# Numerical Analysis and Programming

## Lab Worksheet #2

1. When defining a function, it is possible to specify a *default value* for one or more arguments. This creates a function that can be called with fewer arguments than it is defined to allow. For example,

   ```
   def ask_ok(prompt, retries=4, complaint='Yes or no, please!'):
       while True:
           ok = raw_input(prompt)
           if ok in ('y', 'ye', 'yes'):
               return True
           if ok in ('n', 'no', 'nop', 'nope'):
               return False
           retries = retries - 1
           if retries < 0:
               raise IOError('refuse user')
           print complaint
   ```

   Try the following possibilities and find the outputs.

   - `ask_ok('Do you really want to quit?')`
   - `ask_ok('OK to overwrite the file?', 2)`
   - `ask_ok('OK to overwrite the file?', 2, 'Come on, only yes or no!')`

2. Using the following examples to understand the scope ideas.

   ```
   x=99
   def func(y):
       z=x+y
       return z
   ```

   Run the command

   ```
   >>> print x, func(x)
   ```

   Explain how Python gets the results, and which variables are local to `func`. A slight modification of the code gives,

   ```
   x=99
   def func(y):
       x=1
       z=x+y
       return z
   ```

   Run the `print` command again. Explain how Python gets the results, and which variables are local to `func`.