```
> ##Testcase one
> y <- c(1,3)
> z <- c(6,8)
> x <- cbind(y,z)
> a <- makeCacheMatrix (x)
> a$set(x)
> a$get()
     y z
[1,] 1 6
[2,] 3 8
> a$setsolve(x)
> a$getsolve()
  [,1] [,2]
y -0.8  0.6
z  0.3 -0.1
> ## check if calculation is correct
> solve(x)
  [,1] [,2]
y -0.8  0.6
z  0.3 -0.1
>
> ##Testcase two
> ##check if the calculated inversion is returned
> cacheSolve(a)
getting cached data
  [,1] [,2]
y -0.8  0.6
z  0.3 -0.1
>
> ##Testcase three
> ##check if the inversion is calculated in cacheSolve
> ##when a new matrix if presented to it
> y2 <- c(6, 8)
> y3 <- c(11, 13)
> x <- cbind(y2, y3)
> a <- makeCacheMatrix (x)
> cacheSolve(a)
calculated new data
   [,1] [,2]
y2 -1.3  1.1
y3  0.8 -0.6
> solve(x)
   [,1] [,2]
y2 -1.3  1.1
y3  0.8 -0.6
>
> ##Testcase four
> ##check if the inversion is calculated in cacheSolve
> ##when a new matrix if presented to it
> y1 <- c(1, 3, 3, 5)
> y2 <- c(6, 9, 9, 9)
> y3 <- c(11, 21, 22, 14)
> y4 <- c(19, 26, 18, 19)
> x <- cbind(y1, y2, y3, y4)
> a <- makeCacheMatrix (x)
> cacheSolve(a)
calculated new data
           [,1]       [,2]        [,3]         [,4]
y1 -0.58513932  0.4860681 -0.30340557  0.207430341
y2  0.59545924 -0.7874097  0.37048504  0.131062951
y3 -0.14860681  0.1393189  0.03405573 -0.074303406
y4 -0.01857585  0.1424149 -0.12074303 -0.009287926
> solve(x)
           [,1]       [,2]        [,3]         [,4]
y1 -0.58513932  0.4860681 -0.30340557  0.207430341
y2  0.59545924 -0.7874097  0.37048504  0.131062951
y3 -0.14860681  0.1393189  0.03405573 -0.074303406
y4 -0.01857585  0.1424149 -0.12074303 -0.009287926
```