Jonas van den Brink, jvbrink

j.v.d.brink@fys.uio.no

3.1

```
#!/usr/bin/env python
import numpy as np
import sys

usage = 'Usage: %s outfile dt **infiles' % sys.argv[0]

# Try reading cmd-line arguments
try:
  outfile = sys.argv[1]
  dt = sys.argv[2]
  infiles = sys.argv[3:]
except:
  print usage; sys.exit(1)

# Open outfile for writing
outfile = open(outfile+'.dat', 'w')
outfile.write('Outfile for inverseconvert.py\n%s\n' % dt)

# Read in data from infiles
data = []
for i in range(len(infiles)):
  outfile.write("  " + infiles[i][:-4])
  data.append(np.loadtxt(infiles[i])[:,1])

# Write data to new outfile
outfile.write("\n")
for i in range(len(data)):
  for j in range(len(data[0])):
    length = (len(infiles[j])-6)/2
    outfile.write(' '*length + str(data[j][i]) + ' '*(length+2))
  outfile.write('\n')

'''
user$ python inverseconvert.py outfile 1.5 \
tmp-measurements.dat  tmp-model1.dat  tmp-model2.dat

user$ more outfile.dat
Outfile for inverseconvert.py
1.5
  tmp-measurements  tmp-model1  tmp-model2
       0.0             0.1          1.0
       0.1             0.1          0.188
       0.2             0.2          0.25
'''
```

3.2

```
def add_date(string):
  """Appends current system date to a given string"""
  from time import ctime

  # Get current system date
  now = ctime()
  month = now[4:7]
  mday  = now[8:10]
  year  = now[20:24]
  append = "_%s%s_%s" % (month, mday, year)

  return string+append

if __name__ == '__main__':
  print add_date('myfile')

'''
user$ python add_date.py
myfile_Sep11_2013
'''
```

3.3

```
# Read in data from file
infile = open('efficiency.test', 'r')
lines = infile.readlines()
infile.close()

# Sort out lines with CPU-time data (and remove '\n')
cpulines = [l[:-1] for l in lines if l[:8]=="CPU-time"]

# Sort with respect to time
cpulines.sort(key=lambda line: float(line.split()[1]))

# Print out resulting order
print '\n'.join([line for line in cpulines])

'''
user$ python ranking.py
CPU-time:   5.41   g77 -O3 -ffast-math -funroll-loops original (optimal
    ?) code
CPU-time:   5.55   g77 -O3 original (optimal?) code
CPU-time:   5.62   g77 -O2 original (optimal?) code
...
CPU-time: 255.97   f95 -O0 formatted I/O
CPU-time: 272.90   g77 -O0 formatted I/O
'''
```