow(v,3)*C3-480*n0*pow(b1,2)*pow(y,4)*av*v*C1+11 \\
& 520*n0*b1*q1*pow(v,4)*av*VVC2-11520*n0*b1*q1*pow(v,4)*VC2* \\
& r+5760*N0*y*pow(v,4)*pow(b1,2)*VC2-17280*n0*pow(b1,2)*pow(\\
& q1,2)*pow(v,4)*VVC3-92160*n0*pow(y,2)*pow(v,3)*b1*q1*C4-115 \\
& 200*n0*b1*q1*C4*pow(v,3)+2880*n0*C1*pow(v,2)*r*pow(f,2)+48 \\
& 0*n0*pow(y,2)*C1*pow(v,2)*pow(f,3)+5760*N0*y*pow(v,5)*C2* \\
& r-5760*N0*y*q3*br*b1*pow(v,2)*RC2*f+5760*n0*br*pow(q2,2)*po \\
& w(v,3)*br1*RC2*f-34560*n0*pow(v,3)*b1*q1*VC3*f-10440*n0*po \\
& w(y,2)*pow(b1,2)*pow(q1,2)*C1*pow(f,2)+1440*n0*pow(v,3)*av* \\
& C1*f+86400*N0*y*pow(b1,2)*pow(q1,2)*pow(v,3)*VC3-17280*n0* \\
& b1*q1*pow(v,3)*ar*RC2-2880*N0*y*pow(v,5)*pow(b1,2)*VVC2-288 \\
& 0*n0*pow(v,5)*pow(b1,2)*pow(q1,2)*VVC2+103680*N0*y*pow(v,2) \\
&)*b1*q1*C3*f+17280*N0*y*pow(v,4)*C3*f-2880*n0*b1*q1*pow(v, \\
& 2)*r*C1*f+23040*N0*pow(y,3)*pow(v,4)*b1*q1*VC4+2880*n0*po \\
& w(y,2)*b1*q1*C2*v*pow(f,2)+35040*n0*pow(b1,2)*pow(q1,2)*C2* \\
& v*f-5760*n0*pow(v,4)*b1*q1*VC2*f-23040*n0*pow(b1,2)*pow(q1, \\
& 2)*VC2*pow(v,2)*f+1920*n0*q3*br*b1*v*C1*f+30*n0*pow(y,2)*
\end{aligned}$$

$$\begin{aligned}
& \text{pow}(v, 4) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 - 120 * n0 * \text{pow}(v, 2) * \text{pow}(b1, 3) * \\
& q1 * C1 + 17280 * N0 * \text{pow}(y, 3) * \text{pow}(v, 3) * \text{pow}(b1, 2) * VC3 + 17280 * N0 * y * \\
& \text{pow}(v, 5) * b1 * q1 * VC3 + 17280 * n0 * \text{pow}(y, 2) * \text{pow}(v, 3) * \text{pow}(b1, 2) * VC3 \\
& - 1440 * n0 * \text{pow}(v, 4) * b1 * q1 * r * C1 - 252 * n0 * \text{pow}(b1, 4) * C1 - 276 * n0 * \text{po} \\
& w(y, 2) * \text{pow}(b1, 4) * C1 + 69120 * N0 * y * b1 * q1 * VC4 * \text{pow}(v, 4) + 5760 * N0 * \\
& y * b1 * q1 * \text{pow}(v, 4) * av1 * VC2 - 480 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 4) * \text{pow}(q1, 2) \\
&) * \text{pow}(v, 2) * av1 * C1 - 240 * n0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * \text{pow}(v, 2) * av1 * \\
& C1 - 103680 * N0 * y * \text{pow}(b1, 3) * q1 * v * C2 + 46080 * n0 * b1 * q1 * VC4 * \text{pow}(v, 4) \\
&) + 5760 * n0 * br * q2 * \text{pow}(v, 4) * ar1 * RC2 + 46080 * n0 * br * q2 * \text{pow}(v, 4) * \\
& RC4 - 2880 * N0 * \text{pow}(y, 3) * \text{pow}(v, 4) * \text{pow}(br, 2) * RRC3 + 34560 * N0 * y * \text{po} \\
& w(b1, 2) * VC3 * \text{pow}(v, 3) - 80 * n0 * \text{pow}(b1, 4) * \text{pow}(y, 6) * C1 + 5760 * n0 * \text{po} \\
& w(v, 5) * C2 * r - 34560 * N0 * \text{pow}(y, 3) * \text{pow}(v, 2) * \text{pow}(b1, 2) * C3 - 4320 * N0 \\
& * y * \text{pow}(v, 6) * C3 + 240 * n0 * \text{pow}(v, 2) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C1 * r + 52 \\
& 80 * n0 * \text{pow}(y, 2) * \text{pow}(v, 3) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * C2 + 960 * n0 * \text{pow}(\\
& b1, 3) * \text{pow}(y, 4) * q1 * v * C2 - 27600 * n0 * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C2 * v + 17 \\
& 280 * n0 * b1 * q1 * \text{pow}(v, 3) * C2 * r + 23040 * N0 * y * b1 * q1 * \text{pow}(v, 3) * C2 * r + \\
& 22320 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C2 * v - 8640 * N0 * y * \text{pow}(\\
& v, 3) * \text{pow}(b1, 2) * C2 + 4800 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 3) * q1 * \text{pow}(v, 2) * VC \\
& 2 - 11520 * N0 * y * b1 * q1 * \text{pow}(v, 4) * VC2 * r + 80640 * N0 * y * \text{pow}(b1, 3) * q1 * \\
& \text{pow}(v, 2) * VC2 - 1440 * n0 * b1 * q1 * C1 * \text{pow}(f, 3) + 240 * n0 * \text{pow}(b1, 2) * C1 * \\
& \text{pow}(v, 2) * f - 3000 * n0 * \text{pow}(b1, 2) * \text{pow}(y, 4) * \text{pow}(v, 2) * \text{pow}(q1, 2) * C1 \\
& * f + 2400 * n0 * \text{pow}(b1, 3) * \text{pow}(y, 6) * \text{pow}(q1, 3) * C1 * f + 564 * n0 * \text{pow}(b1 \\
& , 4) * \text{pow}(q1, 2) * C1 + 2652 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 4) * \text{pow}(q1, 2) * C1 + 2 \\
& 880 * n0 * \text{pow}(v, 4) * \text{pow}(b1, 2) * VC2 - 5760 * n0 * \text{pow}(b1, 2) * VVC3 * \text{pow}(\\
& v, 4) - 34560 * n0 * \text{pow}(y, 2) * \text{pow}(v, 2) * \text{pow}(b1, 2) * C3 + 576 * n0 * \text{pow}(b1, \\
& 4) * \text{pow}(y, 4) * C1 + 1440 * n0 * \text{pow}(b1, 3) * \text{pow}(y, 4) * \text{pow}(q1, 3) * VC2 * \text{po} \\
& w(v, 2) - 60 * n0 * \text{pow}(v, 2) * \text{pow}(b1, 3) * \text{pow}(q1, 3) * C1 + 960 * n0 * \text{pow}(y, 2) \\
&) * \text{pow}(b1, 2) * \text{pow}(v, 2) * r * C1 - 6372 * n0 * \text{pow}(b1, 4) * \text{pow}(y, 4) * \text{pow}(\\
& q1, 2) * C1 + 5760 * n0 * \text{pow}(v, 4) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * VC2 - 2880 * n0 * \\
& \text{pow}(y, 2) * \text{pow}(v, 4) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * VC2 + 53760 * n0 * \text{pow}(b1, 3) \\
&) * q1 * \text{pow}(v, 2) * VC2 + 22560 * n0 * \text{pow}(b1, 3) * \text{pow}(q1, 3) * VC2 * \text{pow}(v, 2) \\
&) - 360 * n0 * \text{pow}(y, 2) * \text{pow}(v, 5) * av * C1 + 720 * n0 * b1 * \text{pow}(y, 4) * \text{pow}(v, \\
& 3) * q1 * av * C1 - 5760 * n0 * \text{pow}(y, 2) * \text{pow}(v, 4) * ar * RC3 - 1440 * n0 * \text{pow}(\\
& y, 2) * b1 * q1 * ar * \text{pow}(v, 2) * C1 + 5760 * N0 * y * ar * \text{pow}(v, 3) * C2 - 11520 * n0 \\
& * ar * \text{pow}(v, 4) * RC3 + 4320 * n0 * ar * \text{pow}(v, 4) * C1 - 5760 * n0 * \text{pow}(v, 5) * \\
& ar * RC2 - 7800 * n0 * \text{pow}(y, 2) * \text{pow}(b1, 2) * \text{pow}(q1, 2) * br * q2 * v * C1 + 2880 \\
& * n0 * \text{pow}(br, 2) * \text{pow}(v, 4) * br2 * q2 * RC2 - 480 * n0 * \text{pow}(br, 2) * \text{pow}(v, 3) \\
&) * br2 * q2 * C1 + 2880 * N0 * y * \text{pow}(br, 2) * \text{pow}(v, 4) * br2 * q2 * RC2 - 480 * n0 \\
& * b1 * \text{pow}(y, 4) * q1 * br * \text{pow}(q2, 2) * \text{pow}(v, 2) * br1 * C1 + 3960 * n0 * \text{pow}(\\
& b1, 2) * \text{pow}(y, 4) * \text{pow}(q1, 2) * br * q2 * v * C1 + 1440 * n0 * \text{pow}(v, 6) * br * q2 * \\
& RC2 + 22560 * n0 * \text{pow}(b1, 2) * \text{pow}(q1, 2) * br * q2 * \text{pow}(v, 2) * RC2 - 5760 * N0
\end{aligned}$$

