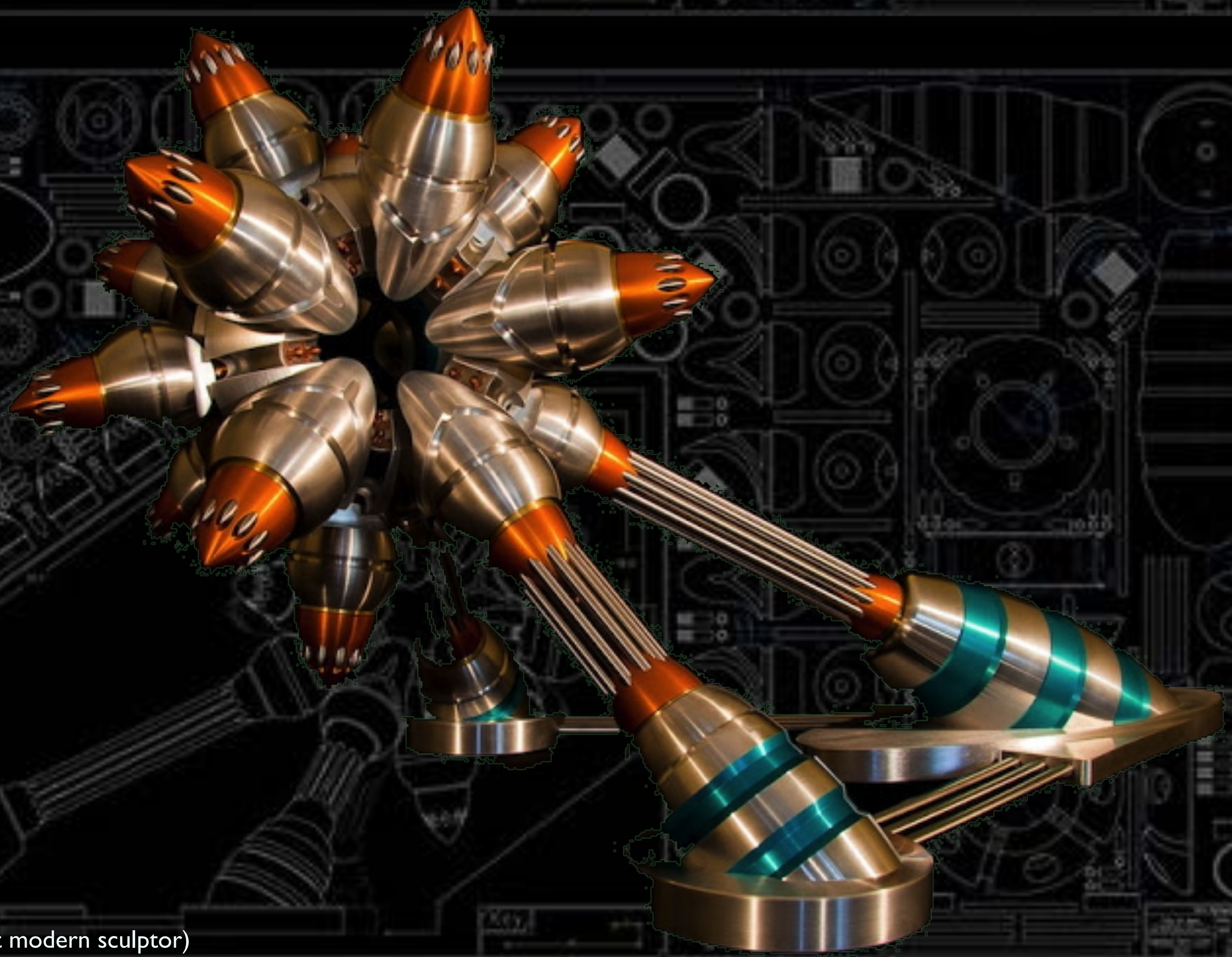
Modules and moving around the OS



background from http://www.chrisbathgate.com (Chris Bathgate: great modern sculptor)

Why the OS first?

