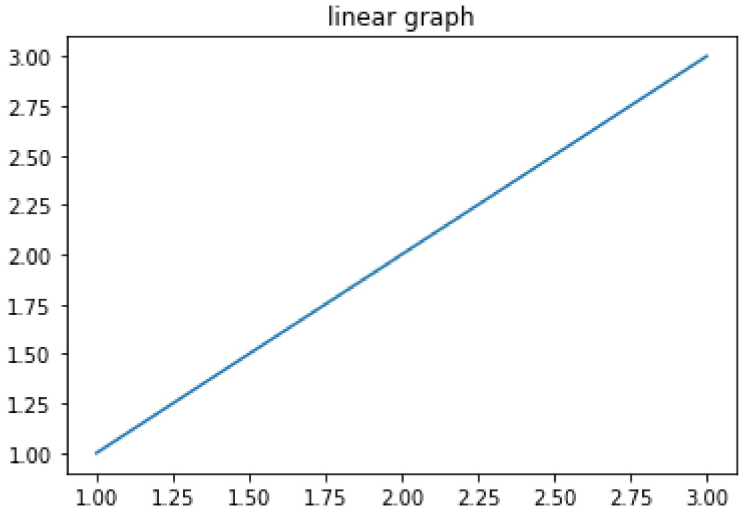
Day Objective

- Data Visualization using Matplotlib library
- Basic 2D Plotting Functions
- plotting 2D graphs from csv dataset
- Income Dataset
- App Store dataset

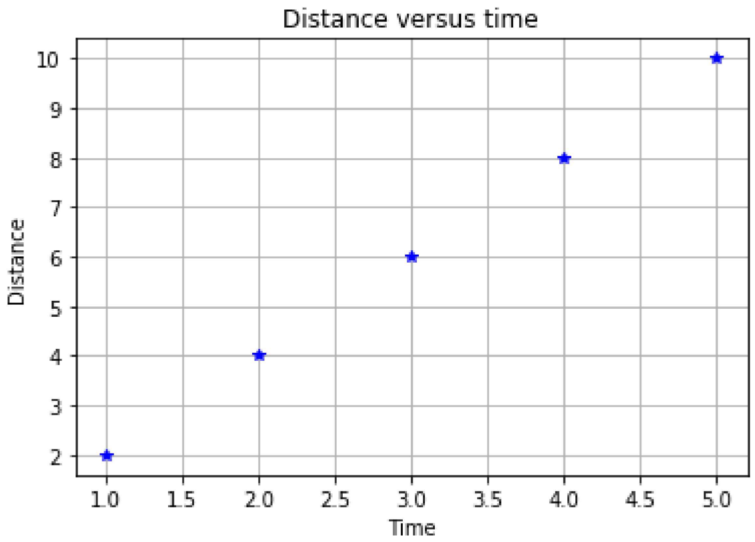
In [5]:

```
import matplotlib.pyplot as pt
pt.plot([1,2,3],[1,2,3])
pt.title("linear graph")
pt.show()
```



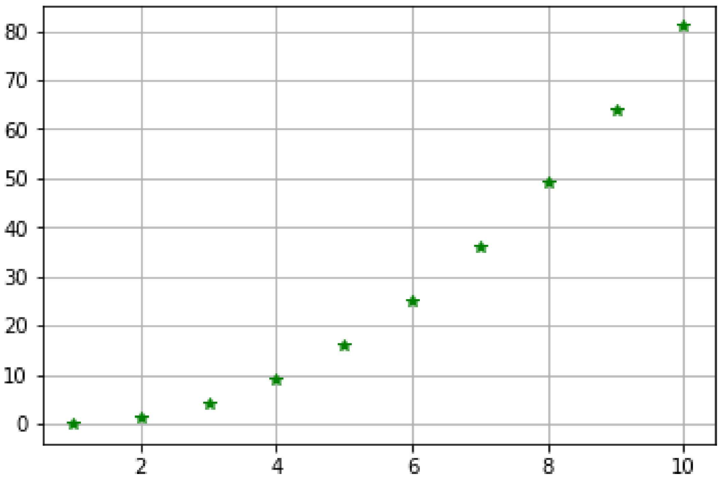
In [18]:

```
# Graph to plot distance and time
distance = list(range(2,11,2))
time =list(range(1,6))
pt.title("Distance versus time")
pt.xlabel("Time")
pt.ylabel("Distance")
pt.grid()
pt.plot(time,distance,"b*")
pt.show()
```