$$\begin{aligned}
& *y*br*q2*pow(v,4)*C2+480*n0*pow(v,4)*br*pow(q2,2)*br1*C1+2 \\
& 160*n0*pow(y,2)*b1*q1*br*pow(q2,2)*pow(v,2)*br1*C1+1920*n0 \\
& *br*pow(q2,3)*pow(v,4)*pow(br1,2)*RC2-240*n0*pow(y,2)*br* \\
& pow(q2,3)*pow(v,3)*pow(br1,2)*C1-17280*n0*br*pow(q2,2)*pow(\\
& v,4)*br1*RC3-17280*N0*y*br*pow(q2,2)*pow(v,4)*br1*RC3-8640* \\
& N0*y*pow(br,2)*pow(v,4)*RRC3-5760*n0*q3*br*b1*pow(v,2)*C2-5 \\
& 760*N0*y*q3*br*b1*pow(v,2)*C2-40320*N0*y*b1*q1*pow(v,3)*av \\
& *VC2-240*n0*pow(av,2)*pow(v,2)*C1+8466*n0*pow(b1,4)*pow(y, \\
& 4)*pow(q1,4)*C1-110880*n0*pow(b1,2)*pow(q1,2)*pow(v,2)*C3+ \\
& 5760*n0*ar*pow(v,4)*br1*q2*RC2-11520*n0*br*q2*pow(v,3)*av* \\
& RC2+480*n0*br*q2*pow(v,2)*av*C1-720*n0*pow(y,2)*pow(br,2)* \\
& pow(q2,2)*pow(v,2)*C1+720*n0*pow(br,2)*pow(q2,2)*pow(v,2)* \\
& C1+2880*N0*y*q3*br*b1*pow(v,3)*C1-240*n0*pow(v,3)*b1*q1*br* \\
& q2*C1-34560*n0*b1*q1*pow(v,4)*br*q2*VRC3-34560*N0*y*b1*q1* \\
& pow(v,4)*br*q2*VRC3-5760*n0*pow(br,2)*pow(v,4)*RRC3+5760*n0 \\
& *pow(br,2)*pow(v,3)*RC2-960*n0*pow(br,2)*pow(v,2)*C1-720* \\
& n0*pow(y,2)*pow(b1,2)*pow(v,2)*av1*C1+480*n0*pow(v,4)*b1*q1 \\
& *av1*C1-9600*n0*b1*q1*pow(v,3)*av1*C2+37440*n0*b1*q1*pow(\\
& v,2)*av*C2+5760*N0*y*pow(v,4)*av*C2+17280*n0*pow(y,2)*pow(\\
& v,3)*av*C3-720*n0*b1*q1*br*pow(q2,2)*pow(v,2)*br1*C1-2880* \\
& n0*pow(y,2)*pow(v,4)*pow(br,2)*RRC3-8640*n0*b1*q1*br*pow(q2 \\
& ,2)*pow(v,3)*br1*RC2-1440*n0*pow(y,2)*br*q2*pow(v,2)*av*C1 \\
& +23040*n0*pow(y,2)*pow(v,4)*br*q2*RC4+69120*n0*b1*q1*pow(\\
& v,3)*br*q2*RC3+5760*n0*pow(br,2)*pow(q2,2)*pow(v,3)*RC2-864 \\
& 0*n0*br*q2*pow(v,4)*C2+5760*n0*pow(v,4)*b1*q1*br*q2*RC2+96 \\
& 00*n0*q3*br*pow(b1,2)*pow(v,2)*q1*RC2+1440*n0*pow(b1,2)*po \\
& w(y,4)*pow(q1,2)*br*q2*pow(v,2)*RC2+17280*N0*y*pow(v,5)*br* \\
& q2*RC3+17280*n0*pow(v,5)*br*q2*RC3+3840*n0*q3*br*b1*pow(v,3 \\
&)*C1-2880*n0*pow(y,2)*pow(v,4)*b1*q1*br*q2*RC2-67680*n0*po \\
& w(b1,3)*q1*v*C2-6960*n0*pow(v,3)*pow(b1,2)*pow(q1,2)*C2-100 \\
& 80*n0*pow(y,2)*pow(b1,3)*q1*v*C2-4560*n0*pow(b1,3)*pow(y,4 \\
&)*pow(q1,3)*C2*v+5760*n0*b1*q1*pow(v,4)*av1*VC2-960*n0*po \\
& w(y,2)*pow(v,3)*pow(b1,2)*C2-2880*n0*pow(y,2)*pow(v,4)*pow(\\
& b1,2)*VVC3+960*n0*br*pow(q2,2)*pow(v,2)*br1*C1*f-960*n0*po \\
& w(y,2)*br*pow(q2,2)*pow(v,2)*br1*C1*f+1440*n0*av*v*C1*pow(\\
& f,2)+4560*n0*pow(b1,3)*pow(y,4)*q1*C1*f+480*n0*pow(y,2)*po \\
& w(v,4)*b1*q1*av1*C1-240*n0*pow(b1,2)*pow(v,2)*av1*C1+1680* \\
& n0*pow(y,2)*pow(b1,2)*pow(q1,2)*pow(v,2)*av1*C1+11520*n0*po \\
& w(v,4)*C3*r-31680*n0*pow(b1,2)*pow(v,2)*C3-2880*N0*y*pow(r, \\
& 2)*pow(v,4)*C1-11520*N0*y*b1*q1*pow(v,3)*av1*C2-240*n0*po \\
& w(y,2)*pow(b1,2)*pow(q1,2)*pow(v,3)*av2*C1+3000*n0*pow(b1,2
\end{aligned}$$

```

    )*pow(y,4)*pow(q1,2)*av*v*C1-5760*n0*pow(y,2)*pow(v,4)*av*
    VC3+17280*N0*pow(y,3)*pow(v,3)*av*C3-28800*n0*b1*q1*pow(v,3
    )*av*VC2-3480*n0*pow(y,2)*pow(b1,2)*pow(q1,2)*av*v*C1+5760
    0*N0*y*b1*q1*pow(v,2)*av*C2-34560*N0*y*pow(v,5)*C4-11520*
    N0*pow(y,3)*pow(v,5)*C4-11520*n0*pow(y,2)*pow(v,5)*C4-17280
    *n0*pow(v,2)*pow(b1,2)*VC2*f+11520*n0*pow(v,3)*ar*RC2*f-20
    160*n0*pow(b1,3)*q1*pow(v,3)*VVC2+69120*n0*pow(b1,2)*pow(
    q1,2)*pow(v,3)*VC3+720*n0*pow(b1,2)*pow(y,4)*pow(v,2)*pow(
    q1,2)*C1*r-2880*n0*pow(v,5)*pow(b1,2)*VVC2-5760*N0*pow(y,3)
    *pow(v,4)*ar*RC3-17280*N0*y*ar*pow(v,4)*RC3+2880*N0*y*ar*
    pow(v,4)*C1-480*n0*pow(b1,2)*pow(v,3)*av2*C1-720*n0*pow(b1,
    2)*pow(y,4)*pow(v,3)*pow(q1,2)*C2)/(pow(v,4)*(15*N0*y+N0*
    pow(y,5)+10*N0*pow(y,3)+n0*pow(y,4)+9*n0*pow(y,2)+8*n0));
}

//Elements of the formula
static void fPn1(double C1, double * result)
{
    * result=C1;
}
static void fPN1(double C1, double z, double * result)
{
    * result=z*C1;
}
static void fPn2(double C1, double C2, double v, double b1,
    double q1, double z, double * result)
{
    *result=1/2.*z*(2*C2*v+b1*q1*C1)/v;
}
static void fPN2(double C1, double C2, double v, double z,
    double f, double * result)
{
    *result=1/2.*(2*C2*v*pow(z,2)+2*C2*v+C1*pow(v,2)-2*C1*f)
    /v;
}
static void fPn3(double C1, double C2, double C3, double VC
    2, double RC2,double r, double v, double z, double f,
    double av, double b1, double br, double q1, double q2, double *
    result)
{

