

# wrapper

November 24, 2017

```
In [9]: #mylist = !(ls | egrep *.events )

In [10]: testdirs = !(ls | egrep test ) # to get all dirs
mylist={}
for dir in testdirs :
    x = !(ls $dir | egrep *.events )# each element of the list represent an implementat
mylist[dir[16:]] = (dir,x)

In [11]: %matplotlib inline
import matplotlib.pyplot as plt
import numpy as np

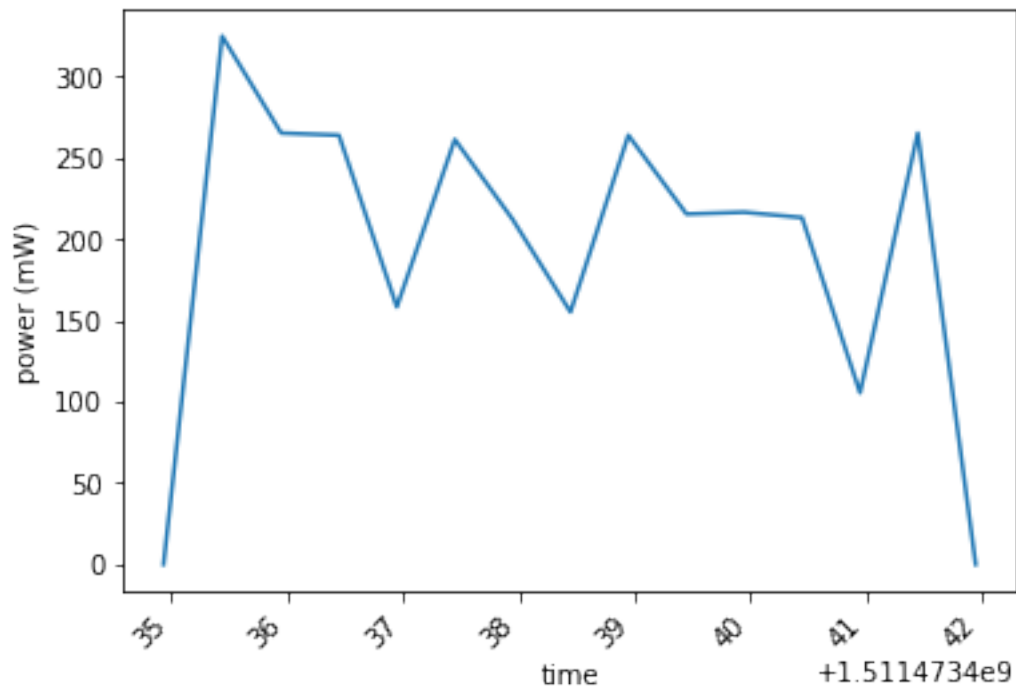
def extractfile(name):
    """ return a dict which contains
    1- the timestamp of start
    2- time stamp of the end
    3- an array of timestamps of mesures
    4- an array of the mesures
    this function is based on the format of the file so anychanges of the structure of
    """
    with open(name) as f :
        x = [i.split(' ') for i in f.read().split('\n')]
        tstart = int(x[1][1][1:]) *1000 # transform them into ms
        tend = int(x[2][1][1:]) *1000 #
        ind = x.index(["##--##"])+1
        times = [int(i[0][10:-1]) for i in x[ind:][0:-1:2]] ## we added -1 due to the l
        powers = [float(i[0][6:]) for i in x[ind:][1:2]]
        duration = (times[-1]- times[0])/1000
        meanpower = np.mean(powers)
        energy = sum(powers) /2
    return {'tstart' : tstart , 'tend': tend , 'times':times , 'powers':powers, 'duration':
```

## 1 a sample of hanoi tower

coded with C implementing the **recursive** solution that solve an instance of 22 disks

```
In [12]: import matplotlib.pyplot as plt
plt.xlabel('time')
plt.ylabel('power (mW)')
tempdir,tempfile = mylist['c_iter']
x = extractfile(tempdir+'/'+tempfile[1])
print('duration %d energy %d mj' % (x['duration'],x['energy']))
plt.plot([i/1000 for i in x["times"]],x["powers"])
s=plt.xticks(rotation=45 ,ha='right')
```

duration 7 energy 1460 mj



```
In [13]: def energy_mesure(language):
l=[]
tempdir,templist = mylist[language]
for tempfile in templist:
x = extractfile(tempdir+'/'+tempfile)
l.append(x['energy'])
return np.mean(l) /1000
```

## 2 infos about data

```
In [14]: %lsmagic
```

```
Out[14]: Available line magics:
%alias %alias_magic %autocall %automagic %autosave %bookmark %cat %cd %clear %

Available cell magics:
%%! %%HTML %%SVG %%bash %%capture %%debug %%file %%html %%javascript %%js %%l

Automagic is ON, % prefix IS NOT needed for line magics.
```

```
In [15]: curdir=!pwd
curdir=curdir[0].split('/')[ -1] #to get the name of the current floder
curdir=curdir.split('.') # just the infos cause his name is bank.mintours/maxtours.nbit
mintours= curdir[-3]
maxtours=curdir[-2]
nbiterations=curdir[-1]
```

we have a set of data composed of instances of hannoitower problem from {{mintours}} towers to {{maxtours}} towers and each instance is run for {{nbiterations}} times

```
In [16]: plt.xlabel('programming languages')
plt.ylabel('energy ')
keys = list(mylist.keys())
values = [energy_mesure(x) for x in keys]
axes=plt.barh(keys , values)
for v in axes :
    plt.text(v.get_width() +1, v.get_y()+v.get_height()/2 , '%.2f' %v.get_width())
```