- Python has the ability to work on multiple operating systems
- Handling files, moving around your directory structure, measuring time, etc. are necessary for any scripting project
- Python provides the means to explore the operating system identically on Windows, MacOSX, and Linux machines

Accessing Python

- From the command line:
  - python
  - iPython (much preferred: very powerful)
  - [pyshell (similar to Matlab shell)]
    - less powerful than iPython

iPython

- Command line completion
- Editing from within iPython
- Extended help
  - help (python's normal help)
  - quickref (help on ipython capabilities)
  - ? (used for help on specific objects)
    - Cannot be used *before* or *after* another command
    - a = [];
      ?a
- history

- If you use Windows, use QtConsole (iPython)

iPython: quickref

useful commands

- These commands only exist in iPython
  - edit (edit a file; uses EDITOR environment variable)
- hist
- cp file dir/newfile
- mkdir dirname
- tab (autocompletion)
- time, timeit (time commands, functions)
- who (print interactive variables)
- .... many others ....

Text Editors

- You will be writing programs, stored in files
- Useful text editors
  - gedit
  - vim: powerful
  - emacs: powerful (my favorite)
  - nano
- You can use your favorite editor
- On Linux-like systems, store the *path* of the editor in the EDITOR environment variable in .bash_profile (if using bash), or .cshrc (if using csh)

Useful Modules

- Construct a python file (a.py)
- Define a simple function inside a.py (print the argument)
- Construct b.py, and *import a*
- Function *a.fun* is now available within b.py
- dir(a) # contents of a

```
In [2]: dir(a)
Out[2]:
['__builtins__',
 '__doc__',
 '__file__',
 '__name__',
 '__package__',
 'myprint',
 'myreverse']
```

a.py

```
#!/usr/bin/env python
# simple module
# contains two functions
# - prints out argument
# - reverse text

def myprint(a):
    print a

def myreverse(a):
    print a[::-1]

if __name__ == "__main__":
    myprint("testing, testing")
    myreverse("testing, testing")
```

b.py

b.py

```
#!/usr/bin/env python
import a

if __name__ == “__main__”:
    a.myprint(“Hello world”)
    a.myreverse(“Hello World”)
```

Doc Strings

iPython

```
import a
help a
```

```
Help on module a:

NAME
    a

FILE
    /Users/beerli/ISC-4304/lectures/L2/a.py

DESCRIPTION
    # simple module
    # prints out argument and reverse text

FUNCTIONS
    myprint(a)
        Prints out its argument

    myreverse(a)
        prints out reverse text
```

a.py

```
#!/usr/bin/env python
# simple module
# contains two functions
# - prints out argument
# - reverse text

def myprint(a):
    """
    Prints out its argument
    """
    print a

def myreverse(a):
    """
    prints out reverse argument
    """
    print a[::-1]

if __name__ == "__main__":
    myprint("testing, testing")
    myreverse("testing, testing")
```

Use triple quotes String as first line of module and first line of any function

dir() gives list of objects

```
In [9]: dir a.myprint.func_code
------> dir(a.fun.func_code)
Out[9]:
['__class__',
'__cmp__',
'__delattr__',
'__doc__',
'__eq__',
'__format__',
'__ge__',
'__getattribute__',
'__gt__',
'__hash__',
'__init__',
'__le__',
'__lt__',
'__ne__',
'__new__',
 '__reduce__',
 '__reduce_ex__',
'__repr__',
'__setattr__',
'__sizeof__',
'__str__',
'__subclasshook__',
'co_argcount',
'co_cellvars',
'co_code',
'co_consts',
'co_filename',
'co_firstlineno',
'co_flags',
'co_freevars',
'co_lnotab',
'co_name',
'co_names',
'co_nlocals',
'co_stacksize',
'co_varnames']
```