In [31]:

```
# v=a*t
t=list(range(1,11))
a=list(range(len(t)))
v=[a[i]*i for i in range(len(t))]
pt.plot(t,v,"g*")
pt.grid()
pt.show()
```

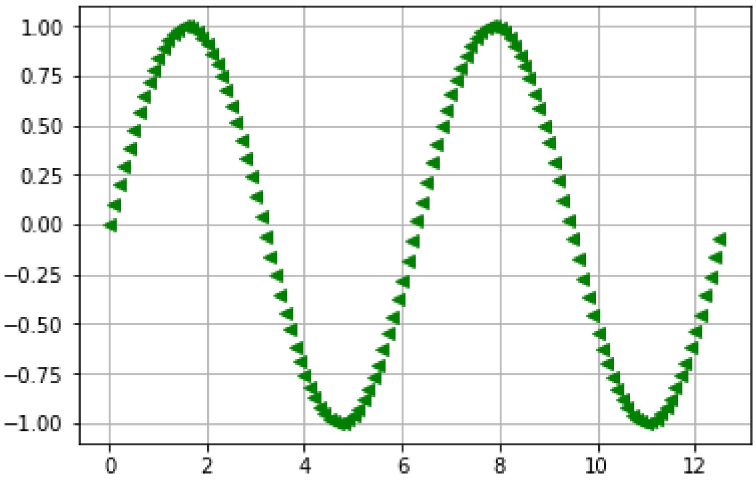


In [45]:

```
#to print a sine wave
import numpy as np
y=np.arange(0,4*np.pi,0.1)
x=np.sin(y)


pt.grid()
pt.plot(y,x,"g<")
```

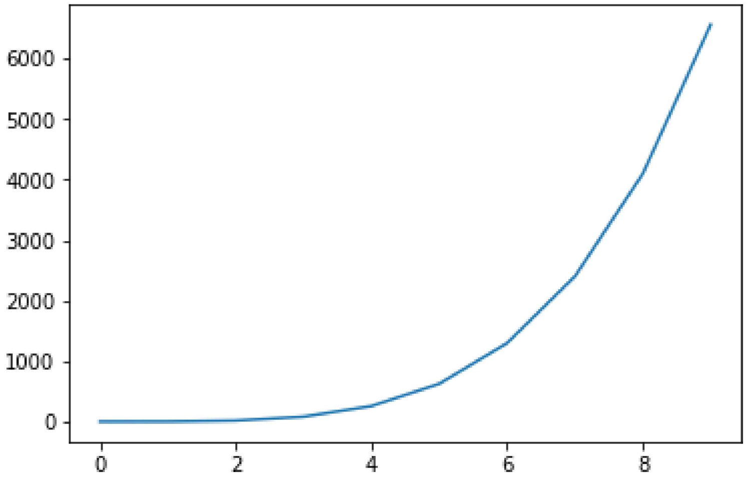
Out[45]:

```
[<matplotlib.lines.Line2D at 0x22f29709d68>]
```



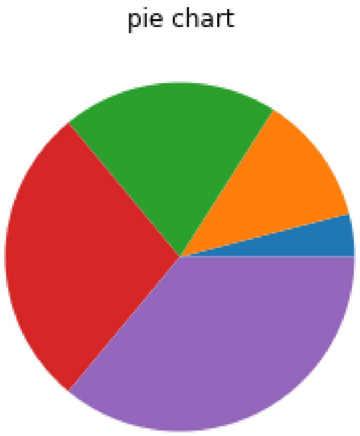
In [70]:

```


n=list(range(10))

k=[i**4 for i in (n)]

pt.plot(n,k)

pt.savefig("Images/rangegraph.png")
pt.show()
```



