In [ ]:

```python
#all my fave imports in case I need them later
import numpy as np
from scipy.optimize import *
from scipy.integrate import *
import matplotlib.pyplot as plt
import math
import warnings
warnings.filterwarnings('ignore') #living dangerously
```

Problem 2A.)

Starting from the general case, accumulation=generation-consumption+convective mass transfer.

Using elements from class notes (`problem2_referenceNotes.pdf`) and the Knauer model,

Accumulation=0

Generation= $k_r R_s + q$

Consumption = $k_f R_s L_c(z)$

Convective Mass Transfer = ${}^{1n}k_m(z)[L_b - L_c(z)]$ (where the expression given in the problem is divided by n_c to make the units work)

Algebraically rearainging the expression occurs as follows:

$0 = k_r R_{*s} + q - k_f R_s L_c(z) + {}^{1n}{}_c k_m(z)[L_b - L_c(z)]$

$[k_f R_s + {}^{k_m(z)n_c}]L_c(z) = k_r R_{*s} + q + {}^{k_m(z)L_b n_c}$

$L_c(z) = k_r R_{*s} + q + {}^{k_m(z)L_b n_c} k_f R_s + {}^{k_m(z)n_c}$

Problem 2B.)

Given $L_c(z) = k_r R_{*s} + q + {}^{k_m(z)L_b n_c} k_f R_s + {}^{k_m(z)n_c}$ as found in part A, we examine two limiting cases.

First, in the case $k_m << 1$, we see that the expression becomes simply:

$L_c(z) = k_r R_{*s} + q k_f R_s$

This is to say that when the mass transfer coefficient, k_m, is very small, there a little to no effects of the bulk concentration. Essentially, the convective term is disregarded, and the concentration is solely depenedent on the binding and 'unbinding' or the secretion rate.

Alternatively in the $k_m >> 1$ case, the expression becomes:

$L_c(z) = {}^{k_m(z)L_b n_c}{}^{k_m(z)n_c} = L_b$

In this case, the concentration is solely based off of the bulk concentration as convective mass transfer is so fast that it outweighs any affect of binding or secretion.

Problem 2C.)

Starting with the expression from a, but with $L_b = 0$,

$L_c(z) = k_r R_{*s} + q k_f R_s + {}^{k_m(z)n_c}$

and the following expression $R_{*s} = {}^{K_{ss}L_c}1 + {}^{K_{ss}L_c}({}^{V_s}k_{*e})$ (from `problem2_referenceNotes.pdf` or Knauer et all (1984)).

In the limit where $K_{ss}L << 1$, the expression simplifies to $R_{*s} = K_{ss}L_c({}^{V_s}k_{*e})$.

Next, the value for R_s is substituted for using a mass balance from the aforementioned notes, at steady state:

$$0 = k_f L_c R_s - k_r R_{*s} - k_{*e} R_{*s}$$

$$R_s = (k_r + k_{*e}) R_{*s} k_f L_c$$

Substituting this into our previous equation for $L_c(z)$ we obtain:

$$L_c(z) = k_r R_{*s} + q k_f (k_r + k_{*e}) R_{*s} k_f L_c + k_m(z) n_c$$

Isolating this expression for L_c produces the following:

$$L_c(z)[(k_r + k_{*e}) R_{*s} L_c + k_m(z) n_c] = k_r R_{*s} + q$$

$$(k_r + k_{*e}) R_{*s} + L_c k_m(z) n_c = k_r R_{*s} + q$$

$$L_c k_m(z) n_c = q - k_{*e} R_{*s}$$

$$L_c = [q - k_{*e} R_{*s}] n_c k_m(z)$$

Next, substituting this expression for L_c into R_{*s} gives:

$$R_{*s} = K_{ss}[q - k_{*e} R_{*s}] n_c k_m(z)(V_s k_{*e})$$

Now, isolating R_{*s},

$$R_{*s}[1 + k_{*e} n_c k_m(z)(V_s k_{*e}) K_{ss}] = K_{ss} q n_c k_m(z)(V_s k_{*e})$$

$$R_{*s} = K_{ss} q n_c k_m(z)(V_s k_{*e}) 1 + k_{*e} K_{ss} n_c k_m(z)(V_s k_{*e})$$

Given that, from the `problem2_referenceNotes.pdf`, $R_{*i} = k_{*c} k_{deg} R_{*s}$, and that $R_{*total} = R_{*s} + R_{*i}$, the following expression is constructed for R_{*total}:

$$R_{*i} = K_{ss} q n_c k_m(z)(V_s k_{*e}) 1 + K_{ss} n_c k_m(z)(V_s)(k_{*e} k_{deg})$$

$$R_{*total} = [1 k_{*e} + 1 k_{deg}] K_{ss} q V_s n_c k_m(z) 1 + K_{ss} V_s n_c k_m(z)$$

Where $K_{ss} = k_{*e} k_f k_e (k_r + k_{*e})$

Problem 2D.)

To plot R_{*total}, the Sherwood number is substituted into the equation for $k_m(z)$.

$$R_{*total} = [1 k_{*e} + 1 k_{deg}] K_{ss} q V_s n_c k_m(z) 1 + K_{ss} V_s n_c k_m(z)$$

$$R_{*total} = [1 k_{*e} + 1 k_{deg}] K_{ss} q V_s n_c k_m(z) + K_{ss} V_s n_c$$

$$k_m(z) = \dot{\gamma}^{1/3} D_{2/3L} z^{1/3}$$

In [2]:

```
#constants
ke=0.001; kestar=0.005; kf=5.14E-21; kr=0.025; kdeg=0.0008;Vs=10;
q=1000; nc=3.0E8; gamma=100; Dl=1.0E-10;

Kss=kestar*kf/(ke*(kr+kestar))

z=np.linspace(0.001,1,1000)
km=np.power(gamma,1/3)*np.power(Dl,2/3)*np.power(z,1/3)
```

```
RtotalStar=(1/kestar + 1/kdeg)*Kss*q*Vs/(km+(Kss*Vs*nc))

RtotalStarMax=np.amax(RtotalStar)   #maximum value of RtotalStar
RtotalStarNorm=RtotalStar/RtotalStarMax   #normalized

plt.plot(z,RtotalStarNorm);
plt.ylabel('Normalized Mitotic Activity');
plt.xlabel('z');
```