

## Chapter 1: An Introduction to NumPy



Image courtesy Fernando Perez & Ondrej Certik

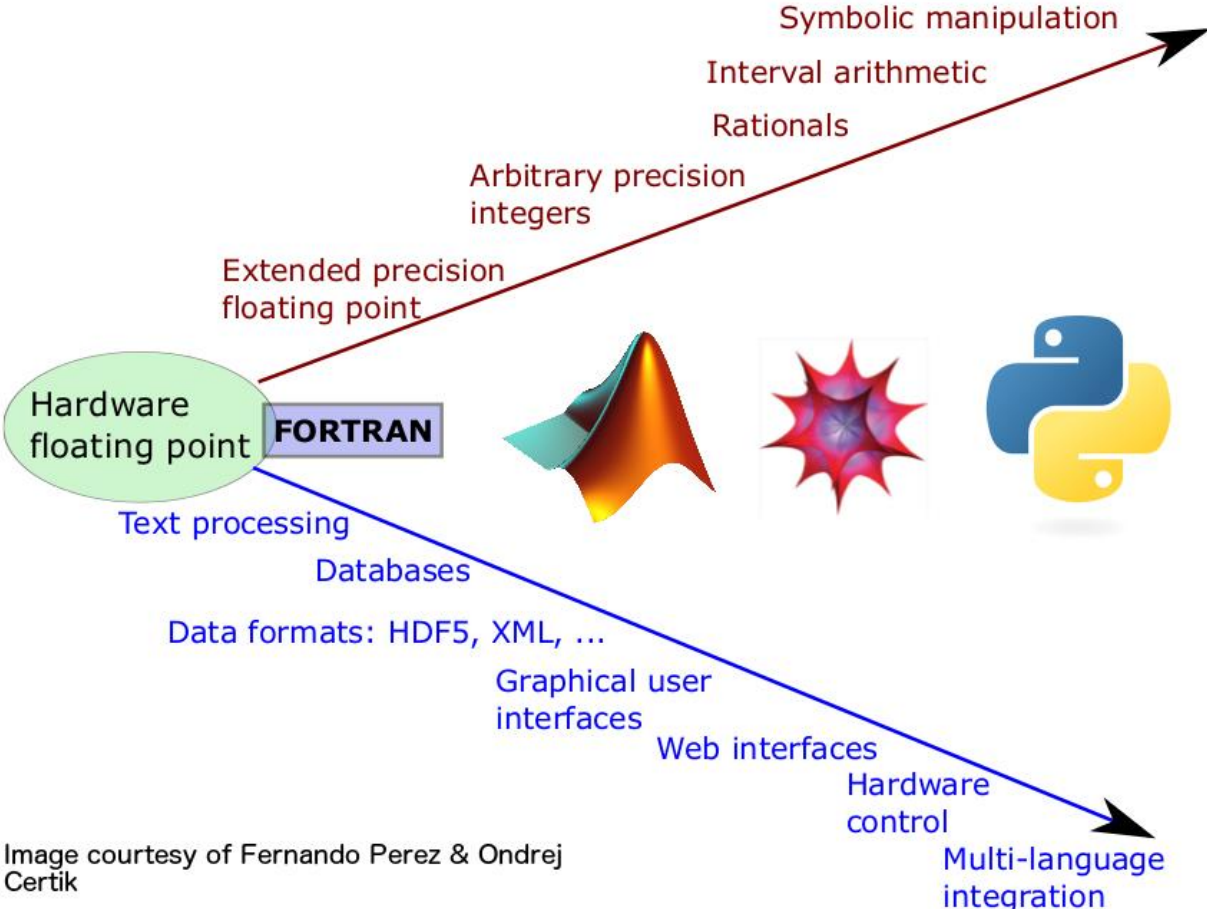


Image courtesy of Fernando Perez & Ondrej Certik

## Chapter 2: The NumPy ndarray Object

```
>>> a[0,3:5]
array([3,4])
```

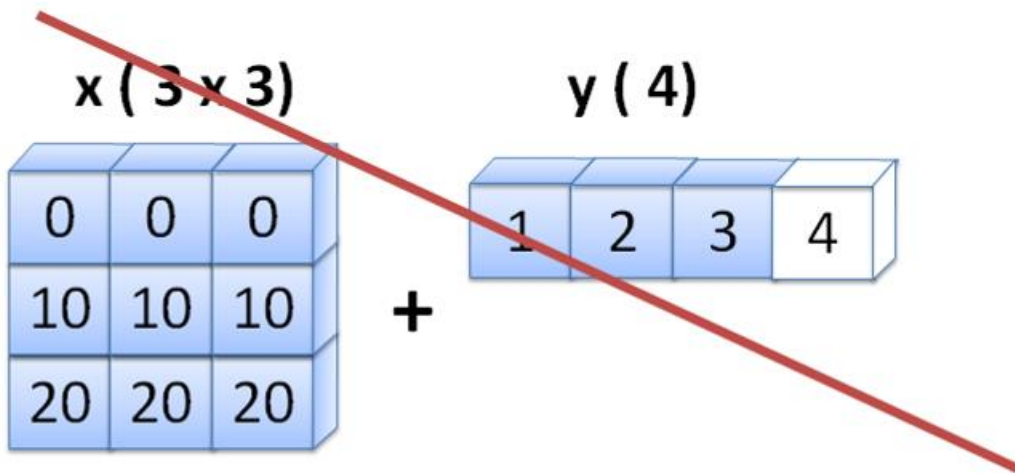
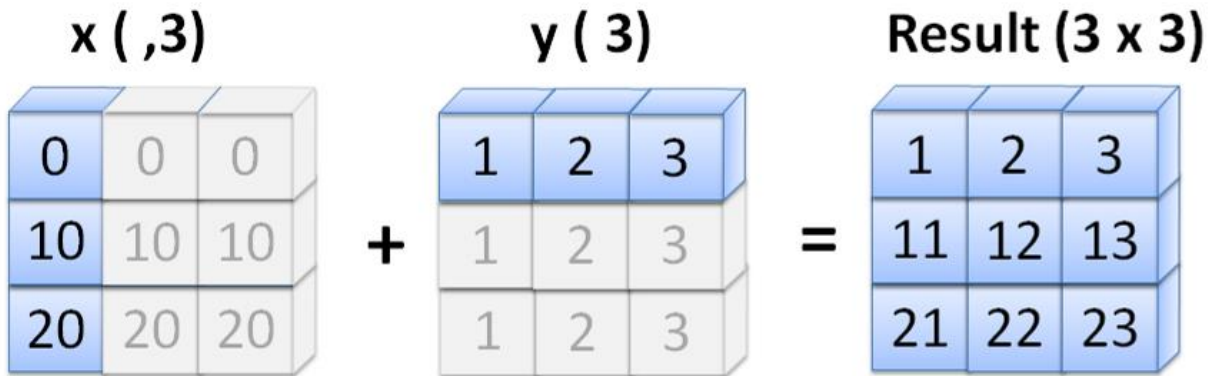
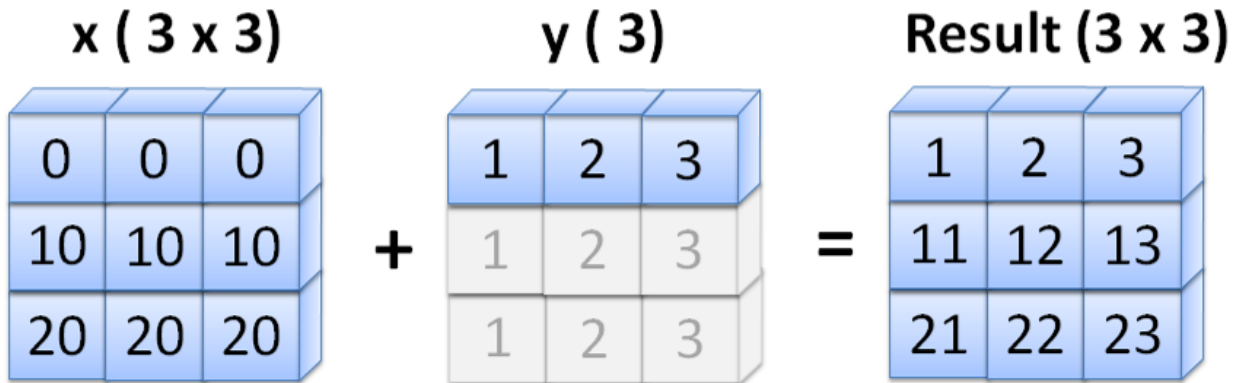
```
>>> a[4:,4:]
array([[44, 45],
       [54, 55]])
```

```
>>> a[:,2]
array([2,12,22,32,42,52])
```

```
>>> a[2::2,::2]
array([[20,22,24]
       [40,42,44]])
```

0	1	2	3	4	5
10	11	12	13	14	15
20	21	22	23	24	25
30	31	32	33	34	35
40	41	42	43	44	45
50	51	52	53	54	55

### Chapter 3: Using NumPy Arrays



# Chapter 4: NumPy Core and Libs Submodules



CPU Cache Size



A solid red horizontal line is drawn below the text 'CPU Cache Size', extending across the width of the array X.