OS module
# of interest in blue

Stored in list: [a,b,c]

```
>>> import os
>>> dir(os)
['EX_CANTCREAT', 'EX_CONFIG', 'EX_DATAERR', 'EX_IOERR', 'EX_NOHOST', 'EX_NOINPUT', 'EX_NOPERM',
'EX_NOUSER', 'EX_OK', 'EX_OSERR', 'EX_OSFILE', 'EX_PROTOCOL', 'EX_SOFTWARE', 'EX_TEMPFAIL',
'EX_UNAVAILABLE', 'EX_USAGE', 'F_OK', 'NGROUPS_MAX', 'O_APPEND', 'O_ASYNC', 'O_CREAT',
'O_DIRECTORY', 'O_EXCL', 'O_EXLOCK', 'O_NDELAY', 'O_NOCTTY', 'O_NOFOLLOW', 'O_NONBLOCK',
'O_RDONLY', 'O_RDWR', 'O_SHLOCK', 'O_SYNC', 'O_TRUNC', 'O_WRONLY', 'P_NOWAIT', 'P_NOWAITO',
'P_WAIT', 'R_OK', 'SEEK_CUR', 'SEEK_END', 'SEEK_SET', 'TMP_MAX', 'UserDict', 'WCONTINUED', 'WCOREDUMP',
'WEXITSTATUS', 'WIFCONTINUED', 'WIFEXITED', 'WIFSIGNALED', 'WIFSTOPPED', 'WNOHANG', 'WSTOPSIG',
'WTERMSIG', 'WUNTRACED', 'W_OK', 'X_OK', '_Environ', '__all__', '__builtins__', '__doc__', '__file__', '__name__',
'__package__', '_copy_reg', '_execvpe', '_exists', '_exit', '_get_exports_list', '_make_stat_result', '_make_statvfs_result',
'_pickle_stat_result', '_pickle_statvfs_result', '_spawnvef', 'abort', 'access', 'altsep', 'chdir', 'chmod', 'chown', 'chroot',
'close', 'closerange', 'confstr', 'confstr_names', 'ctermid', 'curdir', 'defpath', 'devnull', 'dup', 'dup2', 'environ', 'errno',
'error', 'execl', 'execle', 'execlp', 'execlpe', 'execv', 'execve', 'execvp', 'execvpe', 'extsep', 'fchdir', 'fchmod', 'fchown', 'fdopen',
'fork', 'forkpty', 'fpathconf', 'fstat', 'fstatvfs', 'fsync', 'ftruncate', 'getcwd', 'getcwdu', 'getegid', 'getenv', 'geteuid', 'getgid',
'getgroups', 'getloadavg', 'getlogin', 'getpgid', 'getpgrp', 'getpid', 'getppid', 'getsid', 'getuid', 'isatty', 'kill', 'killpg', 'lchmod',
'lchown', 'linesep', 'link', 'listdir', 'lseek', 'lstat', 'major', 'makedev', 'makedirs', 'minor', 'mkdir', 'mkfifo', 'mknod',
'name', 'nice', 'open', 'openpty', 'pardir', 'path', 'pathconf', 'pathconf_names', 'pathsep', 'pipe', 'popen', 'popen2',
'popen3', 'popen4', 'putenv', 'read', 'readlink', 'remove', 'removedirs', 'rename', 'renames', 'rmdir', 'sep', 'setegid',
'seteuid', 'setgid', 'setgroups', 'setpgid', 'setpgrp', 'setregid', 'setreuid', 'setsid', 'setuid', 'spawnl', 'spawnle', 'spawnlp',
'spawnlpe', 'spawnv', 'spawnve', 'spawnvp', 'spawnvpe', 'stat', 'stat_float_times', 'stat_result', 'statvfs', 'statvfs_result',
'strerror', 'symlink', 'sys', 'sysconf', 'sysconf_names', 'system', 'tcgetpgrp', 'tcsetpgrp', 'tempnam', 'times', 'tmpfile',
'tmpnam', 'ttyname', 'umask', 'uname', 'unlink', 'unsetenv', 'urandom', 'utime', 'wait', 'wait3', 'wait4', 'waitpid', 'walk',
'write']
```

```
In [1]: import os

In [3]: os.makedir('f')
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
/Users/erlebach/Documents/sc/courses/programming_scientific_applications_s2011/code/
<ipython console> in <module>()

AttributeError: 'module' object has no attribute 'makedir‘

In [4]: os.mkdir('f')
```

OS-independent

```
In [5]: os.makedirs('a/b/c/d/e/f')
```

not OS-independent
Windows requires ‘\’

```
In [6]: ls
a/                    get_ref_count.py      mfc_font.c
a.py                  gui_hello.py          python/
f/                    hello_world/          references/

In [9]: os.listdir('.')
Out[9]:
['.gui_hello.py.swp', 'a', 'a.py', 'f', 'get_ref_count.py', 'gui_hello.py', 'hello_world', 'mfc_font.c', 'python',
'references']
```

