```
model{                          # *** PROGRAM STARTS
for(i in 1:Narmout){                    #   LOOP THROUGH STUDIES
   mu[i] ~ dnorm(0,.0001)           # vague priors for all trial baselines
   for (k in 1:narms[i]) {            #  LOOP THROUGH ARMS
     prec[i,k] <- pow(se[i,k],-2)      # set precisions
     y[i,k] ~ dnorm(mean[i,k],prec[i,k]) # Normal likelihood
           mean[i,k]<- theta[i,k]*sd.pooled[i] #Transform to standardised scale
     theta[i,k] <- mu[i] + delta[i,k]   #Model on SMD's
     delta[i,k]~dnorm(md[i,k],prec.out)            #Hierarchical model over outcomes
     md[i,k]<-d[t[i,k],type[i]] - d[t[i,1],type[i]]#NMA Model
#Deviance contribution
       dev[i,k] <- (y[i,k]-mean[i,k])*(y[i,k]-mean[i,k])*prec[i,k]
     }
# residual deviance contribution, trial i
   resdev[i] <- sum(dev[i,1:narms[i]])
 }

for (i in (Narmout+1):(Narmout+Nd)){

          prec[i,2]<-pow(se[i,2],-2)

          y[i,2]~dnorm(delta[i,2], prec[i,2])

          delta[i,2]~dnorm(md[i,2],prec.out)    #Hierarchical model over outcomes

  md[i,2]<-d[t[i,2],type[i]] - d[t[i,1],type[i]]

#Deviance contribution, trial i

  resdev[i] <- (y[i,2]-delta[i,2])*(y[i,2]-delta[i,2])*prec[i,2]

}

totresdev <- sum(resdev[])          #Total Residual Deviance

d[1,1]<-0
d[1,2]<-0

# vague priors for treatment effects
for (k in 2:nt){
          d[k,1] ~ dnorm(0,.0001)
          d[k,2] ~ dnorm(0,.0001)
          }

prec.out<-pow(sd.out,-2)
sd.out~dnorm(0,.01)I(0,)

for (k in 1:nt.MH){
          best.MH[k]<- equals(rank(d[,1],k),1) #Smaller values good
```

```
  for (r in 1:nt.MH){
            prank.MH[k,r]<-equals(rank(d[,1],k),r)  #count when k is ranked r
            }
}

for (k in 1:nt.BEH){
            d.rank[k]<-d[k,2]
            best.BEH[k]<- equals(rank(d.rank[],k),1) #Smaller values good
  for (r in 1:nt.BEH){
            prank.BEH[k,r]<-equals(rank(d.rank[],k),r)#count when k is ranked r
            }
}

for (c in 1:(nt.MH-1)){for (k in (c+1):nt.MH){
                        smd.MH[c,k]<- d[k,1] - d[c,1] # All pairwise SMDs
            }
}
            for (c in 1:(nt.BEH-1)){for (k in (c+1):nt.BEH){
                                    smd.BEH[c,k]<- d[k,2] - d[c,2]
            }
}
dum<-s[1]
}

#DATA
list(Narmout=15,Nd=3,nt.MH=8, nt.BEH=5, nt=8)
```

s[]	narms[]	t[,1]	t[,2]	t[,3]	type[]	y[,1]	se[,1]	y[,2]	se[,2]	y[,3]	se[,3]	sd.pooled[]
1	3	1	2	3	1	11.030	1.255	7.290	0.929	11.290	1.412	9.343166844
1	3	1	2	3	2	14.960	1.702	18.310	2.003	12.790	1.097	12.37689479
2	2	1	4	NA	1	37.8	1.648	39.9	1.966	NA	NA	12.95662604
2	2	1	4	NA	2	37.8	1.731	39.6	1.895	NA	NA	12.9942714
3	2	1	4	NA	1	57.2	1.263	56.9	1.250	NA	NA	10.05012438
3	2	1	4	NA	2	45.5	1.438	48	1.388	NA	NA	11.30176977
4	2	4	5	NA	1	51.590	2.277	48.070	1.881	NA	NA	8.859909706
4	2	4	5	NA	2	58.590	3.210	49.790	2.161	NA	NA	11.61019595
6	2	7	8	NA	1	-2.44	0.753	-1.03	0.502	NA	NA	8.810326421
6	2	7	8	NA	1	-7.16	1.690	-1.66	1.180	NA	NA	18.45026993
6	2	7	8	NA	1	-6.66	1.573	-1.53	1.081	NA	NA	17.13600378
6	2	7	8	NA	1	-3.31	0.435	-1.68	0.416	NA	NA	2.663710434
7	2	4	6	NA	1	6.710	0.843	4.420	0.477	NA	NA	3.702017468
8	2	6	7	NA	1	-2.000	0.260	-1.930	0.259	NA	NA	1.246386596
8	2	6	7	NA	2	1.160	0.183	1.600	0.151	NA	NA	0.799522585
9	2	4	5	NA	2	NA	NA	-0.63	0.204	NA	NA	NA
9	2	4	5	NA	2	NA	NA	-0.66	0.31377551	NA	NA	NA
9	2	4	5	NA	2	NA	NA	-0.57	0.301020408	NA	NA	NA

END

```
#INITS
#chain 1
list( mu=c(0, 0, 0, 0, 0,    0, 0,0,0,0,  0, 0, 0, 0, 0), sd.out=1,
d=structure(.Data=c(NA,NA,          -1,-1,     0,0,                  -1,-1,     0,0,
            -1,-1,     0,0,                  -1,-1),.Dim=c(8,2)))
```

```
#chain 2
list(mu=c(-1, 1, 2, -1, -2,    -3, -1,0,-2,0,          -1, -3, 1, -5, 0),  sd.out=0.5,
d=structure(.Data=c(NA,NA,              1,0,        0,0,        1,0,        -1,-2,
            0,-0.5,    0.7,1,                   -0.1,0),.Dim=c(8,2)))
```