Here are the key lines from my input file. It’s not too hard to get this problem to converge.

```
REAC  H2  .148
REAC  CO  .148
REAC  O2  .148
REAC  N2  .556
/
PROD  CO2  .2
PROD  H2O  .2
PROD  N2  .6
/
INTM  O  0.01
INTM  OH  .001
INTM  H  .01
INTM  HCO  .01
```

This gives the plots on page 2. I didn’t plot N2, but that’s pretty much constant at 65 percent.

The first question asks for diffusion velocities. We use the Fick’s relation which requires local values of \( Y_i, \nabla Y_i, \) and \( D_i \). First thing is calculating mass fractions from mole fractions which requires calculation of local mixture molecular weight, then mass fraction, and using central differencing to get the derivatives of each mole fraction locally. For diffusion coefficients, you need to calculate every binary diffusion coefficient for every pair at each location. I do this with a 3 dimensional array. Then, for each species, we use the binary coefficients to calculate mixture averaged coefficients. From that, we get diffusion velocities shown on page 3. See my program attached at the end of this document for details.

Key points are that these follow the gradients and are generally much larger than the bulk velocity for many species, especially H.
Syngas $H_2:CO$ 1:1, stoichiometric in 0.5 atm air
$3 \times 10^{-3}$ g/cm$^2$-s

Distance (cm)

Mole Fraction

Syngas $H_2:CO$ 1:1, stoichiometric in 0.5 atm air
$3 \times 10^{-3}$ g/cm$^2$-s

Distance (cm)

Mole Fraction
For hydrogen peroxide, I took the reactions below from the mechanism. Some have H₂O₂ as a reactant, some as a product, so we have to watch signs. For Reaction 1) I used the low pressure limit, and I also calculated ROP with Alt 1), which is the expression I suggested in the assignment. Both are evaluated on the attached spreadsheet.

1) H₂O₂(+M)=OH+OH(+M)  2.00E+12   0.900   48750.00
   LOW/2.490E+24 -2.300   4.875E+04/
   TROE/0.43 1.0E-30 1.0E+30 1.0E+30/
   H₂/3.7/ CO/2.8/ CO₂/1.6/ O₂/1.2/ HE/0.65/ H₂O/6.0/ H₂O₂/7.7/ N₂/1.50/
2) HO₂+H₂O₂=O₂+H₂O₂     1.03E+14   0.000   11051.
3) HO₂+HO₂=O₂+H₂O₂        1.94E+11   0.000  -1410.2
4) H₂O₂+H=HO₂+H₂          2.15E+10   1.000   6000.
5) H₂O₂+H=OH+H₂O          1.02E+13   0.000   3577.
6) H₂O₂+OH=HO₂+H₂O        7.586E+13  0.000   7269.
7) H₂O₂+OH=HO₂+H₂O        1.738E+12  0.000   318.
8) H₂O₂+O=OH+HO₂           9.630E+06  2.000   3970.
9) HCO+HO₂=H₂O₂+CO         4.000E+11  0.000   0.
Alt 1) H₂O₂+M=OH+OH+M      1.20E+17  0.000   45500.

Key steps were to get CHEMKIN data for all species, then extract Gibbs energies for each point for each species. Use that data to get K_c for each reaction at every point. Then we can use the mole fraction data to get reaction ROPs by taking forward rate times product of reactant concentrations minus backward rate (k_f/K_c) times product of product concentrations.

You’re essentially doing part d. in the process of doing a., so results for d. and a. are shown on page 5.

Mass production rate from the continuity equation can be obtained using just solution data, and I added that in to my program for part 2). Results are shown in the right panel of page 5. The data are similar but not exactly the same. Pretty close for the most part. I expected better agreement, but this is not that bad. Ideally, they should be identical.

