

hw2.sagews

April 4, 2014

Contents

1 Homework 2 - Math 480b - Spring 2014	1
1.1 Problem 1	1
1.2 Problem 2	1
1.3 Problem 3	1
1.4 Problem 4	2
1.5 Problem 5	2
1.6 Problem 6	2
1.7 Problem 7	2

1 Homework 2 - Math 480b - Spring 2014

Due Friday, April 11, 2014 by 6pm

Turn it in by creating a folder called homework2 in your project, with this worksheet in it. It will be automatically collected.

NOTE: You can put this worksheet in a SageMathCloud project by navigating to a directory (using the Files tab), then click +New and paste the following in the blank then click From Web":

<https://raw.githubusercontent.com/williamstein/sage2014/master/homework/homework2/hw2.sagews>

You can also grab this homework from the shared project: <https://cloud.sagemath.com/projects/edf7b34d-8ef9-49ad-b83f-8fa4cde53380/>

1.1 Problem 1

Write a function called `flatten`, which takes a list and flattens it that is, removes any internal lists, while keeping the elements in the same order. This is best explained by example:

```
sage: flatten([1,2,3]) [1, 2, 3] sage: flatten(["house", ["boat"], 3]) ["house", "boat", 3] sage: flatten([[1,2],
[[[3]], 4, [[5, 6]]]]) [1, 2, 3, 4, 5, 6]
```

This is a very standard problem, and solutions can be found online easily, so try hard to do it without just looking up the answer. Its for your benefit.

1.2 Problem 2

Write a function `triangle` which takes a positive integer n and a positive integer k as input and returns the sum of the k powers of the first n integers:

```
triangle(3,1) 1+2+3 6 triangle(3,2) 1+4+9 14 triangle(100, 1) 5050
```

You should be able to do this in one line with a list comprehension dont worry about making it fast, just do something naive and easy to implement.

1.3 Problem 3

Implement the double factorial that is, the function that inputs a positive integer n and returns the product of all the odd integers less than or equal to n .

1.4 Problem 4

Write a function called `copy_and_flatten_directory` that takes as input two directory names. It then copies all files in all subfolders of the first directory into the second subdirectory, ignoring the directory structure completely. (That is, it flattens it.) So, for instance, if the first folder had these files

```
mydir/foo/bar/baz.txt mydir/foo/bar/pie.pdf mydir/foo/bar/stuff.txt mydir/myfile.py mydir/other_stuff.ss
```

then, after this function is run, the second folder should contain

```
baz.txt myfile.py other_stuff.ss pie.pdf stuff.txt
```

Python has functions like `os.path.listdir`, `shutil.copyfile`, etc., which will be useful. Do import `shutil`, then `shutil.[tab key]` to get going.

1.5 Problem 5

Write a function `remove_duplicates` that takes as input a list (or tuple), and returns a new list (or tuple, if the input was a tuple) which contains exactly one copy of each entry in the input list (or tuple). Heres an example:

```
remove_duplicates([3,3,3,3]) [3] remove_duplicates((3,3,3,3)) (3,) remove_duplicates([3, [3]]) [3, [3]] re-  
move_duplicates([3, [3], [3]]) [3, [3]]
```

1.6 Problem 6

I heard somebody claim that for every positive integer n we have

$$(1 + 2 + \dots + n)^2 = 1^3 + 2^3 + \dots + n^3.$$

Write a program which takes as input a positive integer n and verifies (or disproves) this claim for every positive integer up to and including n . (That is, compute both sides and check that they match.) You should print some output as you go but printing something for every single integer would probably be overkill. Try to print a manageable amount of data.

1.7 Problem 7

Use Sage to solve at least 5 Project Euler problems from here: <http://projecteuler.net/problems>

Carefully document which of the 5 problems you solved below, along with the solution, so that the graders will be able to verify that you solved them.