9.9

(a)

$$
A = 1.2\gamma \begin{bmatrix} 0 & 0.2 & 0.1 \\ .05 & 0 & .05 \\ 0.1 & \frac{1}{30} & 0 \end{bmatrix}
$$

$$
b = 0.012\gamma \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}
$$

递推

$$
p(t+1) = Ap(t) + b
$$

得到

$$
p(t) = A^t p(0) + \sum_{i=0}^{t-1} A^i b
$$

所以收敛情形和A的特征值有关。

当$\gamma = 3$时，特征值的绝对值都小于1，所以无条件收敛；当$\gamma = 5$时，有一个特征值的绝对值大于1，所以可能会发散。

计算代码如下：

```
import numpy as np

A = np.array([
        [0, 0.2, 0.1],
        [0.05, 0, 0.05],
        [0.1, 1/30, 0]])
res = np.linalg.eigvals(A)

#gamma=3
print(res * 1.2 * 3)

#gamma=5
print(res * 1.2 * 5)
```

```
[ 0.60848571 -0.36       -0.24848571]
[ 1.01414284 -0.6        -0.41414284]
```

(b)回顾

$$
A = \alpha\gamma\left(\Lambda^{-1} G - I_n\right)
$$

假设$\Lambda^{-1}G-I_n$的特征值为λ_i，那么A的特征值为$\alpha\gamma\lambda_i$，记绝对值最大的λ_i为$\lambda_{\max}$，由之前讨论可得，

$$\gamma_{\text{crit}}=\frac{1}{|\alpha\lambda_{\max}|}$$

10.5

(a)设λ_i对应的特征向量为v_i，那么

$$A^n v_i=\lambda_i^n v_i$$

因此

$$\begin{aligned}
e^A v_i&=\sum_{n=0}^{\infty}\frac{A^n}{n!}v_i\\
&=\sum_{n=0}^{\infty}\frac{\lambda_i^n}{n!}v_i\\
&=e^{\lambda_i}v_i
\end{aligned}$$

所以e^A的特征值为λ_i

(b)由(a)可得

$$\begin{aligned}
\det e^A&=\prod_{i=1}^{n}e^{\lambda_i}\\
&=e^{\sum_{i=1}\lambda_i}\\
&=e^{\mathbf{Tr}A}
\end{aligned}$$

10.6

方程的解为

$$x(t)=e^{At}x(0)$$

通过程序计算得到e^A,e^{2A}:

```
A = np.array([
        [0.5, 1.4],
        [-0.7, 0.5]])

print(expm(A))
```

```
[[ 0.90470626  1.94925032]
 [-0.97462516  0.90470626]]
```

```
print(expm(2 * A))
```

```
[[-1.081295    3.52699794]
 [-1.76349897 -1.081295  ]]
```

假设

$$x(0)=\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

那么由条件可得

$$\begin{aligned}
&x_1>0\\
&x_2<0\\
&0.90470626x_1+1.94925032x_2>0\\
&-0.97462516x_1+0.90470626x_2<0
\end{aligned}$$

目标是判断以下两个式子的符号

$$\begin{aligned}
&-1.081295x_1+3.52699794x_2\\
&-1.76349897x_1-1.081295x_2
\end{aligned}$$

通过画图不难得出

$$\begin{aligned}
y_1(3)&=-1\\
y_2(3)&=-1
\end{aligned}$$

10.8

(a)

$$\det(\lambda I-A)=\det(\lambda I^T-A^T)=\det(\lambda I-A^T)$$

(b)

$$\det A=\prod_{i=1}^{n}\lambda_i$$

(c)$\forall\lambda_i$对应的特征向量v_i，那么

$$\begin{aligned}
Av_i&=\lambda_i v_i\\
A^{-1}v_i&=\frac{1}{\lambda_i}v_i
\end{aligned}$$

(d)

$$
\begin{aligned}
\det(\lambda I - T^{-1}AT) &= \det(T^{-1}\lambda IT - T^{-1}AT) \\
&= \det(T^{-1})\det(\lambda I - A)\det(T) \\
&= \frac{1}{\det(T)}\det(\lambda I - A)\det(T) \\
&= \det(\lambda I - A)
\end{aligned}
$$

10.14

(a)计算特征值即可

```
% (a)
eig(A)
```

```
ans =
  -0.1000 + 5.0000i
  -0.1000 - 5.0000i
  -0.1500 + 7.0000i
  -0.1500 - 7.0000i
```

因为实部都是负数，所以稳定

(b)

```
%(b)
sys = ss(A, [], [], []);
x0 = [1; 1; 1; 1];
[y,t,x] = initial(sys, x0);
for i = 1: 4
    figure(i);
    plot(x(:, i));
end

x0 = rand(4, 1);
[y,t,x] = initial(sys, x0);
for i = 1: 4
    figure(i + 4);
    plot(x(:, i));
end
```

(c)

```
%(c)
expm(15 * A)
```

```
ans =
    0.2032   -0.0068   -0.0552   -0.0708
    0.0340    0.0005   -0.0535    0.1069
    0.0173    0.1227    0.0270    0.0616
    0.0815    0.0186    0.1151    0.1298
```