Part c and d involve analysis of these plots. Basically H₂O₂ is made right away from HO₂ which is made from H attack on O₂. Reactions 2 & 3 combine to form most the H₂O₂ right at the burner, causing peak concentration. There’s some formation early due to OH recombination, but that kicks in more later. Primary consumption throughout the system is attack of H₂O₂ by O, H, and OH. As HO₂ decays, H₂O₂ production decays, and consumption wins. Around 0.1 cm, OH has finished attacking fuels, and now starts recombining in earnest, producing a second wave of H₂O₂ formation that gives the kink in the H₂O₂ concentration curve. The velocity curve follows mostly the derivatives of the mole fraction curve.
FORTRAN CODE TO GET LOCAL DIFFUSION COEFFICIENTS, VELOCITIES, & MOLAR PRODUCTION RATES

dimension dist(100), u(100), t(100), rho(100), x(100,100),
v(100,100), d(100,100), y(100,100), e(100),
sig(100), xmt(100), xmmix(100), db(100,100,100),
ZZ(100,100)
open(31,file='output.txt')
open(32,file='data.txt')
open(33,file='tran.txt')

Set these values for a simulation. Would have to change the
read 31 line as well to change number of species, and also
the output write lines and formatting
P = 0.5
npts = 61
nspec = 12
xmdot = 3e-3

c
Read in data from customized output file. Cleaned up variant
c of Premix.out format.
c
do 10 i = 1,npts
   read(31,*) dist(i),t(i),u(i),rho(i),x(1,i),x(2,i),
   x(3,i),x(4,i),x(5,i),x(6,i),x(7,i),x(8,i),
   x(9,i),x(10,i),x(11,i),x(12,i)
10 continue

c
Read in transport data in order with 3rd column as mwt
c
do 20 j = 1,nspec
   read(33,*) e(j), sig(j), xmwt(j)
20 continue

do 50 i = 1,npts
   xmmix(i) = 0
   do 30 j = 1,nspec
      xmmix(i) = xmmix(i) + xmwt(j)*x(j,i)
   30 continue
   do 40 j = 1,nspec
      y(j,i) = xmwt(j)*x(j,i)/xmmix(i)
40 continue
50 continue

c
Calculate local mixav diffusion coeffs. i = spatial
j = primary species, k = secondary species

do 90 i = 1,npts
   do 80 j = 1,nspec
      do 60 k = 1,nspec
         sigjk = 0.5*(sig(j)+sig(k))
ejk = sqrt(e(j)*e(k))
tstar = t(i)/ejk
omega = 1.0548*tstar**(-.15504) +
   (tstar*.55909)**(-2.1705)
xmu = (xmwt(j)+xmwt(k))/(xmwt(j)*xmwt(k))
db(j,k,i) = 1.8583e-3 * sqrt(t(i)**3 * xmu)
   / (P * sigjk**2 * omega)
90 continue

d(1,i) = (1-y(1,i))/zsum
80 continue
90 continue

c
Calculate diffusion velocities for all but the last point
c
do 110 i = 1,npts-1
   do 100 j = 1,nspec
      if (i .eq. 1) then
         delxj = (x(j,2)-x(j,1))/(dist(2)-dist(1))
      else
         delxj = (x(j,i+1)-x(j,i-1))/
               (dist(i+1)-dist(i-1))
      end if
   100 continue
   v(j,i) = -d(j,i)*delxj/x(j,i)
110 continue
c continue

do 120 i = 1, npts-1
   write(32,600) v(1,i),v(2,i),v(3,i),v(4,i),v(5,i),
                   v(6,i),v(7,i),v(8,i),
                   v(9,i),v(10,i),v(11,i),v(12,i)
   continue
write(32,*)
c

c Calculate local molar production rates based on
species continuity equation

do 140 i = 1, npts-1
   do 130 j = 1, nspec
      if (i .eq. 1) then
         dx = dist(2) - dist(1)
         ZZ(j,i) = xmdot*(y(j,2) - y(j,1))/dx +
                   (rho(2)*y(j,2)*v(j,2) - rho(1))
                   *(y(j,1)*v(j,1))/dx
      else
         dx = dist(i+1) - dist(i-1)
         ZZ(j,i) = xmdot*(y(j,i+1) - y(j,i-1))/dx +
                   (rho(i+1)*y(j,i+1)*v(j,i+1) - rho(i-1))
                   *(y(j,i-1)*v(j,i-1))/dx
      end if
continue
write(32,601) ZZ(1,i),ZZ(2,i),ZZ(3,i),ZZ(4,i),ZZ(5,i),
                   ZZ(6,i),ZZ(7,i),ZZ(8,i),
                   ZZ(9,i),ZZ(10,i),ZZ(11,i),ZZ(12,i)
140 continue

c 600 format(12f9.3)
601 format(12E)
write(32,*)
stop