Module sys

```
>>> dir(sys)
['__displayhook__', '__doc__', '__excepthook__', '__name__',
'__package__', '__stderr__', '__stdin__', '__stdout__', '_clear_type_cache',
'_current_frames', '_getframe', 'api_version', 'argv',
'builtin_module_names', 'byteorder', 'call_tracing', 'callstats', 'copyright',
'displayhook', 'dont_write_bytecode', 'exc_clear', 'exc_info', 'exc_type',
'excepthook', 'exec_prefix', 'executable', 'exit', 'flags', 'float_info',
'getcheckinterval', 'getdefaultencoding', 'getdlopenflags',
'getfilesystemencoding', 'getprofile', 'getrecursionlimit', 'getrefcount',
'getsizeof', 'gettrace', 'hexversion', 'last_traceback', 'last_type', 'last_value',
'maxint', 'maxsize', 'maxunicode', 'meta_path', 'modules', 'path',
'path_hooks', 'path_importer_cache', 'platform', 'prefix', 'ps1', 'ps2',
'py3kwarning', 'setcheckinterval', 'setdlopenflags', 'setprofile',
'setrecursionlimit', 'settrace', 'stderr', 'stdin', 'stdout', 'subversion',
'version', 'version_info', 'warnoptions']
```

using iPython

```
In [21]: import sys

In [22]: print sys.platform
linux2

In [23]: print sys.stdout
<open file '<stdout>', mode 'w' at 0x324078>

In [24]: sys.argv
Out[24]: ['/usr/local/bin/ipython']
```

List of directories to search for modules

```
In [1]: import sys
In [2]: sys.path          # results in a list
Out[2]:
['',
'/Library/Frameworks/Python.framework/Versions/6.3/bin',
'/Applications/Panda3D/1.6.2/lib', '/Users/erlebach/Downloads', '/Library/Frameworks/Python.framewo
Versions/6.3/lib/python26.zip', '/Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6', '/
Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/plat-darwin', '/Library/Frameworks/
Python.framework/Versions/6.3/lib/python2.6/plat-mac', '/Library/Frameworks/Python.framework/
Versions/6.3/lib/python2.6/plat-mac/lib-scriptpackages', '/Library/Frameworks/Python.framework/Versio
6.3/lib/python2.6/lib-tk', '/Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/lib-old', '/
Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/lib-dynload', '/Library/Frameworks/
Python.framework/Versions/6.3/lib/python2.6/site-packages', '/Library/Frameworks/Python.framework/
Versions/6.3/lib/python2.6/site-packages/PIL', '/Library/Frameworks/Python.framework/Versions/6.3/lib
python2.6/site-packages/IPython/Extensions',
u'/Users/erlebach/.ipython']
```

Exercise

- Consider the file igrid.py
  - locate igrid.py   # on linux
    /Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/site-packages/IPython/Extensions/igrid.py
    /Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/site-packages/IPython/Extensions/igrid.pyc
  - the file *igrid.py* is only contained in one location, related to iPython, therefore, it cannot be imported by python

Python import

```
                                 # cannot be imported
>>> import igrid
Traceback (most recent call last):
File "<stdin>", line 1, in <module>ImportError: No module named igrid
>>> import sys
>>> sys.path                # note: no reference to iPython in the list
['', '/Applications/Panda3D/1.6.2/lib', '/Users/erlebach/Downloads', '/Library/Frameworks/
Python.framework/Versions/6.3/lib/python26.zip', '/Library/Frameworks/
Python.framework/Versions/6.3/lib/python2.6', '/Library/Frameworks/Python.framework/
Versions/6.3/lib/python2.6/plat-darwin', '/Library/Frameworks/Python.framework/Versions/
6.3/lib/python2.6/plat-mac', '/Library/Frameworks/Python.framework/Versions/6.3/lib/
python2.6/plat-mac/lib-scriptpackages', '/Library/Frameworks/Python.framework/Versions/
6.3/lib/python2.6/lib-tk', '/Library/Frameworks/Python.framework/Versions/6.3/lib/
python2.6/lib-old', '/Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/lib-
dynload', '/Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/site-
packages', '/Library/Frameworks/Python.framework/Versions/6.3/lib/python2.6/site-
packages/PIL']
```

