Newest first Most popular

Sort replies by: Oldest first

coursera



R Programming

by Roger D. Peng, PhD, Jeff Leek, PhD, Brian Caffo, PhD

Signature Track



Q

Announcements

WEEK-BY-WEEK

Week 1: Getting Started and R Nuts and Bolts

Week 2: Programming with R

Week 3: Loop Functions and Debugging

Week 4: Simulation and Profiling

LECTURES

Lecture Videos

Lecture Slides (link to GitHub)

EXERCISES

Quizzes

Programming Assignments 1, 3, & swirl

Programming Assignment 2

ABOUT THE COURSE

Syllabus

About the Instructor

COMMUNITY

DSS Community Site 🗂

Discussion Forums

Course Wiki

Help Articles

Forums / Programming Assignment 2 (peer assessment): Lexical Scoping

Help

Testing cacheSolve:verify that cacheSolve works and will actually store the inverse

Subscribe for email updates.

No tags yet. + Add Tag

Bryan Signature Track · 7 days ago %

I am not sure how helpful this will be for people but for me, it helped bring home the usefulness of implementing cacheSolve. I made a set of test scripts containing the following:

```
## first script: test cacheSolve repeatedly
x <- matrix(rnorm(160000),400,400)
x_ <- makeCacheMatrix(x)
for (i in 1:1000) {
    x__ <- cacheSolve(x__)
}</pre>
```

```
## second script: test solve repeatedly
x <- matrix(rnorm(160000),400,400)
for (i in 1:1000) {
    x_ <- solve(x)
}</pre>
```

```
## third script: verify equality
x <- matrix(rnorm(160000),400,400)
x_ <- makeCacheMatrix(x)
cacheSolve(x_)
print(identical(x_$getInv(),solve(x)))</pre>
```

Using system.time, I have verified that the time elapsed on calling cacheSolve, for example, on a 400 by 400 cached matrix 1000 times in a loop is about .1 second (assumes that there is no output in cacheSolve, as there was in cachemean). The time elapsed on calling solve on a 400 by 400 non cached matrix 1000 times in a loop is more like 69 seconds. The third script prints out TRUE, as expected.

↑ 9 **↓** • flag

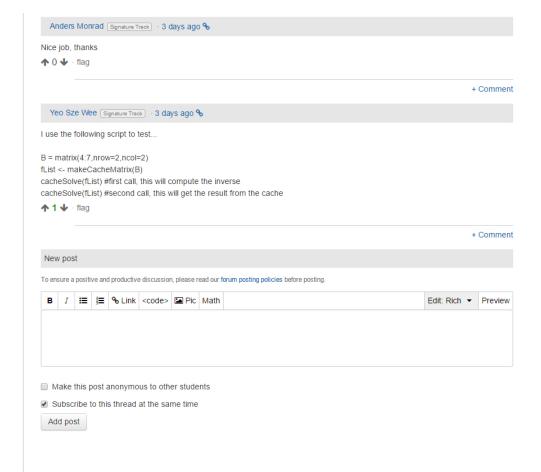
■ Dmitry Grekov - 6 days ago %

Good job, should be of great help.

I'd also suggest making one script which runs your scripts #1 and #2, measures time for both cases and provides the result in a readable form.

↑ 0 **↓** · flag

+ Comment



https://class.coursera.org/rprog-010/forum/thread?thread_id=288 Sun Jan 18 2015 13:21:40 GMT-0700 (Mountain Standard Time)