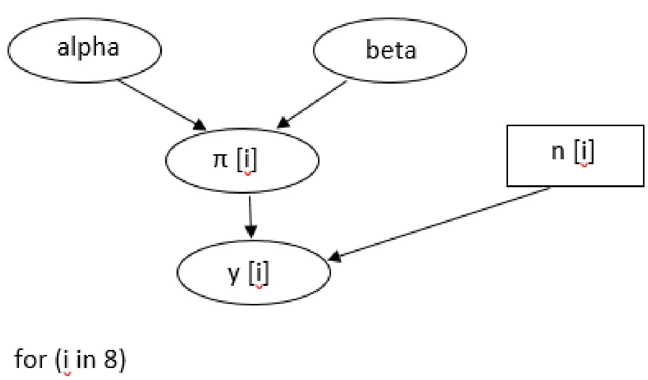
Problem 8.3:

```
tausq <- 3 * tausq0
mu ~ dnorm( -2.75, tausq0)

model
{
# likelihood
for (i in 1:N) {
  y[i] ~ dnorm( mu, tausq )
}
# priors
tausq0 <- 3 * tausq
mu ~ dnorm( -2.75, tausq0)
tausq ~ dgamma( 13.3, 5.35)
sigmasq <- 1/tausq
}

#data
list(y=c(-2.526, -1.715, -1.427, -2.12, -2.659, -2.408, -3.219, -1.966,
-2.526, -1.833, -2.813, -1.772, -2.813, -2.526, -3.219, -2.526,
-2.813, -2.526, -3.507, -2.996, -3.912),N=21)

#inits for model 3
list(mu=-5,tausq = 1)
list(mu=-2.5,tausq = 100)
list(mu=0,tausq = 1000)
```

mean	**sd**	**MC_error**		**val2.5pc**	**median**	**val97.5pc**		**start**	**sample**
mu	-2.586	0.1304	0.001018		-2.844	-2.587	-2.329	1001	18000
tausq	2.576	0.5303	0.003576		1.649	2.54	3.72	1001	18000

The output in Sect. 7.1.1 is -2.563, while the OpenBugs output is **-2.586**. Since Markov Chain error is less than 1/20, the rule of thumb is satisfied. Also, the **95% cs for mu is (-2.844, -2.329)**, which is close to (-2.845, -2.280) of 7.1.1.

Problem 9.1:



Problem 9.4:

```
model
{
  for (i in 1: N) {
    theta[i]~dgamma(alpha, beta)
    lambda[i]<-theta[i]*t[i]
    x[i]~dpois(lambda[i])

  }
  alpha~dexp(1)
  beta~dgamma(0.1, 1.0)
  mu<-alpha/beta
  thetanew~dgamma(alpha, beta)
  lambdanew<-thetanew*tnew
  xnew~dpois(lambdanew)
  }

#data
list(x=c(5,1,5,14,3,19,1,1,4,22), t=c(94.3, 15.7, 62.9, 126, 5.24, 31.4, 1.05, 1.05, 2.1, 10.5),
N=10,tnew=1000)

#inits
list(alpha = 1, beta = 1)
list(alpha = 10, beta = 10)
```

	mean	sd	MC_error		val2.5pc	median	val97.5pc		start	sample
alpha	0.6966	0.2741	0.00582	0.2954	0.654	1.353	1001	6000		
beta	0.9312	0.5438	0.01215	0.1867	0.8333	2.296	1001	6000		
mu	0.9398	0.612	0.01068	0.3631	0.8036	2.253	1001	6000		
theta[1]	0.05992	0.02502	3.481E-4		0.02112	0.05627	0.1176	1001	6000	
theta[2]	0.1025	0.08003	0.001095		0.009112		0.08235	0.3132	1001	6000
theta[3]	0.08911	0.03726	4.456E-4		0.03137	0.08406	0.1767	1001	6000	
theta[4]	0.1156	0.03006	3.89E-4	0.06502	0.1132	0.1812	1001	6000		
theta[5]	0.5995	0.3113	0.004031		0.1542	0.5418	1.344	1001	6000	
theta[6]	0.6104	0.138	0.001976		0.3686	0.6004	0.9121	1001	6000	
theta[7]	0.8781	0.7159	0.00949	0.07644	0.6888	2.687	1001	6000		
theta[8]	0.8964	0.7366	0.01054	0.0724	0.712	2.862	1001	6000		
theta[9]	1.588	0.7716	0.01088	0.4722	1.467	3.496	1001	6000		
theta[10]	1.99	0.425	0.005854		1.243	1.965	2.922	1001	6000	

(a). The failure rate mean is **0.9398** with the MC error to be **0.01068**.

(b) & (c).

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
lambdanew	921.5	1475.0	14.75	1.167	439.0	4713.0	1001	9200
thetanew	0.9215	1.475	0.01475	0.001167	0.439	4.713	1001	9200
xnew	920.9	1475.0	14.75	1.0	441.0	4700.0	1001	9200

Since the x^* ~ Poisson(λ^*), analytically, **mean of x^* = that of λ^***. From the OpenBugs output we can see that the mean of λ^* = 921.5 and the mean of x^* = 920.9. Those outcomes generated from MC are **very close to the analytic results**. Also, the **median and the credible set** are very similar. The **mean of λ = mean of (1000θ)**.