	A	B	C
1	0	0.0702	7/10/2014
2	1	0.4863	12/3/2014
3	2	0.9525	3/11/2014
4	3	0.3971	1/2/2014
5	4	0.8537	9/14/2014
6	5	0.2176	4/24/2014

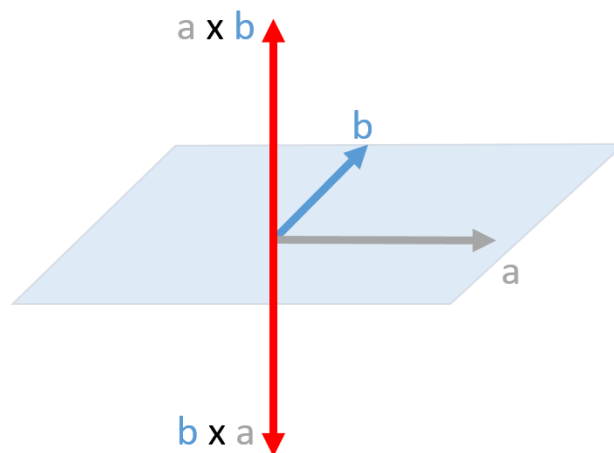
## Chapter 5: Linear Algebra in NumPy

$$\begin{bmatrix} 1*10+2*30 & 1*20+2*40 \\ 3*10+4*30 & 3*20+4*40 \end{bmatrix} = \begin{bmatrix} 70 & 100 \\ 150 & 220 \end{bmatrix}$$

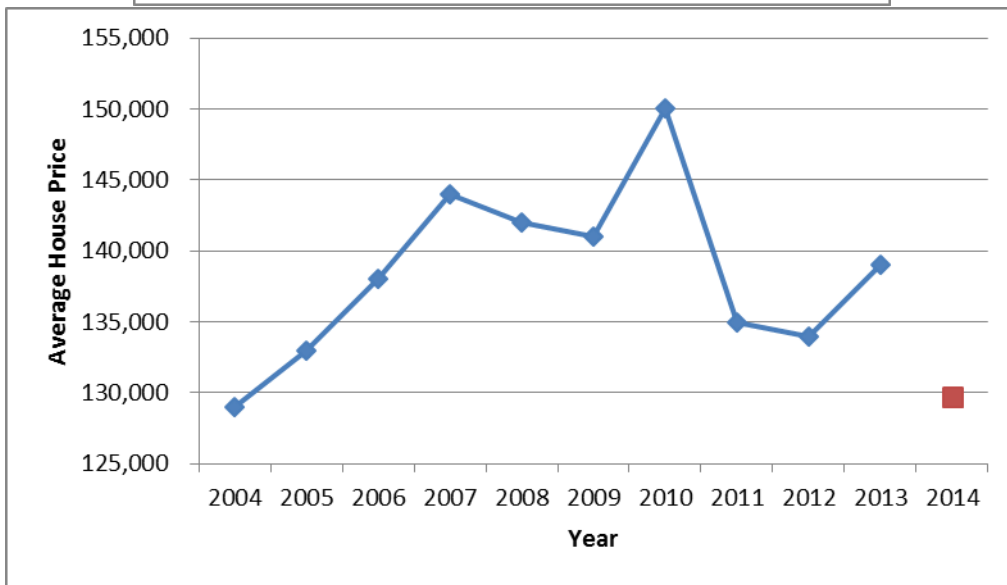
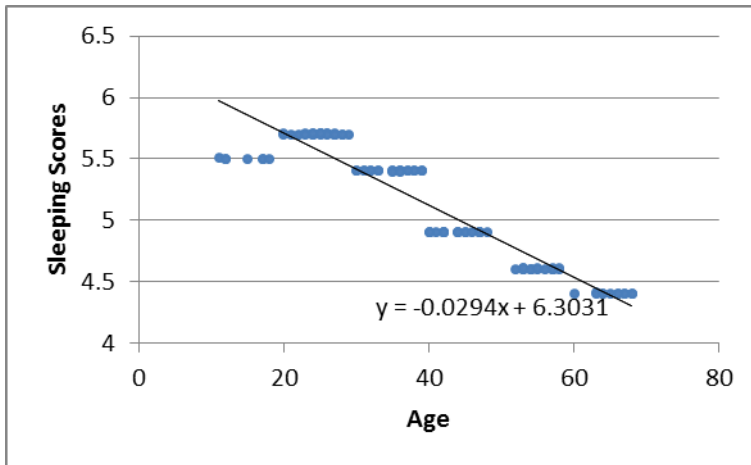
$$\begin{bmatrix} 1*10+2*30 & 1*20+2*40 \\ 3*10+4*30 & 3*20+4*40 \end{bmatrix}$$