```



```

*result=1/24.*(24*C3*pow(v,2)*pow(z,2)+48*C3*pow(v,2)+3*
    pow(b1,2)*pow(q1,2)*C1*pow(z,4)-6*b1*C1*pow(z,2)*q1*pow(v,2)
    )-10*pow(b1,2)*C1*pow(z,2)*pow(q1,2)+12*b1*C1*pow(z,2)*q1*
    f+4*pow(b1,2)*C1*pow(z,2)+12*C1*pow(f,2)-12*C1*pow(v,2)*f+1
    2*av*v*C1-24*r*C1*pow(v,2)+2*pow(b1,2)*C1-12*b1*q1*C1*f+12
    *br*q2*v*C1-48*b1*q1*VC2*pow(v,2)+72*b1*q1*C2*v-48*br*q2*
    pow(v,2)*RC2-48*C2*v*f+24*C2*pow(v,3)+3*C1*pow(v,4)+pow(b1,
    2)*pow(q1,2)*C1-6*b1*q1*C1*pow(v,2))/pow(v,2);
}
static void fPN3(double C1, double C2, double C3, double VC
    2, double RC2, double z, double r, double v, double f,
    double b1, double br, double q1, double q2, double * result)
{
    *result=-z*(-C3*v*pow(z,2)-3*C3*v+2*br*q2*v*RC2+2*C2*f-4
        *b1*q1*C2+2*b1*q1*VC2*v+r*C1*v-C2*pow(v,2))/v;
}
static void fPn4(double C1, double C2, double C3, double C4
    , double VC2, double VVC2, double VC3, double RC2, double
    RC3, double b1, double br, double br1, double q1, double q2
    , double av, double av1, double ar, double z, double v,
    double y, double r, double f, double * result )
{
    *result= 1/48.*z*(48*ar*pow(v,3)*RC2-48*C2*pow(v,3)*r+12
        8*pow(b1,2)*C2*v+24*pow(b1,2)*VVC2*pow(v,3)-12*av*pow(v,3)
        *C1-96*av*pow(v,2)*C2+48*av*pow(v,3)*VC2-144*b1*q1*VC3*po
        w(v,3)+8*b1*q1*pow(v,2)*av1*C1-12*br*q2*pow(v,3)*C1-144*br*
        q2*pow(v,3)*RC3-28*b1*q1*br*q2*v*C1-C1*pow(v,6)+72*C3*pow(
        v,4)+42*pow(b1,2)*pow(q1,2)*C1*f-36*b1*q1*C1*pow(f,2)+24*av
        *v*C1*f+pow(b1,3)*pow(q1,3)*C1*pow(z,6)-18*pow(b1,3)*pow(
        z,2)*q1*C1+33*pow(b1,3)*pow(z,2)*pow(q1,3)*C1-4*pow(b1,2)*
        pow(z,2)*C1*pow(v,2)+8*pow(b1,2)*pow(z,2)*C1*f+4*pow(b1,3)*
        q1*C1*pow(z,4)-13*pow(b1,3)*pow(q1,3)*C1*pow(z,4)+8*br*pow(
        q2,2)*pow(v,2)*br1*C1-48*b1*q1*br*q2*pow(v,2)*RC2+24*b1*q1*
        C2*pow(v,3)+432*b1*q1*C3*pow(v,2)-48*b1*q1*C2*v*f-12*pow(b1
        ,2)*C1*f+6*C1*pow(v,4)*f-12*C1*pow(v,2)*pow(f,2)-144*C3*po
        w(v,2)*f+240*C4*pow(v,3)+48*C4*pow(v,3)*pow(z,2)+8*C1*pow(
        f,3)+12*b1*q1*C1*pow(v,2)*f+24*br*q2*v*C1*f-12*pow(b1,2)*po
        w(z,2)*pow(q1,2)*C2*v+12*b1*pow(z,2)*q1*av*v*C1+12*b1*pow(
        z,2)*q1*br*q2*v*C1-12*b1*pow(z,2)*q1*C1*pow(v,2)*f+12*b1*po
        w(z,2)*q1*C1*pow(f,2)+10*pow(b1,2)*pow(z,2)*pow(q1,2)*C1*po
        w(v,2)+3*b1*pow(z,2)*q1*C1*pow(v,4)-44*pow(b1,2)*pow(z,2)*

```

```

pow(q1,2)*C1*f+6*pow(b1,2)*pow(q1,2)*C1*pow(z,4)*f-3*pow(b1
,2)*pow(q1,2)*C1*pow(z,4)*pow(v,2)-96*pow(b1,2)*VC2*pow(v,
2)-12*b1*q1*av*v*C1-10*pow(b1,2)*C1*pow(v,2)-9*pow(b1,3)*
pow(q1,3)*C1+2*pow(b1,3)*q1*C1-pow(b1,2)*pow(q1,2)*C1*pow(
v,2)-24*b1*q1*pow(v,2)*r*C1+100*pow(b1,2)*pow(q1,2)*C2*v+3*
b1*q1*C1*pow(v,4)-48*pow(b1,2)*pow(q1,2)*VC2*pow(v,2))/pow(
v,3);
}
static void fPN4(double C1, double C2, double C3, double C4
, double VC2, double VVC2, double VC3, double RC2, double
RC3, double b1, double br, double br1, double q1, double q2
, double av, double av1, double ar, double z, double v,
double y, double r, double f, double * result )
{
*result=1/4.*(-4*av*v*C2+4*av*pow(v,2)*VC2+4*ar*pow(v,2)
*RC2+8*br*q2*v*RC2*f+8*b1*q1*VC2*v*f-4*C2*pow(v,2)*r+6*po
w(z,2)*C3*pow(v,3)+24*b1*q1*C3*v+6*pow(b1,2)*C2+2*pow(b1,2)
*VVC2*pow(v,2)+6*C3*pow(v,3)-4*pow(b1,2)*VC2*v-12*br*q2*po
w(v,2)*RC3+2*br*q2*pow(v,2)*C1+4*b1*q1*C2*pow(v,2)+4*pow(b1
,2)*pow(q1,2)*pow(v,2)*VVC2-4*b1*q1*VC2*pow(v,3)+4*br*q2*
v*C2-4*br*q2*pow(v,3)*RC2+16*pow(b1,2)*pow(q1,2)*C2-12*pow(
b1,2)*pow(q1,2)*VC2*v+4*r*C1*v*f-16*b1*q1*C2*f+4*pow(z,2)*
ar*pow(v,2)*RC2-12*pow(z,2)*C3*v*f-8*pow(z,2)*av*v*C2-8*po
w(z,2)*pow(b1,2)*VC2*v-12*b1*q1*VC3*pow(v,2)+2*pow(z,2)*po
w(b1,2)*VVC2*pow(v,2)+12*C4*pow(v,2)-4*pow(z,2)*C2*pow(v,2)
*r+4*pow(z,2)*av*pow(v,2)*VC2-12*C3*v*f-2*r*C1*pow(v,3)+2*
av*pow(v,2)*C1-2*ar*v*C1+C2*pow(v,4)+4*C2*pow(f,2)-12*pow(
z,2)*br*q2*pow(v,2)*RC3-12*pow(z,2)*b1*q1*VC3*pow(v,2)+36*
pow(z,2)*b1*q1*C3*v+4*br*pow(q2,2)*pow(v,2)*br1*RC2-12*b1*
q1*br*q2*v*RC2+pow(b1,2)*C1*v-4*C2*pow(v,2)*f+12*pow(z,2)*
pow(b1,2)*C2+4*C4*pow(v,2)*pow(z,4)+24*C4*pow(v,2)*pow(z,2)
)/pow(v,2);
}
static void fPn5(double C1, double C2, double C3, double C4
, double C5, double VC2, double VVC2, double VVC3, double
VC3, double VC4, double RC2, double RC3, double RC4,
double RRC3, double VRC3, double b1, double br, double br1,
double br2, double q1, double q2, double q3, double av, double
av1, double av2, double ar, double ar1, double z, double
v, double y, double r, double f, double * result )
{

