```python
from astropy.table import Table
import matplotlib.pyplot as plt
import numpy as np
from scipy import stats

t=Table.read('/home/mljones1/table.tbl',format='ipac')

#Part B
j=t['mag_j']
h=t['mag_h']
k=t['mag_ks']

mag3_6 = t['mag3_6']
mag4_5 = t['mag4_5']
mag5_8 = t['mag5_8']
mag8_0 = t['mag8_0']

total = 7563

Arr_j_k = []
Arr_36 = []
Arr_45 = []
Arr_58 = []
Arr_80 = []
Arr_h_k=[]
Arr_k=[]
for i in xrange(0,total):
  if j.mask[i] == False and h.mask[i] == False and k.mask[i] == False and mag3_6.mask[i] == False and
mag4_5.mask[i] == False and mag5_8.mask[i] == False and mag8_0.mask[i] == False:
    Arr_k.append(k[i])
    Arr_h_k.append(h[i]-k[i])
    Arr_j_k.append(j[i]-k[i])
    Arr_36.append(mag3_6[i]-k[i])
    Arr_45.append(mag4_5[i]-k[i])
    Arr_58.append(mag5_8[i]-k[i])
    Arr_80.append(mag8_0[i]-k[i])

Arr_j_k=sorted(Arr_j_k)
Arr_h_k=sorted(Arr_h_k)
Arr_k=sorted(Arr_k)
Arr_36=sorted(Arr_36)
Arr_45=sorted(Arr_45)
Arr_58=sorted(Arr_58)
Arr_80=sorted(Arr_80)

slope_36, intercept, r_value, p_value, std_err = stats.linregress(Arr_j_k,Arr_36)
slope_45, intercept, r_value, p_value, std_err = stats.linregress(Arr_j_k,Arr_45)
slope_58, intercept, r_value, p_value, std_err = stats.linregress(Arr_j_k,Arr_58)
slope_80, intercept, r_value, p_value, std_err = stats.linregress(Arr_j_k,Arr_80)
slope_J, intercept, r_value, p_value, std_err = stats.linregress(Arr_j_k,Arr_j_k)
slope_H, intercept, r_value, p_value, std_err = stats.linregress(Arr_j_k,Arr_h_k)


#Part C

A_36 = (1.5)*slope_36+1
A_45 = (1.5)*slope_45+1
A_58 = (1.5)*slope_58+1
A_80 = (1.5)*slope_80+1
A_J= (1.5)*slope_J+1
A_H = (1.5)*slope_H+1


#Part D

plt.plot([1.24,1.664,3.6,4.5,5.8,8.0],[A_J,A_H,A_36,A_45,A_58,A_80])
```

```python
plt.loglog()
plt.ylim([1,3])
plt.xlim([1,10])
plt.xlabel('$\lambda$ (microns)')
plt.ylabel('A${_[\lambda]}$/A$_k$')
plt.show()
```