$$1*10+2*20+3*30+4*40=300$$

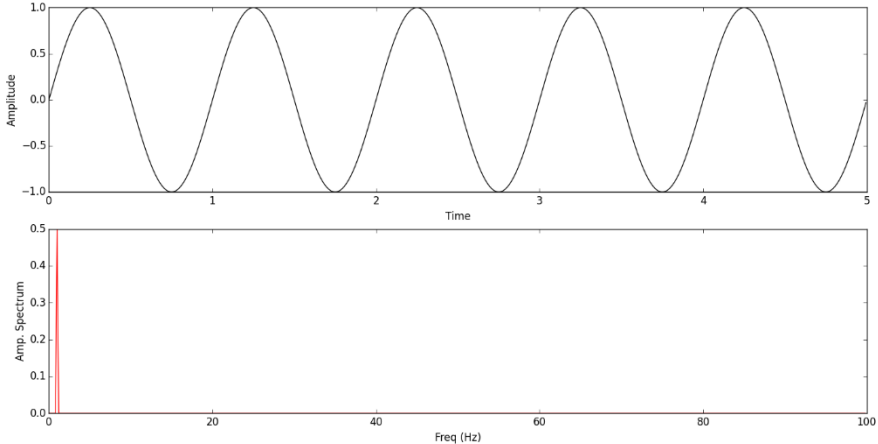
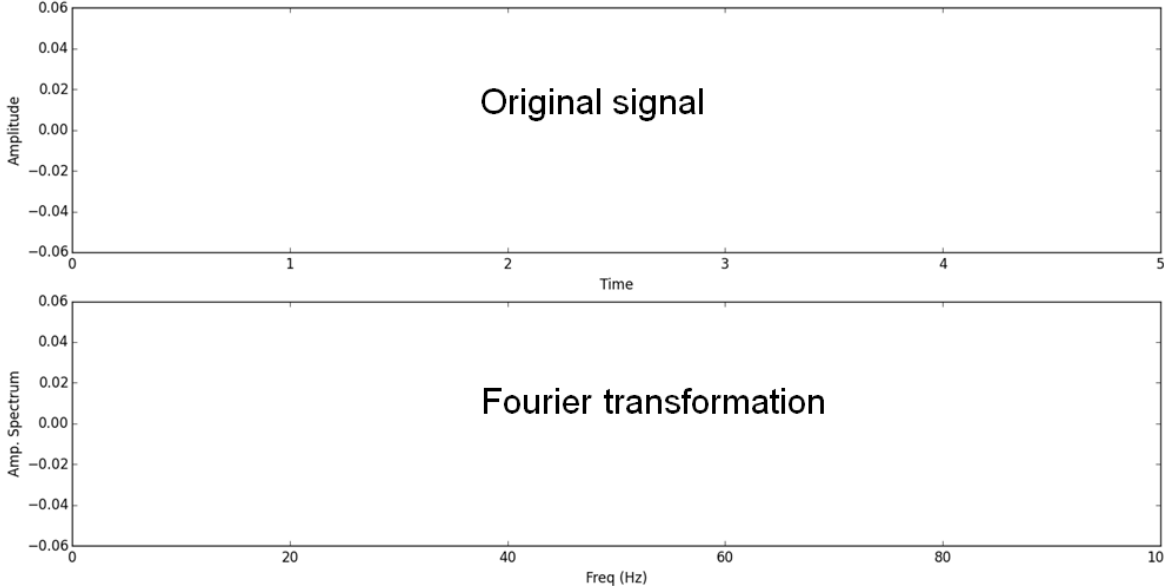
$$\begin{bmatrix} 1*10 & 1*20 & 1*30 & 1*40 \\ 2*10 & 2*20 & 2*30 & 2*40 \\ 3*10 & 3*20 & 3*30 & 3*40 \\ 4*10 & 4*20 & 4*30 & 4*40 \end{bmatrix} = \begin{bmatrix} 10 & 20 & 30 & 40 \\ 20 & 40 & 60 & 80 \\ 30 & 60 & 90 & 120 \\ 40 & 80 & 120 & 160 \end{bmatrix}$$