```

```

*result=1/5760.*(-11520*pow(v,4)*C3*r+31680*pow(b1,2)*po
w(v,2)*C3+2880*pow(v,5)*pow(b1,2)*VVC2-17280*pow(v,5)*b1*q1
*VC3+67680*pow(b1,3)*q1*v*C2+6960*pow(v,3)*pow(b1,2)*pow(
q1,2)*C2+4320*pow(v,6)*C3+23040*pow(v,5)*C4+240*pow(b1,2)*
pow(q1,2)*pow(v,2)*av1*C1-720*C1*pow(v,6)*r-46080*b1*q1*VC4
*pow(v,4)-480*pow(v,4)*b1*q1*av1*C1+2880*pow(v,5)*pow(b1,2
)*pow(q1,2)*VVC2+480*b1*pow(z,4)*q1*C1*pow(f,3)-3000*pow(
b1,2)*pow(z,4)*pow(q1,2)*av*v*C1-720*b1*pow(z,4)*pow(v,3)*
q1*av*C1-60*b1*pow(z,4)*pow(v,6)*q1*C1+11760*pow(b1,3)*pow(
z,4)*pow(q1,3)*C1*f-4440*pow(b1,2)*pow(z,4)*pow(q1,2)*C1*po
w(f,2)-4560*pow(b1,3)*pow(z,4)*q1*C1*f-480*pow(b1,2)*pow(z,
4)*pow(v,2)*C1*f+3000*pow(b1,2)*pow(z,4)*pow(v,2)*pow(q1,2
)*C1*f-1440*pow(b1,2)*pow(z,4)*pow(q1,2)*C2*v*f+1440*b1*po
w(z,4)*q1*br*q2*v*C1*f+1440*b1*pow(z,4)*q1*av*v*C1*f-720*po
w(b1,2)*pow(z,4)*pow(v,2)*pow(q1,2)*C1*r-576*pow(b1,4)*pow(
z,4)*C1+46080*C5*pow(v,4)+5760*C5*pow(v,4)*pow(z,4)+51840*
C5*pow(v,4)*pow(z,2)+240*pow(b1,2)*pow(v,2)*av1*C1-53760*po
w(b1,3)*q1*pow(v,2)*VC2+120*C1*pow(v,6)*f-69120*pow(b1,2)*
pow(q1,2)*pow(v,3)*VC3-1320*pow(b1,2)*pow(q1,2)*br*q2*v*C1+
360*b1*pow(z,4)*pow(v,4)*q1*C1*f-720*b1*pow(z,4)*pow(v,2)*
q1*C1*pow(f,2)+2880*pow(r,2)*pow(v,4)*C1-22560*pow(b1,2)*po
w(q1,2)*br*q2*pow(v,2)*RC2+240*pow(v,3)*b1*q1*br*q2*C1+720*
b1*q1*br*pow(q2,2)*pow(v,2)*br1*C1-480*pow(b1,2)*pow(v,2)*
r*C1+9600*b1*q1*pow(v,3)*av1*C2-5760*b1*q1*pow(v,4)*av1*VC2
-17280*av*pow(v,3)*C3-1440*br*q2*v*C1*pow(f,2)-1440*pow(v,
3)*br*q2*C1*f+23040*b1*q1*br*q2*pow(v,2)*RC2*f-960*br*pow(
q2,2)*pow(v,2)*br1*C1*f-480*av*v*pow(b1,2)*C1+120*pow(b1,2)
*pow(q1,2)*av*v*C1-720*pow(v,3)*b1*q1*av*C1-5760*pow(v,4)*
pow(b1,2)*pow(q1,2)*VC2+720*pow(b1,2)*C1*pow(f,2)-360*pow(
v,4)*C1*pow(f,2)+23040*pow(b1,2)*pow(q1,2)*VC2*pow(v,2)*f-2
880*av*pow(v,3)*r*C1+60*pow(v,2)*pow(b1,3)*pow(q1,3)*C1-56
4*pow(b1,4)*pow(q1,2)*C1+1080*pow(v,5)*av*C1+240*pow(av,2)
*pow(v,2)*C1+11520*av*pow(v,4)*VC3+5760*pow(v,5)*av*VC2-28
80*pow(v,4)*av*C2+177*pow(b1,4)*pow(q1,4)*C1-240*pow(b1,3)
*pow(z,6)*pow(q1,3)*C2*v+360*pow(b1,2)*pow(z,6)*pow(q1,2)*
av*v*C1+480*pow(b1,3)*pow(z,6)*q1*C1*f+360*pow(b1,2)*pow(z,
6)*pow(q1,2)*br*q2*v*C1-5760*pow(z,2)*pow(v,4)*C3*r+34560*
pow(z,2)*pow(b1,2)*pow(v,2)*C3-120*pow(z,2)*C1*pow(v,6)*f-1
7280*pow(z,2)*av*pow(v,3)*C3-2160*pow(z,2)*pow(b1,2)*C1*po
w(f,2)+360*pow(z,2)*pow(v,4)*C1*pow(f,2)-2652*pow(z,2)*pow(
b1,4)*pow(q1,2)*C1+360*pow(z,2)*pow(v,5)*av*C1+720*pow(z,2)