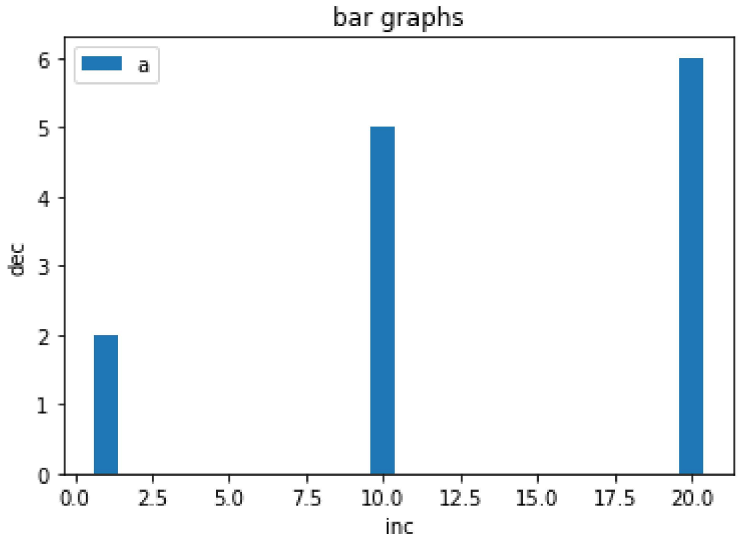
In [58]:

```
x=list(range(1,10,2))
pt.pie(x)

pt.title("pie chart")
pt.show()
```



In [71]:

```
x=[1,10,20]
y=[2,5,6]
pt.bar(x,y)
pt.title("bar graphs")
pt.xlabel("inc")
pt.ylabel("dec")
pt.legend("a")

pt.savefig("Images/bargraphs.png")
pt.show()
```



In [ ]:

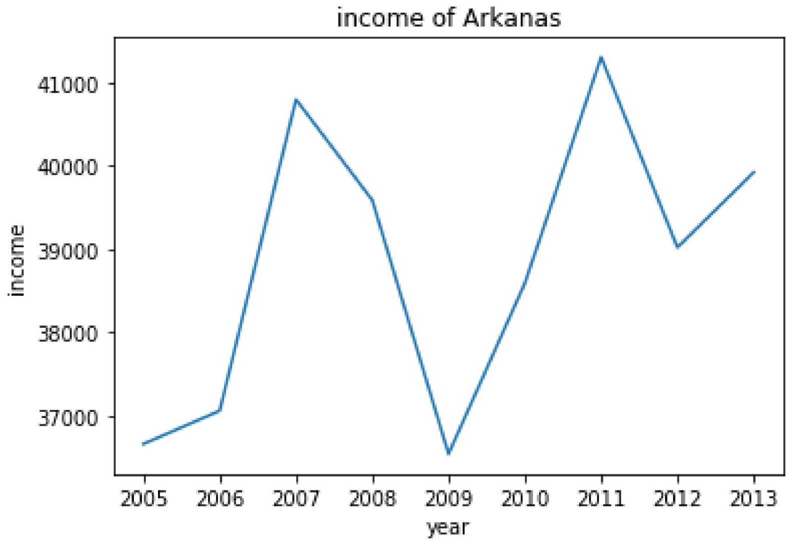
```
x=[1,10,20]
y=[2,5,6]
z=[12,34,23]

```

In [74]:

```
#plot Income of Arkanas from 2005 to 2013
import pandas as pd
df=pd.read_csv("DataFiles/Income.csv")
years=df.columns[2:]
incomeArkansas=df.values[3,2:]
pt.title("income of Arkanas")
pt.xlabel("year")
pt.ylabel("income")
pt.plot(years,incomeArkansas)
```