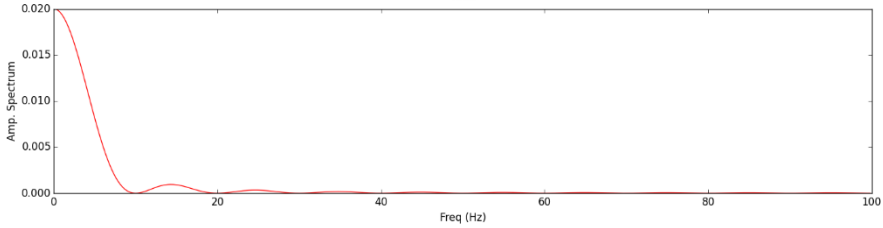
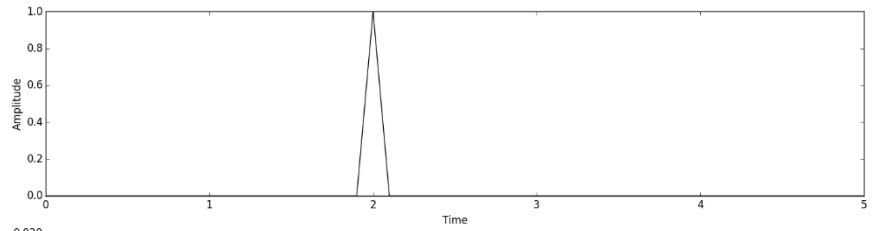
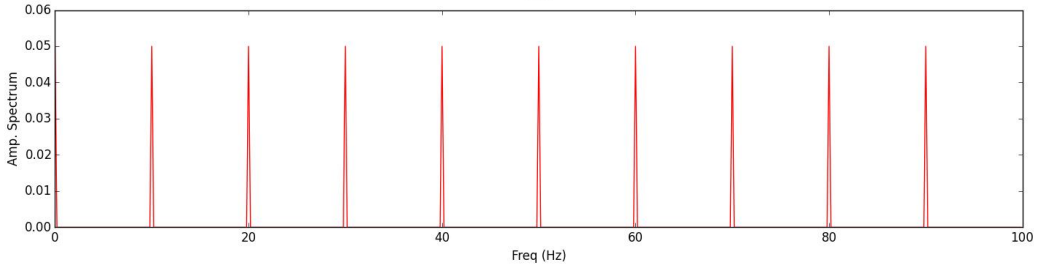
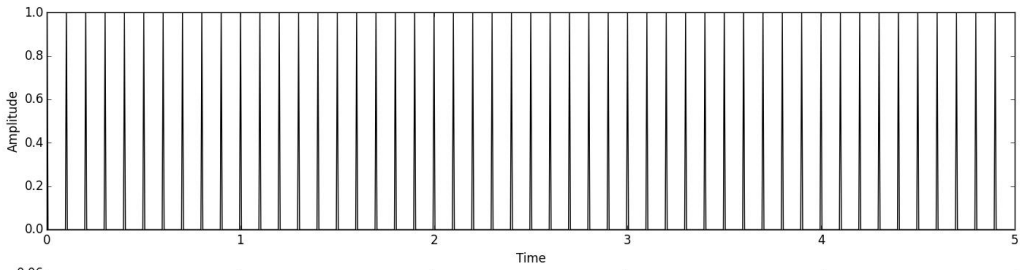
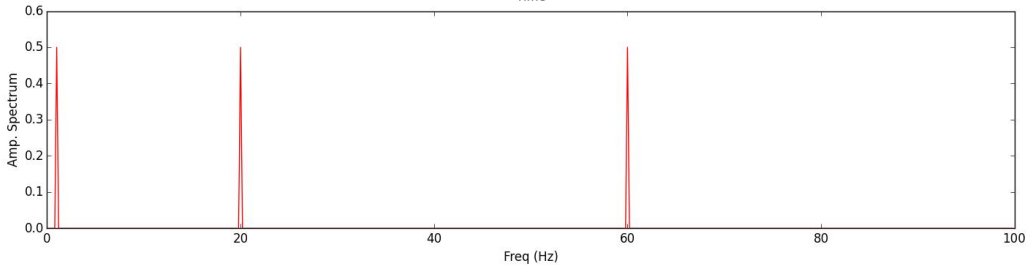
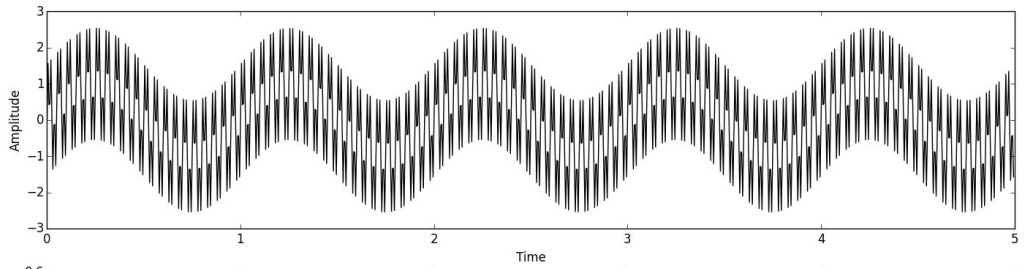
$$\frac{1}{5}x^5 - \frac{1}{4}x^4 + \frac{35}{3}x^3 - 25x^2 + 24x$$

