

## Homework #5

**Due:** March 7, 2012

**Points:** 100

As for all earlier assignments, please put refinements for all your programs into one file called “refinements.txt” or “refinements.pdf”, and error journal entries for all your programs into one file called “errors.txt” or “errors.pdf”.

Problem 1 asks you to write a *function*, not a program. You do not need (and in fact should not submit) anything to provide a user interface. When we grade, we will use our own programs to test your function.

Problem 2 requires a program, though, and the extra credit requires both a function and a program.

- (50 points) Write a function “flatten” that returns a simple list containing all the values in a nested list. For example, the following function calls:

```
flatten([2,9,[2,1,13,2],8,[2,6]])
flatten([[9,[7,1,13,2],8],[7,6]])
flatten([[9,[7,1,13,2],8],[2,6]])
flatten(['this',['a',['thing'],'a'],'is'],['a','easy'])
```

should produce the following lists

```
[2,9,2,1,13,2,8,2,6]
[9,7,1,13,2,8,7,6]
[9,7,1,13,2,8,2,6]
['this','a','thing','a','is','a','easy']
```

(Text, chapter 14, problem 7, modified)

**Submit:** Please turn in your program in the file “flatten.py”.

*Hint:* Doing this recursively is considerably easier than any other way.

- (50 points) The National Weather Service computes the windchill index using the following formula:

$$35.74 + 0.6215T - 35.75V^{0.16} + 0.4275TV^{0.16}$$

where  $T$  is the temperature in degrees Fahrenheit and  $V$  is the wind speed in miles per hour.

Write a program that prints a nicely formatted table of windchill values. Rows represent wind speed for 0 to 50 in 5 mph increments, and columns represent temperatures from  $-20$  to  $+60$  in 10-degree increments. Your program must use the above formula to compute the entries in the table.

**Submit:** Please turn in your program in the file “chill.py”.

### Extra Credit

Submit this program in the Extra Credit 6 area of SmartSite—not in the Homework 6 area.

- (20 points) Write a function to determine whether a year, given as the argument, is a leap year. A year is a leap year if it is divisible by 4, unless it is divisible by 100 and not 400. So 2000 was a leap year, but 2100 and 2200 will not be. Then write a program that asks the user to enter a year and uses the function you wrote to determine whether the year is a leap year. The program then prints the result.

**Submit:** Please turn in your program in the file “leap.py”.