Update the path

```
>>> sys.path.append('/Library/Frameworks/Python.framework/   \
    Versions/6.3/lib/python2.6/site-packages/IPython/Extensions')
>>> import igrid      # imported properly
>>>
```

Line continuation: single backslash as *last character* of a line

Environment variables

stored in dictionary

```
>>> import os
>>> print os.environ
{'_': '/Library/Frameworks/Python.framework/Versions/Current/bin/python', 'CUDA_PROFILE': '1', 'ed':
'erlebach@gorerle.com', 'TERM_PROGRAM_VERSION': '273', 'LOGNAME': 'erlebach', 'USER': 'erlebach', 'ENJA_DIR': '/Users/
erlebach/Documents/src/blender-particles/enjacl', 'HOME': '/Users/erlebach', 'bones': 'gerlebacher@bones.sc.fsu.edu', 'PATH': '/
Library/Frameworks/Python.framework/Versions/Current/bin:/Applications/Panda3D/1.6.2/bin:/opt/openmpi/bin/:/Library/
Frameworks/Python.framework/Versions/Current/bin:/Users/erlebach/bin/:/usr/local/cuda/bin/:/opt/local/bin/:/usr/bin:/bin:/usr/
sbin:/sbin:/usr/local/bin:/usr/texbin:/usr/X11/bin:/usr/local/bin:/opt/local/bin:/opt/local/sbin', 'DISPLAY': '/tmp/launch-7LOtRB/
org.x:0', 'MAKEFLAGS': '-w -I/Users/erlebach/Documents/src/blender2.5/bf-blender/source', 'blender': '/Applications/
blender-2.49b-OSX-10.5-py2.5-intel/blender.app/Contents/MacOS/', 'LANG': 'en_US.UTF-8', 'TERM': 'xterm-color', 'SHELL': '/
bin/bash', 'GRAPHIC_LIBS_HOME': '/Users/erlebach/Documents/src/graphics_libs', 'FLIBS': 'libgfortran.so', 'SHLVL': '1',
'GDFONTPATH': '/Library/Fonts', 'sciprog': 'erlebach_sciprog@belmopan.dreamhost.com', 'acs1':
'erlebach_acs1@gorerle.com', 'MAYA_SCRIPT_PATH': '/Applications/Panda3D/1.6.2/plugins:', 'kirk':
'gerlebacher@kirk.sc.fsu.edu', 'EDITOR': '/usr/bin/vim', 'pd': 'gerlebacher@pamd.sc.fsu.edu', 'SSH_AGENT_PID': '188',
'LOCAL_CUDA': '/Developer/GPU_Computing/C', 'MKL_NUM_THREADS': '1', 'roland1':
'rmartin1@cluster.univ-pau.fr', 'CUDA_HOME': '/usr/local/cuda', 'PYTHONPATH': '/Applications/Panda3D/1.6.2/lib:',
'NVSDKCUDA_ROOT': '/Developer/CUDA', 'SSH_AUTH_SOCK': '/tmp/ssh-cxURkmJ1r9/agent.186', 'TERM_PROGRAM':
'Apple_Terminal', 'Apple_PubSub_Socket_Render': '/tmp/launch-WX7get/Render', 'F77': '/usr/local/bin/gfortran', 'TMPDIR': '/
var/folders/zT/zT3+c30eFeqXxsbODmGWOE+++TI/-Tmp-/', 'CUDA_LOCAL': '/Developer/GPU_Computing/C',
'DYLD_LIBRARY_PATH': '/Applications/Panda3D/1.6.2/lib:/Users/erlebach/Documents/src/armadillo-0.9.52:/opt/openmpi/lib/:/
usr/local/cuda/lib:/System/Library/Frameworks/OpenCL.framework/Versions/A/Libraries/', 'lincoln': 'erlebach@Login-
lincoln.ncsa.teragrid.org', 'AXIS2C_HOME': '/Users/erlebach/Documents/src/axis2c-src-1.6.0', 'MAYA_PLUG_IN_PATH': '/
Applications/Panda3D/1.6.2/plugins:', 'COMPILE_WITH_COCOA': '0', 'OLDPWD': '/Users/erlebach/python2011', 'acm':
'gerlebacher@acm.sc.fsu.edu', '__CF_USER_TEXT_ENCODING': '0x1F5:0:0', 'PWD': '/Users/erlebach/python2011/code',
'mark2': 'erlebacher@mark2.msi.umn.edu', 'NANBLENDERHOME': '/Users/erlebach/Documents/src/blender2.5/bf-blender',
'COMMAND_MODE': 'unix2003'}
```

