Name: ____

Directions:

Use OMSI for this quiz. Note the following:

- Your OMSI window must fill your laptop screen at all times. Use of any Internet communication other than intended OMSI functions is not allowed.
- For coding problems, you will be given a test function, which you will copy-and-paste into your Answers window, together with the function(s) you write for the problem. When you hit Run, OMSI will run what is in your window. (Do NOT insert a line **if __name__ ....**)
- You must hit Save for all problems.
- For noncoding problems, do not hit Run.
- If you wish, you may take a picture of your screen during the last 60 seconds of the quiz, or within 60 seconds of your leaving the room if you do so early.

1. (100) In this problem, you will write a function `countinst(startdir,flname)` that searches the directory tree rooted at `startdir`, and returns the count of all instances of files with the name `flname`. Both directories and ordinary files with that name are included in the count.

If you wish, you may call my version of the `walk()` function in our homework:

```
walk <- function(currdir,f,arg) {
   # "leave trail of bread crumbs"
   savetop <- getwd()
   setwd(currdir)
   fls <- list.files()
   arg <- f(currdir,fls,arg)
   # subdirectories of this directory
   dirs <- list.dirs(recursive=FALSE)
   for (d in dirs) arg <- walk(d,f,arg)
   setwd(savetop) # go back to calling directory
   arg
}
```

If you do use it, you must copy-and-paste it into your OMSI answer window. If you don't use it, you still may find its code helpful.

You must include this test function, and a call test() in your OMSI answer window, so that it will run:

```
test <- function() {
   unlink('mydir',recursive=TRUE)  # rmdir
   dir.create('mydir')
   file.create('mydir/x')
   dir.create('mydir/d1')
   dir.create('mydir/d2')
   file.create('mydir/d2/x')
   print(countinst('mydir','x'))
}
```

Solutions: 1.

```
# code to find the number of instances of files with the given name in
# the given directory tree

test <- function() {
   unlink(’mydir’,recursive=TRUE)
   dir.create(’mydir’)
   file.create(’mydir/x’)
   dir.create(’mydir/d1’)
   dir.create(’mydir/d2’)
   file.create(’mydir/d2/x’)
   print(countinst(’mydir’,’x’))
    # print(countinst.alt(’mydir’,’x’))
}

countinst <- function(startdir,flname) {
   walk(startdir,checkname,list(flname=flname,tot=0))
}

checkname <- function(drname,filelist,arg) {
   if (arg$flname %in% filelist) arg$tot <- arg$tot + 1
   arg
}

# descend the directory tree starting at currdir, calling the
# user-supplied f(currdir,fls,arg) at each step; walk provides f() with
# the files list fls in the given directory, and arg acts as a ”running
# total” specific to the given application

walk <- function(currdir,f,arg) {
   # ”leave trail of bread crumbs”
   savetop <- getwd()
   setwd(currdir)
   fls <- list.files()
   arg <- f(currdir,fls,arg)
   # subdirectories of this directory
   dirs <- list.dirs(recursive=FALSE)
   for (d in dirs) arg <- walk(d,f,arg)
   setwd(savetop) # go back to calling directory
   arg
}

countinst.alt <- function(startdir,flname,arg=0) {
   savetop <- getwd()
   setwd(startdir)
   fls <- list.files()
   if (flname %in% fls) arg <- arg + 1
   dirs <- list.dirs(recursive=FALSE)
   if (length(dirs) > 0)
   for (d in dirs) {
      arg <- countinst.alt(d,flname,arg)
   }
   setwd(savetop)
   arg
}
```

Solutions:

```
    file.create('mydir/x')
    dir.create('mydir/d1')
    dir.create('mydir/d2')
    file.create('mydir/d2/x')
    print(countinst('mydir','x'))
}
```

Norm

```
From nsmatloff@ucdavis.edu Sun Mar 12 12:28:09 2017
Date: Sun, 12 Mar 2017 12:28:09 -0700
From: Norm Matloff <nsmatloff@ucdavis.edu>
To: Austen Winters <abwinters@ucdavis.edu>
Subject: Re: ECS 145 Quiz 1 Grade
Message-ID: <20170312192809.GA13990@laura>
References: <CALBBWY-5DS3e26_tvb4Bg46=rmxz5Og3y9Vgue9Ud3AqmTDY=Q@mail.gmail.com>
MIME-Version: 1
```