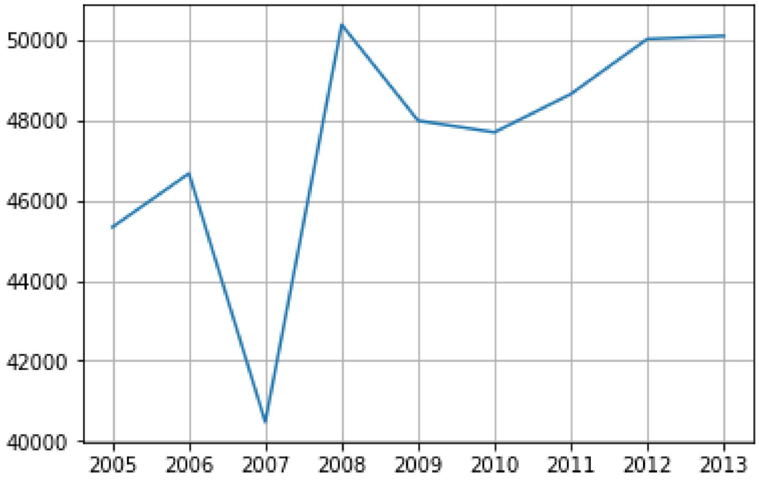
Out[74]:

```
[<matplotlib.lines.Line2D at 0x22f2babfe80>]
```



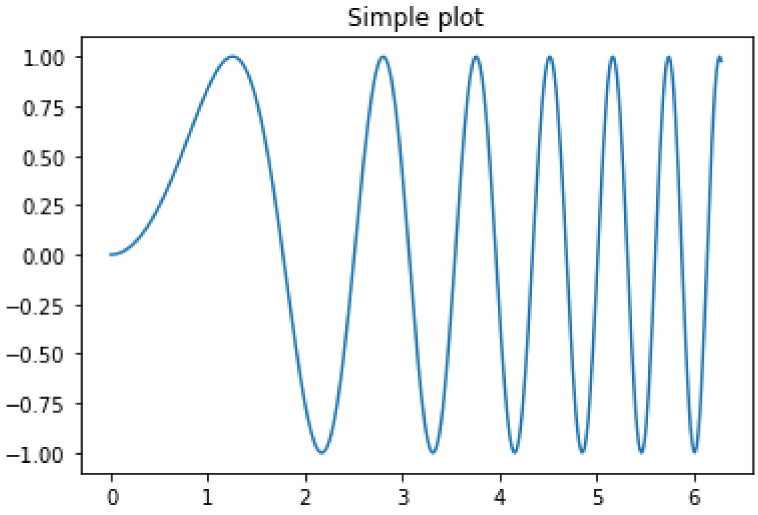
In [79]:

```
#Income of all states from 2015 to 2013
incomeallstates=df.iloc[:,2:].mean()
pt.plot(incomeallstates)
pt.grid()
pt.show()
```



In [82]:

```
# First create some toy data:
x = np.linspace(0, 2*np.pi, 400)
y = np.sin(x**2)

# Creates just a figure and only one subplot
fig, ax = pt.subplots()
ax.plot(x, y)
ax.set_title('Simple plot')

# Creates two subplots and unpacks the output array immediately
f, (ax1, ax2) = pt.subplots(1, 2, sharey=True)
ax1.plot(x, y)
ax1.set_title('Sharing Y axis')
ax2.scatter(x, y)

# Creates four polar axes, and accesses them through the returned array
fig, axes = pt.subplots(2, 2, subplot_kw=dict(polar=True))
axes[0, 0].plot(x, y)
axes[1, 1].scatter(x, y)

# Share a X axis with each column of subplots
pt.subplots(2, 2, sharex='col')

# Share a Y axis with each row of subplots
pt.subplots(2, 2, sharey='row')

# Share both X and Y axes with all subplots
pt.subplots(2, 2, sharex='all', sharey='all')

# Note that this is the same as
pt.subplots(2, 2, sharex=True, sharey=True)

# Creates figure number 10 with a single subplot
# and clears it if it already exists.
fig, ax=pt.subplots(num=10, clear=True)
```



In [ ]:

```
```