Key: Value

math/cmath modules

- math: real numbers
- cmath: complex numbers

```
>>> import math
>>> dir(math)
['__doc__', '__file__', '__name__', '__package__', 'acos', 'acosh', 'asin', 'asinh', 'atan',
'atan2', 'atanh', 'ceil', 'copysign', 'cos', 'cosh', 'degrees', 'e', 'exp', 'fabs', 'factorial', 'floor',
'fmod', 'frexp', 'fsum', 'hypot', 'isinf', 'isnan', 'ldexp', 'log', 'log10', 'log1p', 'modf', 'pi', 'pow',
'radians', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'trunc']

>>> import cmath
>>> dir(cmath)
['__doc__', '__file__', '__name__', '__package__', 'acos', 'acosh', 'asin', 'asinh', 'atan',
'atanh', 'cos', 'cosh', 'e', 'exp', 'isinf', 'isnan', 'log', 'log10', 'phase', 'pi', 'polar', 'rect', 'sin', 'sinh',
'sqrt', 'tan', 'tanh']
```

import math

Complex arithmetic not allowed

```
In [11]: import math

In [12]: math.cos(2.)
Out[12]: -0.41614683654714241

In [13]: math.cos(2.+3j)
---------------------------------------------------------------------------TypeError
Traceback (most recent call last)
/Users/erlebach/Documents/sc/courses/
programming_scientific_applications_s2011/code/<ipython console> in
<module>()

TypeError: can't convert complex to float

In [14]:
```

Uncaught exception
==> code crash

import cmath

Complex math module

```
In [14]: import cmath

In [15]: cmath.cos(2.)
Out[15]: (-0.41614683654714241-0j)

In [16]: cmath.cos(2.+3j)
Out[16]: (-4.189625690968807-9.109227893755337j)
```

Name conflicts

```
>>> import math
>>> import cmath
>>> math.cos(3)
-0.98999249660044542
>>> cmath.cos(3)
(-0.98999249660044542-0j)
>>>
```

Identical function names in each module: **cos**

However, module names are different:

math, cmath

Modules

Simplified Usage

- Shortcut names:
  - import math as m
  - import cmath as cm   # cm.sqrt(3.45j)
- Import selected commands
  - from math import sqrt
    - only sqrt is available
- Import everything from module
  - from math import *   # sqrt(3.45)

However,

there is danger ...

Don’t Do This

```
>>> from cmath import *
>>> cos(3j)(10.067661995777765-0j)
>>>
>>> from math import *
>>> cos(3j)
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
TypeError: can't convert complex to float
>>>
```

Overwrite complex functions

Running a script.py file

- From the command line
  - python script.py
  - python ./a/b/script.py
  - python /full_path/script.py
  - ./script.py
  - ./a/b/script.py

Requires that the location of python binary be in the first line

- From within the python shell
  - execfile(‘script.py’)
  - execfile(‘./a/b/script.py’)
  - execfile(‘full_path/script.py’)

Variables

- Way to label and access information
- Naming variables
  - A variable name can contain only numbers, letters and underscores.
  - A variable name cannot start with a number.

