1 Ideal gas mixer

1.1 Utility functions

For emptying the input pipe I into the furnace

```
fillFurnace:
yield
s Furnace SettingInput 100
l x IAnalyzer TotalMoles
brgtz -3
s Furnace SettingInput 0
j ra
```

And for emptying the furnace into the filtration system, which filters all CO_2 back into the input pipe.

```
emptyFurnace:
yield
s Furnace SettingOutput 100
l x Furnace TotalMoles
brgtz -3
s Furnace SettingOutput 0
j ra
```

1.2 Mixing algorithm

Let $n_R = 0$ be the moles to remove, and calculate constant terms.

$$e_T = t_T n_T$$
$$n_{T-F} = n_T - n_F$$
$$t_{H-C} = t_H - t_C$$

```
move nR 0
mul eT tT nT
sub dn nT nF
sub dt tH tC
```

1.3 Calculate input moles $n_I = n_H + n_C$ given n_R

Calculate total moles input

$$n_I = n_{T-F} + n_R$$

```
calculateInputMoles:
add nI dn nR
```

Calculate temperature input:

$$t_I = \frac{e_T - t_F(n_F - n_R)}{n_I}$$

```
sub x nF nR
mul x tF x
sub tI eT x
div tI tI nI
```

Calculate moles from hot/cold source:

$$n_H = \frac{n_I(t_I - t_C)}{t_{H-C}}, \quad n_C = n_I - n_H$$

```
sub nH tI tC
mul nH nI nH
div nH nH dt
sub nC nI nH
j ra
```

1.4 Searching for n_R

For n_R to be valid, n_H and n_C must both be positive.

```
checkInputMoles:
min x nH nC
sgtz x x # x = nH > 0 and nC > 0
j ra
```

For $n_R = 0$, if this is the case we are done.

```
move nR 0
jal calculateInputMoles
jal checkInputMoles
beq x 1 compose # nR=0, nH>0, nC>0: skip
```

Else, we begin a binary search of the domain $[0, n_F]$ for a satisfactory value of n_R, each time recalculating and rechecking the input moles via ??. Beginning with $n_R = {}^{n_F}/_2$, we bisect i times.

```
define BisectionIterations 20
# ...
move i BisectionIterations
div nR nF 2
div step nF 4
jal calculateInputMoles
jal checkInputMoles
select x x 1 -1
mul x x step
add nR nR x # nR += (nH>0 && nC>0) ? step : -step
div step step 2
sub i i 1
brgtz i -7
```

1.5 Removing n_R and composing I

Having found satisfactory n_R, n_H and n_C, we first remove n_R moles from the furnace (or remove none when n_R is zero). The furnace, containing only CO_2, is emptied into the input pipe I. Moles are removed, via a volume pump, until the total moles in the pipe n_I is less-than or equal to the initial number of moles n_F minus the number of moles to remove n_R. Then this potentially reduced amount is reinserted into the furnace.

```
jal emptyFurnace
sub nF nF nR
yield
s InputDumpPump On 1
l x IAnalyzer TotalMoles
brgt x nF -3
s InputDumpPump On 0
jal fillFurnace
```

We then compose the I mixture by first adding H moles via a volume pump while the moles in I is less-than to n_H, then doing the same for C until moles in I is less-than to n_I.

```
alias HPump d0
alias CPump d1
alias HAnalyzer d2
alias CAnalyzer d3
```

```
alias nH r0
alias hC r1
alias i r2

move i 0

fillUntil:
yield
s dr2 On 1
add i i 2
l x dr3 TotalMoles
sub i i 2
brlt x rr2 - 3
s dr0 On 0
add i i 1
```

$$
\begin{aligned}
n_T &= n_F - n_R + n_T \\
\Rightarrow n_I &= n_T - n_F + n_R
\end{aligned}
$$

$$
\begin{aligned}
t_T n_T &= t_F(n_F - n_R) - t_I n_I \\
&= t_F(n_F - n_R) - t_I(n_T - n_F + n_R) \\
\Rightarrow f(t_I) = n_R &= n_F - \frac{n_T(t_T + t_I)}{t_F + t_I}
\end{aligned}
$$