Assignment 1

This is the second assignment of this class. The previous assignment was just to evaluate your overal knowledge of programming, and also to practice on LaTeX. The programming section of this assignment will be about the use of file I/O. We do not assume that you have a lot of experience with MATLAB i/o operations, though you are highly encouraged to read about them, if you want to have less painfull moments.

Contents

Problem 1

Your task for this problem is pretty straight forward. You will be given a file `data.txt` that contains some dataset you need to work on. Your task is to read this data into MATLAB (or any programming language of your choice, though the started code is given only for MATLAB/Octave). File I/O are very important in general, and they are also important in our case. Think of these data as some data from a GPS that you want to get to your PC. This assignment will **not** go to deep about parsing data. The data is pretty neat, parsing it should not be very challenging. You will also be given a ready made function to perform some computations. This function is quite simple, though it is important

- Unlike what you have been familiar to see, this code is a vectorized implementation. Vectorization is very important in scientific computations world. Make sure you understand it.
- While you do not need to worry about the function, you can use it without even looking at it, it gives you a very simple and intuitive way of how vectorization works in MATLAB/Octave.

Note

- This function is not fully compatible with MATLAB. If you need to run it using MATLAB, please replace the hashes at the beginning of each line with the precent sign.
- You might also need to replace any occurence of double quotes ", with a single quote sign '. After that it should be work just fine in your MATLAB.

Starter code

In file spherical2cartesian.m you will find a function that converts spherical coordinates to cartesian ones. As we said, you do not need to worry too much about it.

Dataset

The dataset for this week assignment can be found on data.txt file. It is a space delimiter file with three columns. The columns are [phi, lambda, height], respectively.

Your task

Read this data into a MATLAB array (or matrix), you can also read them into sperate vectors if you like. That's it actually you just need to make a function that read them into MATLAB's array.

In particular you are asked to do the following

- Read a date from a file, store it in some variable.
- Using the provided function spherical2cartesian, convert these data.
- Store the results of step two into another file.

Testing your code

Consider that I've called my function read_delim, and the file I would like to read is also called data.txt. Here how it should work

```
A = read_delim("path/to/my/file", ARGS)
A
1 4 6
5 7 8
```

In this case, the contents of myfile data.txt is just these numbers.

```
function [x, y, z] = spherical2cartesian(phi, lambda, height)
  # This function converts spherical coordinates to cartesian ones
  # -*- Usage -*-
  # spherical2cartesian(lat, long, height)
  # Where lat, long, and height can be scalars or vectors.
  #
  # Note that this is a vectorized version. It is very fast, but might be a little bit
  # harder to understand. Basically I've used something called array broadcasting.
  # Last important note about this function. While it is quite difficult compared to
  # the non-vecotrized one, it is actually a lot faster. Indexing, and the good understanding
  # of vectorization is a powerful tool. Try to always write your functions that way. You will
  # get a c-level code for free!

  if isscalar(height)
    x = prod([cos(phi); cos(lambda)], dim=1) * height;
    y = prod([cos(phi); sin(lambda)], dim=1) * height;
    z = sin(phi) * height;
  else
    x = prod([cos(phi); cos(lambda); height], dim=1);
    y = prod([cos(phi); sin(lambda); height], dim=1);
    z = prod([sin(phi); height], dim=1);
  endif
```

Shperical to casrtesian function.

You need to call this function on your parsed data, and store the results into another file

---

Published with GNU Octave 4.2.0