- Guidelines
  - Choose descriptive names (ex. ***score*** instead of ***s***)
  - Be consistent with multiword names (***high_score*** or ***highscore***)
  - Follow the traditions of the language
    - variables start with a lowercase letter
    - avoid using an underscore as the first character
  - Keep the length in check
    - try to keep names under 15 characters

Python's Keywords

- and elif if print as else import raise
- assert except in return break exec is try
- class finally lambda while continue for not with def from or yield del global pass

- Can't use these keywords as variable names.

```
In [28]: and=1
  File "<ipython-input-28-2b18fd315f8e>", line 1
    and=1
      ^
SyntaxError: invalid syntax
```

Numeric Variable Types

- In languages like C/C++/Java, one has 4 byte ints, 8 byte longs, 4 byte floats, 8 byte doubles (on a 32-bit machine)
- In Python
  - int
    - 4 bytes on 32-bit machines,
    - 8-bytes on 64-bit machine
  - long
    - infinite precision
    - sys.maxint (on my mac) returns 9223372036854775807 = 2**63 - 1
  - float
    - equivalent to double in C++, so 8 bytes on 32-bit machine
  - complex (**j** represents cmath.sqrt(-1))
    - a = 1+3j

Digression

- One byte = 8 bits
- one bit: 0 or 1
- two bits: 00, 01, 10, 11 = 0, 1, 2, 3
    - represents the integers 0 to 3=2**2-1
      (** is exponentiation), or integers -2,0,1,2 = [-2**1,2**1-1]
- 3 bits: 000, 001, 010, 011, 100, 101, 110, 111
    - represents the positive integers 0 to 7=2**3-1 or
      pos/neg integers [-2**2, 2**2-1] = [-4,3]
- 8 bits represents integers 0 to 2**8-1 = 255 or [-128,127]
- 32 bits represents ints 0 to 2**32-1 or [-2**31, 2**31-1] = **2147483647L**
- **64** bits represents ints 0 to 2**63-1 or [-2**63, 2**63-1] = [**0,9223372036854775807L]**
- sys.maxint = **9223372036854775807**

What is going on?

- 2147483647   versus   2147483647**L**
- An **L** denotes *long,* which means infinite precision
- Given that 2**31-1 fits into 4 bytes, why is there a **L**?
  - ON 32bit machines:
    it is because 2**31 does *not* fit into 4 bytes, and it must be calculated first.
  - second step: subtract 1 from a long; therefore, Python does automatic format conversion from the longer format to the shorter one (not to lose precision).
  - The final result is *not* converted back to int size.

Strings

```
>>> a = "going to class“
>>> b = 'going to class‘
>>> c = """going to class""“
>>> d = r'going to class’
>>> f = u’done with class’
```

‘\n’ is a carriage return

raw format

unicode

using **Triple quotes**

```
In [39]: a="""going
   ....:     to school
   ....:   late
   ....: """

In [40]: a
Out[40]: 'going \n    to school\n  late\n'

In [41]: print a
going
    to school
  late
```

Advantage of raw format

- I want to encode the string: “path\to\file”
- “\” is a special character and one must do:“\\”

```
In [72]: a='\a\b\c'

In [73]: a
Out[73]: '\x07\x08\\c'

In [74]: print a
\c

In [75]: a="""\a\b\c"""

In [76]: print a
\c
```

```
In [77]: a=r'\a\b\c'

In [78]: print a
\a\b\c

In [79]: a='\\a\\b\\c'

In [80]: print a
\a\b\c
```

Booleans

- A boolean is either True or False
- In some languages, -1 if true, all else is false, in others, 0 is false, all else is true
- In Python : the number 0 is false, all other numbers are true. Do not assume this!!!
  - None, (), [], 0, “” returns false
    - bool(None) ==> False
    - bool([]) ==> False
    - bool(34) ==> True
    - bool(None or 34) ==> True
    - bool(34 and (not 0 or “”)) ==> True

What is None?

```
In [1]: bool(trip)
------------------------------------------------------------------
NameError                                 Traceback (most recent
call last)
<ipython-input-1-2b29e319ed42> in <module>()
----> 1 bool(trip)

NameError: name 'trip' is not defined

In [2]: trip=None
In [3]: bool(trip)
Out[3]: False

In [4]: trip=""
In [5]: bool(trip)
Out[5]: False

In [6]: trip=3
In [7]: bool(trip)
Out[7]: True
```

- **None** is the absence of definition
- **“”** is the empty string

Experiments

- Experiment with
  - changing directories
  - moving/copying files
  - creating directories
  - creating temporary files
  - real and complex arithmetic
  - explore sys