```

```

*pow(av,2)*pow(v,2)*C1+5760*pow(z,2)*av*pow(v,4)*VC3+3771*
pow(z,2)*pow(b1,4)*pow(q1,4)*C1+2880*pow(z,2)*pow(br,2)*po
w(v,4)*RRC3+2880*pow(z,2)*pow(b1,2)*VVC3*pow(v,4)-17280*po
w(z,2)*pow(b1,2)*VC3*pow(v,3)+420*pow(z,2)*pow(v,4)*pow(b1,
2)*C1+5760*pow(z,2)*ar*pow(v,4)*RC3-480*pow(z,2)*pow(v,2)*
C1*pow(f,3)-23040*pow(z,2)*C4*pow(v,3)*f+960*pow(z,2)*pow(
v,3)*pow(b1,2)*C2+11520*pow(z,2)*pow(v,5)*C4+480*pow(b1,2)*
pow(v,3)*av2*C1+28800*b1*q1*pow(v,3)*av*VC2-37440*b1*q1*po
w(v,2)*av*C2+27600*pow(b1,3)*pow(q1,3)*C2*v+120*pow(v,2)*po
w(b1,2)*pow(q1,2)*C1*f-1920*q3*br*b1*v*C1*f-5760*br*q2*pow(
v,2)*C2*f+5760*pow(v,4)*br*q2*RC2*f+115200*b1*q1*C4*pow(v,3
)+960*av*pow(v,3)*av1*C1+720*pow(b1,2)*q1*q3*br*v*C1+5760*
pow(br,2)*pow(v,4)*RRC3+960*pow(br,2)*pow(v,2)*C1-5760*pow(
br,2)*pow(v,3)*RC2-22560*pow(b1,3)*pow(q1,3)*VC2*pow(v,2)+1
0080*pow(z,2)*pow(b1,3)*q1*v*C2-5280*pow(z,2)*pow(v,3)*po
w(b1,2)*pow(q1,2)*C2-1680*pow(z,2)*pow(b1,2)*pow(q1,2)*pow(
v,2)*av1*C1-23040*pow(z,2)*b1*q1*VC4*pow(v,4)-480*pow(z,2)*
pow(v,4)*b1*q1*av1*C1+720*pow(z,2)*pow(b1,2)*pow(v,2)*av1*
C1-4800*pow(z,2)*pow(b1,3)*q1*pow(v,2)*VC2+7800*pow(z,2)*po
w(b1,2)*pow(q1,2)*br*q2*v*C1-1440*pow(z,2)*br*q2*v*pow(b1,2
)*C1+13440*pow(z,2)*pow(b1,2)*pow(q1,2)*br*q2*pow(v,2)*RC2
+960*pow(z,2)*pow(v,3)*b1*q1*br*q2*C1-2160*pow(z,2)*b1*q1*
br*pow(q2,2)*pow(v,2)*br1*C1-960*pow(z,2)*pow(b1,2)*pow(v,2
)*r*C1+1440*pow(z,2)*br*q2*v*C1*pow(f,2)-1440*pow(z,2)*po
w(v,3)*br*q2*C1*f-5760*pow(z,2)*b1*q1*br*q2*pow(v,2)*RC2*f+
960*pow(z,2)*br*pow(q2,2)*pow(v,2)*br1*C1*f-960*pow(z,2)*
av*v*pow(b1,2)*C1+3480*pow(z,2)*pow(b1,2)*pow(q1,2)*av*v*C1
+2880*pow(z,2)*pow(v,4)*pow(b1,2)*pow(q1,2)*VC2-5760*pow(
z,2)*pow(b1,2)*pow(q1,2)*VC2*pow(v,2)*f+840*pow(z,2)*pow(v,
2)*pow(b1,3)*pow(q1,3)*C1-2880*pow(z,2)*b1*q1*pow(v,2)*av*
C2-22320*pow(z,2)*pow(b1,3)*pow(q1,3)*C2*v-3720*pow(z,2)*po
w(v,2)*pow(b1,2)*pow(q1,2)*C1*f+92160*pow(z,2)*b1*q1*C4*po
w(v,3)-720*pow(z,2)*pow(b1,2)*q1*q3*br*v*C1+13440*pow(z,2)*
pow(b1,3)*pow(q1,3)*VC2*pow(v,2)-720*pow(z,2)*pow(v,5)*b1*
q1*C2+16320*pow(z,2)*pow(b1,2)*pow(q1,2)*C2*v*f-720*pow(z,2
)*pow(v,4)*b1*q1*C1*f-5760*br*pow(q2,2)*pow(v,3)*br1*RC2*
f-5760*pow(b1,2)*pow(q1,2)*pow(v,3)*VVC2*f-86400*pow(v,2)*
b1*q1*C3*f-17280*pow(v,4)*C3*f+17280*pow(v,2)*C3*pow(f,2)-1
920*C2*v*pow(f,3)+2880*pow(v,3)*C2*pow(f,2)-1440*C2*pow(v,
5)*f+80*pow(b1,4)*pow(z,6)*C1+15*pow(z,2)*pow(v,8)*C1+240*
pow(z,2)*C1*pow(f,4)+276*pow(z,2)*pow(b1,4)*C1+720*pow(v,5)