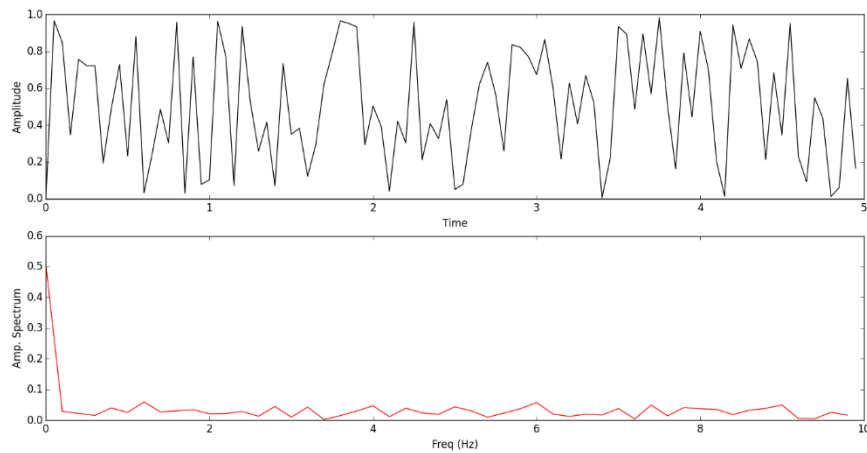


# Chapter 6: Fourier Analysis in NumPy







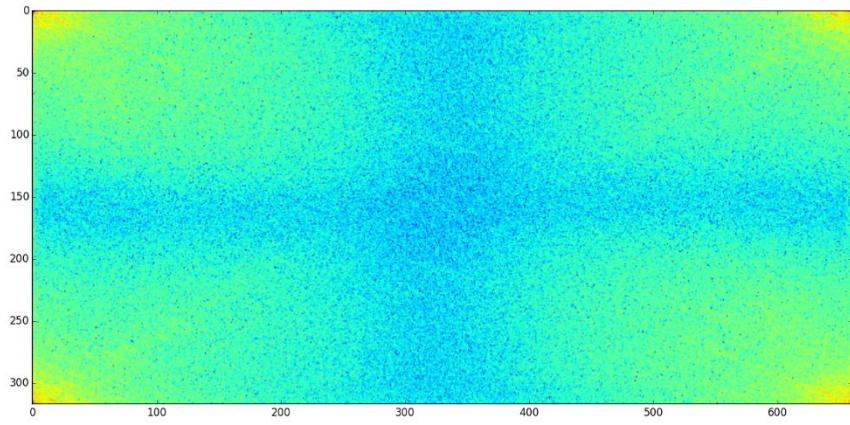
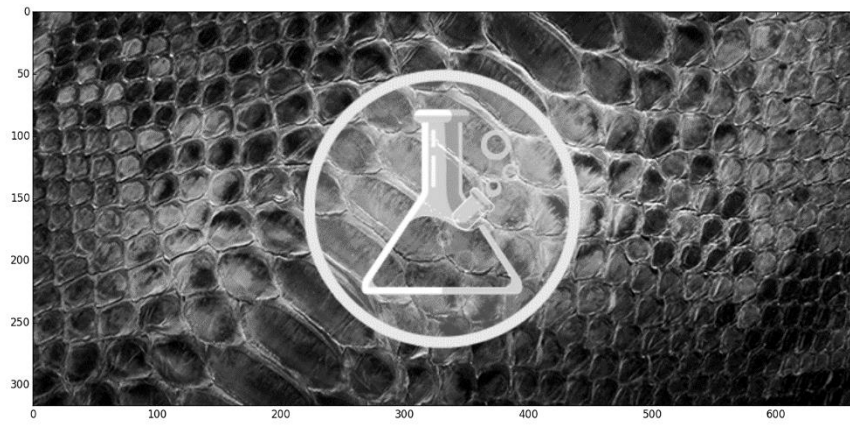


