# Advice for the Tests

- These two things can make the difference of whether you ***pass or fail*** this class before taking the exams
  - Go through each class days notes and example programs on the website
  - Practice coding over and over!!! This is the only way to really learn.
- Review by reading the book!!

15

Try things. Redo an in–class exercise without looking at your notes or the posted solution. Type in the code and see if it works. Compare your solution to the posted solution or the solution in your notes. Do you understand the concepts and techniques and patterns better after doing this?

Read and study code examples from class and from lab and from the textbook. What does each statement do? Why was it needed? Why did it need to be exactly in the location it was placed in the entire program?

Try things in the IDLE shell to verify that the Python you are reading about does what the notes or examples or book says it does. Test out ideas in the shell before using that specific idea in a larger program.

Resist the urge to code. Work on understanding WHAT it is first BEFORE you try to develop a PLAN for HOW to solve it. And then translate each step of your step by step PLAN into Python code. Your PLAN is an outline, is an algorithm.

Try to solve a problem a different way. For example, if you have a program that works with a for loop, modify that program to use a while loop instead. Do you understand that program a little bit better after typing it in and trying it out and then modifying it to use a WHILE loop instead of the original FOR loop and getting that to work?

Can you modify the http://www.cs.uni.edu/~jacobson/1510/sessions/11/binaryPractice.py Decimal to Binary (base ten to base two) program so that it converts a number to base 4? Try it. Practice, practice, practice.

Can you modify one of the example hexadecimal Python programs at the session 12 link to convert from base ten decimal to base 18 (instead of base 16 hexadecimal)? Base 18 digits are: 0123456789ABCDEFGH with G being the digit to represent 16 and H being the digit to represent 17. See the session #12 URL (Thursday of week #6), which is: http://www.cs.uni.edu/~jacobson/1510/sessions/11/base16.html and modify one or more of those versions to produce base 18 results. It will help you understand the base 16 hexadecimal examples when you modify them to produce a different result. Practice, practice, practice!

**The output for base 10 numbers (21, 33, and 234) in base 2, base 16, base 4 and base 18:**

**obase=2 Base Two Binary**

```
21                  ← 21 is the decimal input
10101               ← 10101 is the binary output
33
100001
234
11101010
```

**obase=16 Base Sixteen Hexadecimal**

```
21
15
33
21
234                 ← 234 is the decimal input
EA                  ← EA is the base 16 output
```

**obase=4 Base Four – Go Chicago CUBS!**

```
21
111
33                  ← 33 is the base 10 input
201                 ← 201 is the base 4 output    (See note: ****)
234
3222
```

**obase=18 Base Eighteen**

```
21
13
33
1F
234
D0
234 + 17            ← 251 is the decimal input
DH                  ← DH is the base 18 output
```

**** Hopefully, the Chicago CUBS will get lots of base 4 output in the playoffs. Back to the Future and Marty McFly?