```

$$\begin{aligned}
& *b1*q1*C2-11520*b1*q1*pow(v,4)*av*VVC2-15*pow(v,8)*C1+240* \\
& pow(v,7)*C2-60*pow(b1,3)*pow(q1,3)*C1*pow(z,8)*pow(v,2)+120 \\
& *pow(b1,3)*pow(q1,3)*C1*pow(z,8)*f+360*pow(b1,2)*pow(z,6)* \\
& pow(q1,2)*C1*pow(f,2)+840*pow(b1,3)*pow(z,6)*pow(v,2)*pow(\\
& q1,3)*C1+90*pow(b1,2)*pow(z,6)*pow(v,4)*pow(q1,2)*C1-2400* \\
& pow(b1,3)*pow(z,6)*pow(q1,3)*C1*f+3530*pow(b1,4)*pow(z,6)* \\
& pow(q1,4)*C1-35040*pow(b1,2)*pow(q1,2)*C2*v*f+3360*b1*q1*br \\
& *q2*v*C1*f+1440*b1*q1*av*v*C1*f+2880*b1*q1*pow(v,2)*r*C1* \\
& f-360*pow(v,4)*b1*q1*C1*f-720*pow(v,2)*b1*q1*C1*pow(f,2)+14 \\
& 400*b1*q1*C2*v*pow(f,2)-960*b1*q1*pow(v,2)*av1*C1*f+110880 \\
& *pow(b1,2)*pow(q1,2)*pow(v,2)*C3+12480*b1*q1*br*q2*pow(v,2) \\
&)*C2-720*pow(br,2)*pow(q2,2)*pow(v,2)*C1+1080*pow(v,5)*br* \\
& q2*C1-17280*pow(v,5)*br*q2*RC3-8640*pow(v,3)*b1*q1*C2*f+115 \\
& 20*pow(v,3)*C2*r*f-5760*pow(v,3)*pow(b1,2)*VVC2*f+17280*po \\
& w(v,2)*pow(b1,2)*VC2*f-11520*pow(v,3)*av*VC2*f+17280*pow(v, \\
& 2)*av*C2*f-11520*pow(v,3)*ar*RC2*f+2880*ar*pow(v,2)*C1*f+1 \\
& 20*pow(v,2)*pow(b1,3)*q1*C1-17280*b1*q1*pow(v,3)*C2*r-240* \\
& C1*pow(f,4)+8640*pow(b1,3)*pow(q1,3)*pow(v,3)*VVC2+25920*po \\
& w(v,4)*b1*q1*C3-7680*pow(z,2)*b1*q1*br*q2*v*C1*f-5760*pow(\\
& z,2)*b1*q1*av*v*C1*f-2880*pow(z,2)*b1*q1*pow(v,2)*r*C1*f+28 \\
& 80*pow(z,2)*pow(v,2)*b1*q1*C1*pow(f,2)-2880*pow(z,2)*b1*q1 \\
& *C2*v*pow(f,2)+960*pow(z,2)*b1*q1*pow(v,2)*av1*C1*f+4320* \\
& pow(z,2)*pow(b1,2)*pow(q1,2)*pow(v,2)*C3-2880*pow(z,2)*b1* \\
& q1*br*q2*pow(v,2)*C2+720*pow(z,2)*pow(br,2)*pow(q2,2)*pow(\\
& v,2)*C1+360*pow(z,2)*pow(v,5)*br*q2*C1+2880*pow(z,2)*pow(v, \\
& 3)*b1*q1*C2*f-2880*pow(z,2)*pow(b1,3)*pow(q1,3)*pow(v,3)* \\
& VVC2-30*pow(z,2)*pow(v,4)*pow(b1,2)*pow(q1,2)*C1-480*pow(z, \\
& 2)*pow(v,4)*br*pow(q2,2)*br1*C1-2880*pow(z,2)*b1*q1*br*po \\
& w(q2,2)*pow(v,3)*br1*RC2+240*pow(z,2)*br*pow(q2,3)*pow(v,3) \\
& *pow(br1,2)*C1+1440*pow(z,2)*br*q2*pow(v,2)*av*C1-1920*po \\
& w(z,2)*br*q2*pow(v,2)*pow(b1,2)*RC2+5760*pow(z,2)*q3*br*b1* \\
& VRC3*pow(v,4)+2880*pow(z,2)*pow(v,4)*b1*q1*br*q2*RC2+240*po \\
& w(z,2)*pow(br,2)*pow(q2,3)*pow(v,3)*br2*C1+6720*pow(z,2)*po \\
& w(b1,3)*q1*C1*f+240*pow(z,2)*pow(v,2)*pow(b1,2)*C1*f-2880* \\
& pow(z,2)*b1*q1*C1*pow(f,3)-1440*pow(z,2)*av*pow(v,3)*C1*f+1 \\
& 440*pow(z,2)*av*v*C1*pow(f,2)-1920*pow(z,2)*v*pow(b1,2)*C2 \\
& *f+2400*pow(z,2)*pow(v,2)*pow(b1,2)*pow(q1,2)*C1*r+1440*po \\
& w(z,2)*b1*q1*ar*pow(v,2)*C1+1440*pow(z,2)*pow(v,4)*b1*q1*r* \\
& C1-150*pow(v,4)*pow(b1,2)*pow(q1,2)*C1+252*pow(b1,4)*C1+576 \\
& 0*pow(b1,2)*VVC3*pow(v,4)-480*pow(v,4)*br*pow(q2,2)*br1*C1 \\
& +8640*b1*q1*br*pow(q2,2)*pow(v,3)*br1*RC2-240*br*pow(q2,3)
\end{aligned}$$

```

*pow(v,3)*pow(br1,2)*C1-5760*q3*br*b1*pow(v,3)*VC2-480*br*
q2*pow(v,2)*av*C1-16320*br*q2*pow(v,2)*pow(b1,2)*RC2+11520*
q3*br*b1*VRC3*pow(v,4)-2880*pow(v,3)*br*q2*C1*r-5760*pow(v,
4)*b1*q1*br*q2*RC2-240*pow(br,2)*pow(q2,3)*pow(v,3)*br2*C1
-17280*pow(b1,2)*VC3*pow(v,3)+17280*pow(br,2)*pow(q2,2)*po
w(v,4)*RRC3+900*pow(v,4)*pow(b1,2)*C1-1440*pow(v,6)*b1*q1*
VC2-240*pow(b1,3)*q1*C1*f-240*pow(v,2)*pow(b1,2)*C1*f+1440*
b1*q1*C1*pow(f,3)-1440*av*pow(v,3)*C1*f-1440*av*v*C1*pow(f,
2)-24000*v*pow(b1,2)*C2*f+180*pow(v,6)*b1*q1*C1-240*pow(v,
2)*pow(b1,2)*pow(q1,2)*C1*r+11520*ar*pow(v,4)*RC3+5760*po
w(v,5)*ar*RC2-4320*ar*pow(v,4)*C1-5760*ar*pow(v,3)*C2-1440*
b1*q1*ar*pow(v,2)*C1+1440*pow(v,4)*b1*q1*r*C1-9600*q3*br*po
w(b1,2)*pow(v,2)*q1*RC2+8640*br*q2*pow(v,4)*C2-1440*pow(v,6
)*br*q2*RC2-5760*pow(br,2)*pow(q2,2)*pow(v,3)*RC2+17280*br
*q2*pow(v,3)*C3-5760*pow(v,5)*C2*r+17280*b1*q1*pow(v,3)*ar
*RC2-69120*b1*q1*pow(v,3)*br*q2*RC3+11520*br*q2*pow(v,3)*
av*RC2+34560*b1*q1*pow(v,4)*br*q2*VRC3+2880*q3*br*b1*pow(v,
4)*RC2-5760*br*q2*pow(v,4)*ar1*RC2-5760*b1*q1*br*q2*pow(v,
3)*VC2+960*br*q2*pow(v,3)*ar1*C1+11520*br*q2*pow(v,4)*RC2*
r+34560*pow(v,3)*b1*q1*VC3*f-5760*b1*q1*VC2*pow(v,2)*pow(f,
2)+5760*pow(v,4)*b1*q1*VC2*f-5760*br*q2*pow(v,2)*RC2*pow(
f,2)+5760*q3*br*b1*pow(v,2)*RC2*f+34560*pow(v,3)*br*q2*RC3*
f-435*pow(b1,4)*pow(q1,4)*C1*pow(z,8)+120*pow(b1,4)*pow(q1,
2)*C1*pow(z,8)-1900*pow(b1,4)*pow(z,6)*pow(q1,2)*C1+480*po
w(v,2)*C1*pow(f,3)-46080*C4*pow(v,3)*f+7200*pow(v,3)*pow(b1
,2)*C2-1920*pow(br,2)*pow(q2,3)*pow(v,4)*br2*RC2+480*pow(
br,2)*pow(v,3)*br2*q2*C1-2880*pow(br,2)*pow(v,4)*br2*q2*RC2
-3840*q3*br*b1*pow(v,3)*C1+5760*q3*br*b1*pow(v,2)*C2+2880*
pow(v,5)*br*pow(q2,2)*br1*RC2+17280*br*pow(q2,2)*pow(v,4)*
br1*RC3-1920*br*pow(q2,3)*pow(v,4)*pow(br1,2)*RC2-1920*br*
pow(q2,2)*pow(v,3)*br1*C2+20160*pow(b1,3)*q1*pow(v,3)*VVC2+
240*pow(z,2)*pow(b1,2)*pow(q1,2)*pow(v,3)*av2*C1-12960*po
w(z,2)*pow(b1,3)*pow(q1,3)*C1*f+10440*pow(z,2)*pow(b1,2)*po
w(q1,2)*C1*pow(f,2)-23040*pow(z,2)*br*q2*pow(v,4)*RC4-960*
pow(b1,3)*pow(z,4)*q1*v*C2+720*pow(b1,2)*pow(z,4)*pow(v,3)*
pow(q1,2)*C2+480*pow(b1,2)*pow(z,4)*av*v*C1-2400*pow(b1,3)*
pow(z,4)*pow(v,2)*pow(q1,3)*C1+4560*pow(b1,3)*pow(z,4)*pow(
q1,3)*C2*v-1440*pow(b1,3)*pow(z,4)*pow(q1,3)*VC2*pow(v,2)+8
40*pow(b1,3)*pow(z,4)*pow(v,2)*q1*C1-390*pow(b1,2)*pow(z,4
)*pow(v,4)*pow(q1,2)*C1+480*pow(b1,2)*pow(z,4)*pow(q1,2)*
pow(v,2)*av1*C1-3960*pow(b1,2)*pow(z,4)*pow(q1,2)*br*q2*v*