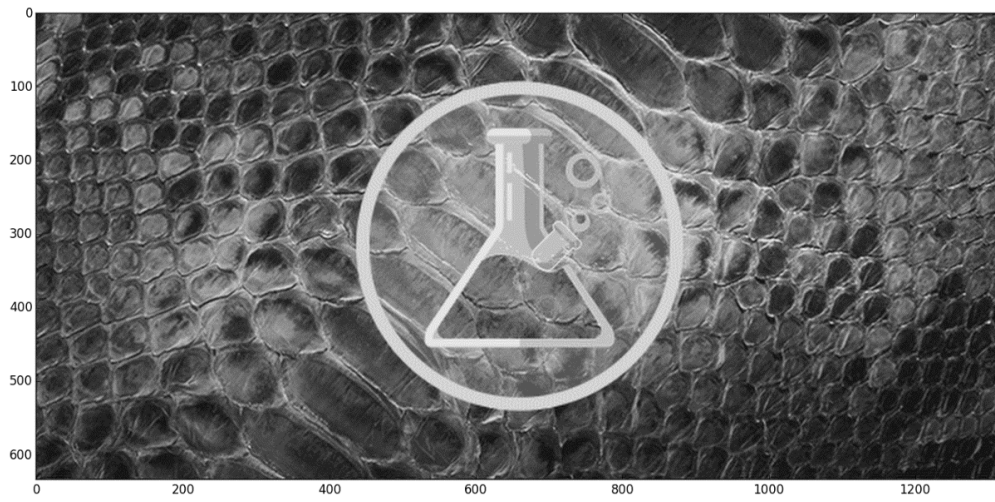
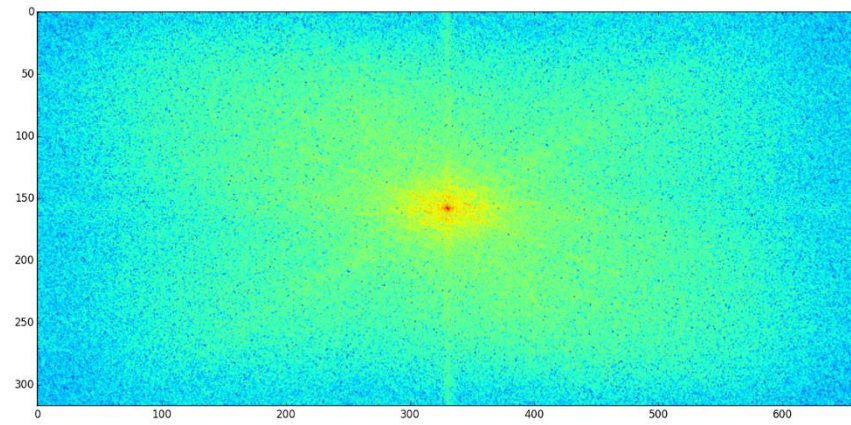
$$A_k = \sum_{m=0}^{n-1} a_m \exp\left\{-2\pi i \frac{mk}{n}\right\} \quad k = 0, 1, \dots, n-1 \quad \exp\left\{-2\pi i \frac{mk}{n}\right\} \exp\left\{-2\pi i \frac{mk}{n}\right\}$$

$$a_m = \frac{1}{n} \sum_{k=0}^{n-1} A_k \exp\left\{2\pi i \frac{mk}{n}\right\} \quad m = 0, 1, \dots, n-1$$

$$A_{kl} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} a_{mn} \exp\left\{-2\pi i \left(\frac{mk}{M} + \frac{nl}{N}\right)\right\} \quad k = 0, 1, \dots, M-1; l = 0, 1, \dots, N-1$$