(d)

```
%(d)
expm(- 20 * A)
```

```
ans =
    6.2557    3.3818    1.7034    2.2064
   -2.1630   -2.8107  -14.2950   12.1503
   -3.3972   17.3931   -1.6257   -2.8004
   -1.7269   -6.5353   10.7081    2.9736
```

(e)Z的元素较小，Y的元素较大

(f)使用下式子计算即可

$$x(0) = e^{-10A}\begin{bmatrix}1\\1\\1\\1\end{bmatrix}$$

对应代码为

```
%(e)
x10 = [1; 1; 1; 1];
expm(10 * A) \ x10
```

```
ans =
    3.9961
    1.0650
    3.8114
    1.7021
```

11.3

假设A可对角化，即

$$A = T\Lambda T^{-1}, \Lambda = \mathrm{diag}\{\lambda_1, \ldots, \lambda_n\}$$

那么

$$
\begin{aligned}
\lim_{k\to\infty}(I+A/k)^k &= \lim_{k\to\infty}(I+T\Lambda T^{-1}/k)^k \\
&= \lim_{k\to\infty}\left(T(I+\Lambda/k)T^{-1}\right)^k \\
&= T\left(\lim_{k\to\infty}(I+\Lambda/k)^k\right)T^{-1} \\
&= T\left(\mathrm{diag}\{1+\lambda_1/k,\ldots,1+\lambda_n/k\}^k\right)T^{-1} \\
&= T\mathrm{diag}\{e^{\lambda_1},\ldots,e^{\lambda_n}\}T^{-1} \\
&= e^A
\end{aligned}
$$

11.6a

转移规则对应的矩阵为

$$
A=\begin{bmatrix}
0 & 0 & 1 & 0 & 1\\
1 & 1 & 0 & 1 & 0\\
1 & 0 & 0 & 0 & 1\\
0 & 0 & 0 & 1 & 0\\
0 & 1 & 0 & 1 & 0
\end{bmatrix}
$$

记$B^{(k)}=A^k$，那么$B_{ij}^{(k)}$表示以j开头，i结尾的长度为$k+1$的语言数量，所以

$$
K_N=\sum_i\sum_j B_{ij}^{(N-1)}=1_n^TA^{N-1}1_n
$$

接着计算A^k即可，注意此时A的特征值不相同，所以A相似于对角阵，即

$$
A=T\Lambda T^{-1}=\sum_{i=1}^n\lambda_i v_i w_i^T
$$

那么

$$
A^{N-1}=T\Lambda^{N-1}T^{-1}=\sum_{i=1}^n\lambda_i^{N-1}v_i w_i^T
$$

假设$\lambda_1=\max\{|\lambda_i|\}$，那么

$$
\begin{aligned}
A^{N-1}&=\lambda_1^{N-1}\sum_{i=1}^n\left(\frac{\lambda_i}{\lambda_1}\right)^{N-1}v_i w_i^T\\
&\approx\lambda_1^{N-1}v_1w_1^T\\
K_N&=1_n^TA^{N-1}1_n\\
&\approx\lambda_1^{N-1}(1_n^Tv_1)(w_1^T1_n)
\end{aligned}
$$

因此

$$
\begin{aligned}
R &= \lim_{N\to\infty} \frac{\log_2 K_N}{N} \\
&= \lim_{N\to\infty} \frac{(N-1)\log_2(\lambda_1) + \log_2((1_n^T v_1)(w_1^T 1_n))}{N} \\
&= \log_2 \lambda_1
\end{aligned}
$$

利用计算机得到结果为

```
0.8113704627516485
```

对应代码为

```
import numpy as np

A = np.array([
        [0, 0, 1, 0, 1],
        [1, 1, 0, 1, 0],
        [1, 0, 0, 0, 1],
        [0, 0, 0, 1, 0],
        [0, 1, 0, 1, 0]], dtype=np.float64)

res = np.linalg.eigvals(A)
print(np.log2(np.max(np.abs(res))))
```

对于需要比较的情形，此时

$$
A = 1_n 1_n^T
$$

注意到

$$
\begin{aligned}
A^k &= 1_n(1_n^T 1_n)1_n^T A^{k-2} \\
&= n 1_n 1_n^T A^{k-2} \\
&= n A^{k-1} \\
&= \dots \\
&= n^{k-1} A
\end{aligned}
$$

因此

$$
K_N = \sum_i \sum_j B_{ij}^{(N-1)} = 1_n^T A^{N-1} 1_n = n^{N-2} 1_n^T A 1_n = n^{N-1}
$$

因此

$$
\begin{aligned}
R &= \lim_{N \to \infty} \frac{\log_2 K_N}{N} \\
&= \lim_{N \to \infty} \frac{\log_2 n^{N-1}}{N} \\
&= \log_2 n \\
&= \log_2 5 \\
&\approx 2.321928094887362
\end{aligned}
$$