```



```

C1+480*pow(b1,2)*pow(z,4)*br*q2*v*C1-1440*pow(b1,2)*pow(z,4)
)*pow(q1,2)*br*q2*pow(v,2)*RC2-720*b1*pow(z,4)*pow(v,3)*q1
*br*q2*C1+480*b1*pow(z,4)*q1*br*pow(q2,2)*pow(v,2)*br1*C1-
8466*pow(b1,4)*pow(z,4)*pow(q1,4)*C1+480*pow(b1,2)*pow(z,4)
)*C1*pow(f,2)+6372*pow(b1,4)*pow(z,4)*pow(q1,2)*C1+120*po
w(b1,2)*pow(z,4)*pow(v,4)*C1+11520*b1*q1*pow(v,4)*VC2*r-288
0*pow(v,4)*pow(b1,2)*VC2-240*pow(b1,3)*pow(z,6)*pow(v,2)*
q1*C1-360*pow(b1,2)*pow(z,6)*pow(v,2)*pow(q1,2)*C1*f+17280*
pow(b1,2)*pow(q1,2)*pow(v,4)*VVC3-240*pow(b1,2)*pow(q1,2)*
pow(v,3)*av2*C1+960*ar*pow(v,3)*br1*q2*C1+15*pow(b1,4)*pow(
q1,4)*C1*pow(z,10)+2880*pow(v,4)*C1*r*f-2880*pow(v,2)*C1*r*
pow(f,2)+1080*pow(b1,3)*pow(q1,3)*C1*f-2520*pow(b1,2)*pow(
q1,2)*C1*pow(f,2)-5760*ar*pow(v,4)*br1*q2*RC2-46080*br*q2*
pow(v,4)*RC4)/pow(v,4);
}
static void fPN5(double C1, double C2, double C3, double C4
, double C5, double VC2, double VVC2, double VVC3, double
VC3, double VC4, double RC2, double RC3, double RC4,
double RRC3, double VRC3, double b1, double br, double br1,
double br2, double q1, double q2, double q3, double av, double
av1, double ar, double ar1, double z, double v, double y,
double r, double f, double * result )
{
*result=1/4.*z*(8*b1*q1*pow(v,2)*av1*C2-4*b1*q1*pow(v,3)
*av1*VC2+28*b1*q1*pow(v,2)*av*VC2-40*b1*q1*v*av*C2-8*b1*q1
*pow(v,3)*av*VVC2-16*b1*q1*pow(v,2)*C2*r+24*pow(v,3)*b1*q1
*C3+12*pow(br,2)*pow(q2,2)*pow(v,3)*RRC3+4*br*q2*pow(v,3)*
C2+12*br*q2*pow(v,2)*C3+20*pow(b1,3)*q1*pow(v,2)*VVC2+6*po
w(b1,2)*VVC3*pow(v,3)-4*q3*br*b1*pow(v,2)*VC2-16*br*q2*v*po
w(b1,2)*RC2+12*q3*br*b1*VRC3*pow(v,3)+4*pow(v,4)*ar*RC2-2*
ar*pow(v,3)*C1-4*ar*pow(v,2)*C2-4*pow(v,4)*C2*r+6*pow(v,2)*
pow(b1,2)*C2-4*pow(v,3)*pow(b1,2)*VC2-24*pow(b1,2)*C2*f-48*
C4*pow(v,2)*f-12*pow(v,3)*C3*f+12*v*C3*pow(f,2)+8*pow(z,2)*
pow(v,4)*C4+4*C5*pow(v,3)*pow(z,4)+40*C5*pow(v,3)*pow(z,2)+
24*pow(v,2)*b1*q1*VC3*f+4*q3*br*b1*v*RC2*f+24*pow(v,2)*br*
q2*RC3*f-16*pow(z,2)*pow(v,3)*br*q2*RC4-12*pow(z,2)*pow(v,2)
)*pow(b1,2)*VC3+2*pow(z,2)*pow(v,3)*pow(br,2)*RRC3-16*pow(
z,2)*pow(v,2)*C4*f+4*pow(z,2)*pow(v,3)*q3*br*b1*VRC3-4*pow(
z,2)*pow(v,3)*C3*r+4*pow(z,2)*pow(v,3)*ar*RC3+4*pow(z,2)*po
w(v,3)*av*VC3-12*pow(z,2)*pow(v,2)*av*C3+24*pow(z,2)*v*pow(
b1,2)*C3+2*pow(z,2)*pow(v,3)*pow(b1,2)*VVC3+60*C5*pow(v,3)-

```

```

8*q3*br*pow(b1,2)*v*q1*RC2+16*b1*q1*pow(v,2)*ar*RC2-60*b1*
q1*pow(v,2)*br*q2*RC3+12*br*q2*pow(v,2)*av*RC2+24*b1*q1*po
w(v,3)*br*q2*VRC3+2*q3*br*b1*pow(v,3)*RC2-4*br*q2*pow(v,3)*
ar1*RC2+8*br*q2*pow(v,3)*RC2*r-2*pow(br,2)*pow(v,3)*br2*q2*
RC2-2*q3*br*b1*pow(v,2)*C1+4*q3*br*b1*v*C2+12*br*pow(q2,2)*
pow(v,3)*br1*RC3+12*pow(b1,2)*pow(q1,2)*pow(v,3)*VVC3-48*br
*q2*pow(v,3)*RC4+16*v*pow(b1,2)*VC2*f-8*pow(v,2)*av*VC2*f+
16*v*av*C2*f-8*pow(v,2)*ar*RC2*f+8*pow(v,2)*C2*r*f-4*pow(
v,2)*pow(b1,2)*VVC2*f+8*b1*q1*pow(v,3)*VC2*r-4*ar*pow(v,3)*
br1*q2*RC2-72*v*b1*q1*C3*f-12*pow(v,4)*b1*q1*VC3+72*pow(b1,
3)*q1*C2-48*b1*q1*VC4*pow(v,3)-56*pow(b1,3)*q1*v*VC2-60*po
w(b1,2)*pow(q1,2)*pow(v,2)*VC3+144*b1*q1*C4*pow(v,2)+108*po
w(b1,2)*pow(q1,2)*v*C3-12*pow(v,4)*br*q2*RC3-12*pow(v,3)*C3
*r+42*pow(b1,2)*v*C3+2*pow(v,4)*pow(b1,2)*VVC2+2*pow(r,2)*
pow(v,3)*C1-24*av*pow(v,2)*C3+12*av*pow(v,3)*VC3+4*pow(v,4)
*av*VC2-4*pow(v,3)*av*C2+6*pow(br,2)*pow(v,3)*RRC3-4*pow(
br,2)*pow(v,2)*RC2-24*pow(b1,2)*VC3*pow(v,2)+12*ar*pow(v,3)
*RC3-16*pow(z,2)*pow(v,3)*b1*q1*VC4+64*pow(z,2)*pow(v,2)*
b1*q1*C4+3*pow(v,5)*C3+24*pow(v,4)*C4)/pow(v,3);
}

static void price_barrier_put(double y, double S, double K,
    double t, double v, double r, double d, double kv,
    double vbar, double sigmav, double kr, double rbar, double sigma
r, double q1, double q2, double q3, double f, double z,
    double b1, double av, double av1, double av2, double q, double
    ar, double ar1, double br, double br1, double br2, double *
    result)
{
    double /*f, z, b1, av, av1, av2, q, ar, ar1, br, br1, br
        2, */n, N, n0, N0, C1, VC1, C2, VC2, RC2, VVC2, C3, VC3,
        RC3, VVC3, VRC3, RRC3, C4, VC4, RC4, C5, Pn1, PN1, Pn2, PN2
        , Pn3, PN3, Pn4, PN4, Pn5, PN5 ;

    N0=cdf_nor(y);
    n0=pnl_normal_density(y);

    fC1(N0, n0, K, y, v, & C1);
    fVC1(N0, n0, K, y, & VC1);
    fC2(N0, n0, C1, K, v, f, y, b1, q1, & C2);
    fVC2(N0, n0, C1, K, y, v, & VC2);

```



```

fRC2(N0, n0, C1, y, v, & RC2);
fVVC2(N0, n0, C1, K, v, y, & VVC2);
fC3(N0, n0, C1, C2, VC2, RC2, b1, br, q1, q2, av, K, v,
    y, r, f, & C3);
fVC3(N0, n0, C1, C2, VC2, VVC2, RC2, b1, br, q1, q2, av,
    av1, K, v, y, r, f, & VC3 );
fRC3(N0, n0, C1, C2, RC2, b1, br1, q1, q2, K, v,
    y, r, f, & RC3 );
fVVC3( N0, n0, C1, C2, VC2, VVC2, b1, q1, av2,
    K, v, y, r, f, & VVC3);
fVRC3(N0, n0, C1, C2, RC2, b1, br1, q1, q2, K, v, y, r,
    f, & VRC3);
fRRC3(N0, n0, C1, RC2, br2, q2, K, v, y, r, f, & RRC3);
fC4(N0, n0, C1, C2, C3, VC2, VC3, VVC2, RC2, RC3, b1, br
    , br1, q1, q2, av, av1, ar, K, v, y, r, f, & C4);
fRC4(N0, n0, C1, C2, C3, VC2, VC3, VVC2, RC2, RC3, b1,
    br, br1, q1, q2, av, av1, ar, K, v, y, r, f, & RC4);
fVC4(N0, n0, C1, C2, C3, VC2, VC3, VVC2, VVC3, VRC3, RC2
    , RC3, b1, br, br1, q1, q2, av, av1, av2, ar, K, v, y, r,
    f, & VC4);
fC5(N0, n0, C1, C2, C3, C4, VC2, VC3, VC4, VVC2, VVC3,
    VRC3, RC2, RC3, RC4, RRC3, b1, br, br1, br2, q1, q2, q3, av
    , av1, av2, ar, ar1, K, v, y, r, f, & C5);

fPn1(C1, & Pn1);
fPN1(C1, z, & PN1);
fPn2(C1, C2, v, b1, q1, z, & Pn2);
fPN2(C1, C2, v, z, f, & PN2);
fPn3(C1, C2, C3, VC2, RC2, r, v, z, f, av, b1, br, q1,
    q2, & Pn3);
fPN3(C1, C2, C3, VC2, RC2, z, r, v, f, b1, br, q1, q2, &
    PN3);
fPn4(C1, C2, C3, C4, VC2, VVC2, VC3, RC2, RC3, b1, br,
    br1, q1, q2, av, av1, ar, z, v, y, r, f, & Pn4);
fPN4( C1, C2, C3, C4, VC2, VVC2, VC3, RC2, RC3, b1, br,
    br1, q1, q2, av, av1, ar, z, v, y, r, f, & PN4);
fPn5(C1, C2, C3, C4, C5, VC2, VVC2, VVC3, VC3, VC4, RC2,
    RC3, RC4, RRC3, VRC3, b1, br, br1, br2, q1, q2, q3, av,
    av1, av2, ar, ar1, z, v, y, r, f, & Pn5);
fPN5(C1, C2, C3, C4, C5, VC2, VVC2, VVC3, VC3, VC4, RC2,
    RC3, RC4, RRC3, VRC3, b1, br, br1, br2, q1, q2, q3, av,