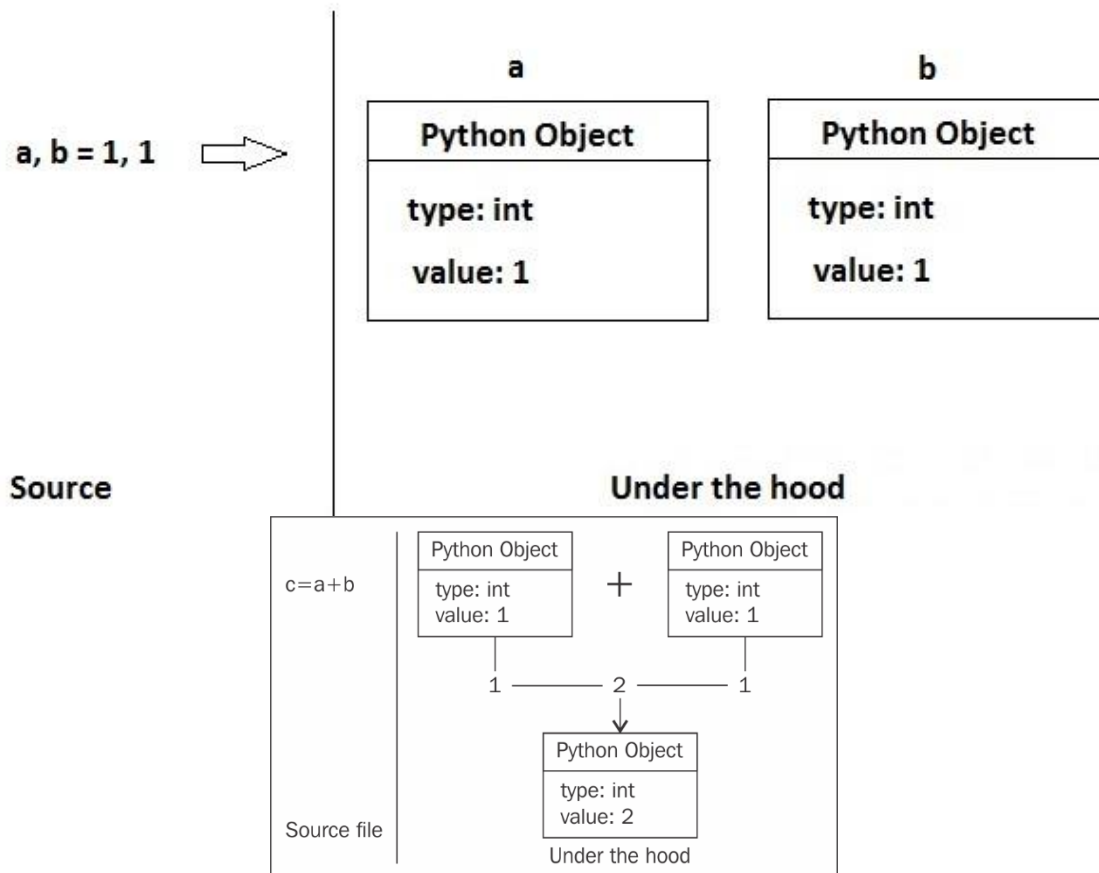




## Chapter 7: Building and Distributing NumPy Code

```
~/py_hello $ python setup.py sdist
running sdist
running egg_info
writing requirements to py_hello.egg-info/requires.txt
writing py_hello.egg-info/PKG-INFO
writing top-level names to py_hello.egg-info/top_level.txt
writing dependency links to py_hello.egg-info/dependency_links.txt
reading manifest file 'py_hello.egg-info/SOURCES.txt'
reading manifest template 'MANIFEST.in'
warning: no files found matching 'README.txt'
warning: no files found matching '*.py' under directory '*.txt'
writing manifest file 'py_hello.egg-info/SOURCES.txt'
running check
creating py_hello-0.1.0
creating py_hello-0.1.0/bin
creating py_hello-0.1.0/greeter
creating py_hello-0.1.0/py_hello.egg-info
making hard links in py_hello-0.1.0...
hard linking MANIFEST.in -> py_hello-0.1.0
hard linking README -> py_hello-0.1.0
hard linking setup.py -> py_hello-0.1.0
hard linking bin/greeter.bat -> py_hello-0.1.0/bin
hard linking greeter/__init__.py -> py_hello-0.1.0/greeter
hard linking greeter/greeter.py -> py_hello-0.1.0/greeter
hard linking py_hello.egg-info/PKG-INFO -> py_hello-0.1.0/py_hello.egg-info
hard linking py_hello.egg-info/SOURCES.txt -> py_hello-0.1.0/py_hello.egg-info
hard linking py_hello.egg-info/dependency_links.txt -> py_hello-0.1.0/py_hello.e
gg-info
hard linking py_hello.egg-info/requires.txt -> py_hello-0.1.0/py_hello.egg-info
hard linking py_hello.egg-info/top_level.txt -> py_hello-0.1.0/py_hello.egg-info
Writing py_hello-0.1.0/setup.cfg
Creating tar archive
~/py_hello $ python
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import greeter.greeter
>>> import greeter.greeter as g
>>> g.greet()
Hi tanmay, you are awesome
>>>
```

## Chapter 8: Speeding Up NumPy with Cython



```
In [6]: timeit(compute_fibonacci(20))
100000 loops, best of 3: 1.86 µs per loop
```

```
In [11]: timeit(compute_fibonacci_cython(20))
10000000 loops, best of 3: 64.5 ns per loop
```

```
C:\dev\packtcode>python setup.py build_ext --inplace
running build_ext
skipping 'first.c' Cython extension (up-to-date)
building 'testing' extension
C:\Anaconda\Scripts\gcc.bat -DMS_WIN64 -mdll -O -Wall -IC:\Anaconda\include -IC:\Anaconda\PC -c first.c -o build\temp.win-amd64-2.7\Release\first.o
writing build\temp.win-amd64-2.7\Release\testing.def
C:\Anaconda\Scripts\gcc.bat -DMS_WIN64 -shared -s build\temp.win-amd64-2.7\Release\first.o build\temp.win-amd64-2.7\Release\testing.def -LC:\Anaconda\libs -LC:\Anaconda\PCbuild\amd64 -lpython27 -lnsvcr90 -o C:\dev\packtcode\testing.pyd
Cannot export inittesting: symbol not defined
collect2.exe: error: ld returned 1 exit status
error: command 'C:\Anaconda\Scripts\gcc.bat' failed with exit status 1
```

## **Chapter 9: Introduction to the NumPy C-API (No Images)**

# Chapter 10: Further Reading

