makeCacheMatrix <- **function**(x = matrix()) {

i <- NULL

set <- **function**(y) {

x <<- y

i <<- NULL

}

get <- **function**() x

setinverse <- **function**(inverse) i <<- inverse

getinverse <- **function**() i

list(set = set,

get = get,

setinverse = setinverse,

getinverse = getinverse)

}

cacheSolve <- **function**(x, **...**) {

i <- x$getinverse()

**if** (!is.null(i)) {

message("getting cached data")

**return**(i)

}

data <- x$get()

i <- solve(data, **...**)

x$setinverse(i)

i

}

B <- matrix(c(1,2,3,4),2,2)

*#solve(B) #We pretend that this cant't happen xD*

B1 <- makeCacheMatrix(B)

cacheSolve(B1) *#inverse returned after computation*

## [,1] [,2]

## [1,] -2 1.5

## [2,] 1 -0.5

cacheSolve(B1) *#inverse returned from cache*

## getting cached data

## [,1] [,2]

## [1,] -2 1.5

## [2,] 1 -0.5