```

```

    av1, ar, ar1, z, v, y, r, f, & PN5);
N=cdf_nor(z);
n=pnl_normal_density(z);
*result=(Pn1*n+PN1*N)*sqrt(t)+(Pn2*n+PN2*N)*t+(Pn3*n+PN
3*N)*sqrt(t)*t+(Pn4*n+PN4*N)*pow(t,2)+(Pn5*n+PN5*N)*sqrt(
t)*pow(t,2);
}

static void pricing_american_put_three_factor_model(double
    S, double K, double t, double v, double r, double d,
    double kv, double vbar, double sigmav, double kr, double rbar,
    double sigmar, double q1, double q2, double q3, double result[])
{
    double f, z, b1, av, av1, av2, q=0., ar, ar1, br, br1,
    br2, y, y0, step, price0, price1, price2;
    int dir, search, i;
    fb1(sigmav, & b1);
    fav(kv, vbar, b1, v, & av);
    fav1(kv, vbar, b1, v, & av1);
    fav2(kv, vbar, b1, v, & av2);
    ff(r, d, & f);
    far(kr, rbar, r, & ar);
    far1(kr, & ar1);
    fbr(sigmar, r, & br);
    fbr1(sigmar, r, & br1);
    fbr2(sigmar, r, & br2);

    theta(S, K, t, v, & z);

    fy0(S, K, t, v, & y0);
    step=1;
    y=y0;
    price0=K-S;
    while (step>=0.001)
    {
        dir=1;
        search=1;
        i=0;
        while (search==1 && i<=1000)
        {

```

```

        i=i+1;
        price_barrier_put(y+dir*step, S, K, t, v, r, d, kv
        , vbar, sigmav, kr, rbar, sigmar, q1, q2, q3, f, z, b1,
        av, av1, av2, q, ar, ar1, br, br1, br2, & price1);
        if (price1>price0)
        {
            y=y+dir*step;
            result[3]=y;
            price0=price1;
        }
        else
        {search=0;}
    }
    step=step/10;
    price_barrier_put(y-step, S, K, t, v, r, d, kv, vbar
    , sigmav, kr, rbar, sigmar, q1, q2, q3, f, z, b1, av,
    av1, av2, q, ar, ar1, br, br1, br2, & price1);
    price_barrier_put(y+step, S, K, t, v, r, d, kv, vbar
    , sigmav, kr, rbar, sigmar, q1, q2, q3, f, z, b1, av,
    av1, av2, q, ar, ar1, br, br1, br2, & price2);
    if (price2>price1) dir=1;
    else dir=-1;
}
result[3]=y;// Exercised Bound
result[0]=price0;// American put price
price_barrier_put(10000, S, K, t, v, r, d, kv, vbar, si
    gmav, kr, rbar, sigmar, q1, q2, q3, f, z, b1, av, av1, av
    2, q, ar, ar1, br, br1, br2, & price2);
result[1]=price2; //European put price
result[2]=result[0]-result[1];// Early Exercise premium
}

int ApMedvedevScaillet(int am,double spot,NumFunc_1 *p,
    double maturity,double interest, double kr,double rbar, double
    sigmar,double v0,double kv,double vbar,double sigmav,
    double rho12,double rho13,double rho23,double *ptprice,double *
    ptdelta)
{
    double strike,price,delta,spot_inc,volatility;
    double Result[4];
    double inc=0.0001;

```

```

double dividend=0.;
volatility=sqrt(v0);
strike=p->Par[0].Val.V_PDOUBLE;

pricing_american_put_three_factor_model(spot, strike,
    maturity, volatility, interest, dividend, kv, vbar, sigmav,
    kr, rbar, sigmar, rho12, rho13, rho23, Result);

if(am)
    price=Result[0];
else price=Result[1];
//Price
*ptprice=price;

//Delta
spot_inc=spot*(1+inc);
pricing_american_put_three_factor_model(spot_inc, strike,
    maturity, volatility, interest, dividend, kv, vbar, si
    gmav, kr, rbar, sigmar, rho12, rho13, rho23, Result);
if(am)
    delta=(Result[0]-price)*(spot*inc);
else
    delta=(Result[1]-price)*(spot*inc);

*ptdelta=delta;

return OK;
}

int CALC(AP_MedvedevScaillet)(void *Opt, void *Mod, Pricing
    Method *Met)
{
    TYPEOPT* ptOpt=(TYPEOPT*)Opt;
    TYPEMOD* ptMod=(TYPEMOD*)Mod;

return ApMedvedevScaillet(ptOpt->EuOrAm.Val.V_BOOL,ptMod->
    SO.Val.V_PDOUBLE,
                                ptOpt->PayOff.Val.V_NUMFUNC_1,
                                ptOpt->Maturity.Val.V_DATE-ptMod->

```

```

>T.Val.V_DATE,

    ptMod->r0.Val.V_PDOUBLE
    ,ptMod->kr.Val.V_PDOUBLE,
    ptMod->thetar.Val.V_PDOUBLE,
    ptMod->Sigmar.Val.V_PDOUBLE,
    ptMod->V0.Val.V_PDOUBLE
    ,ptMod->kV.Val.V_PDOUBLE,
    ptMod->thetaV.Val.V_PDOUBLE,
    ptMod->SigmaV.Val.V_PDOUBLE,
    ptMod->RhoSr.Val.V_RGDOUBLE,
    ptMod->RhoSV.Val.V_RGDOUBLE,
    ptMod->RhorV.Val.V_RGDOUBLE,
    &(Met->Res[0].Val.V_DOUBLE),
    &(Met->Res[1].Val.V_DOUBLE));
}

static int CHK_OPT(AP_MedvedevScaillet)(void *Opt, void *
    Mod)
{
    if ((strcmp( ((Option*)Opt)->Name,"PutAmer")==0)|| (strcmp
        ( ((Option*)Opt)->Name,"PutEuro")==0))
        return OK;
    return WRONG;
}

#endif //PremiaCurrentVersion
static int MET(Init)(PricingMethod *Met,Option *Opt)
{
    if ( Met->init == 0)
    {
        Met->init=1;
    }

    return OK;
}

PricingMethod MET(AP_MedvedevScaillet)=
{
    "AP_MedvedevScaillet",
    {" " ,PREMIA_NULLTYPE,{0},FORBID}},
    CALC(AP_MedvedevScaillet),

```

```
{{"Price",DOUBLE,{100},FORBID},
 {"Delta",DOUBLE,{100},FORBID} ,
 {" ",PREMIA_NULLTYPE,{0},FORBID}},
CHK_OPT(AP\_MedvedevScaillet),
CHK_ok,
MET(Init)
};
```

References