

Problem 2: Analyze battle

In this exercise, you will be analyzing the battle script output that is created in json format by the WorldOfTextCraft executable.

In [1]:

```
import numpy as np
from readbattle import readbattle
import matplotlib.pyplot as plt
```

Get the data

In [2]:

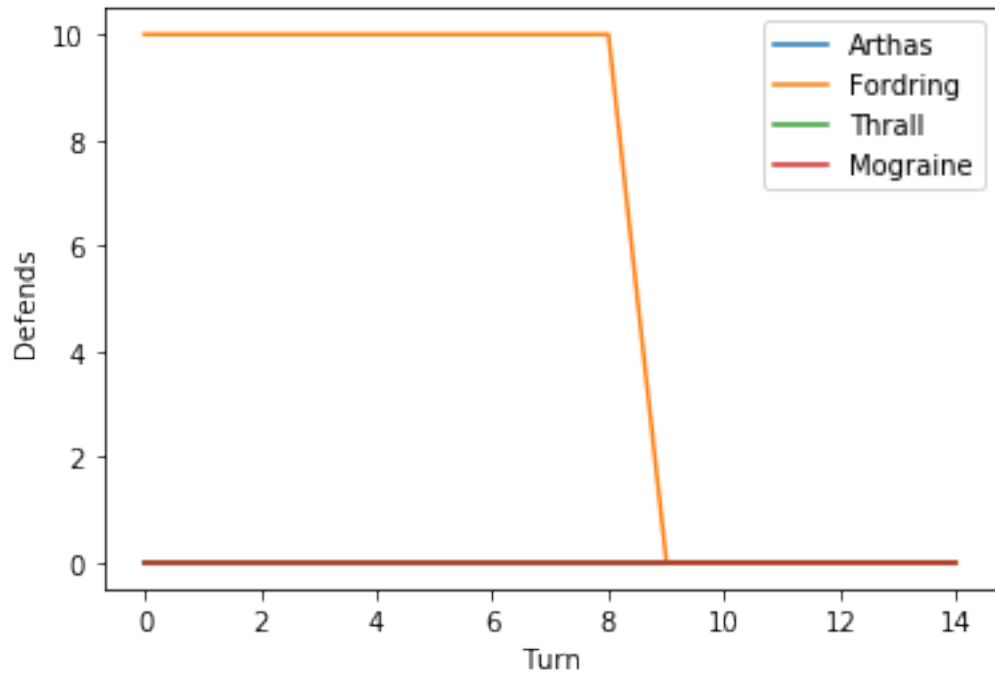
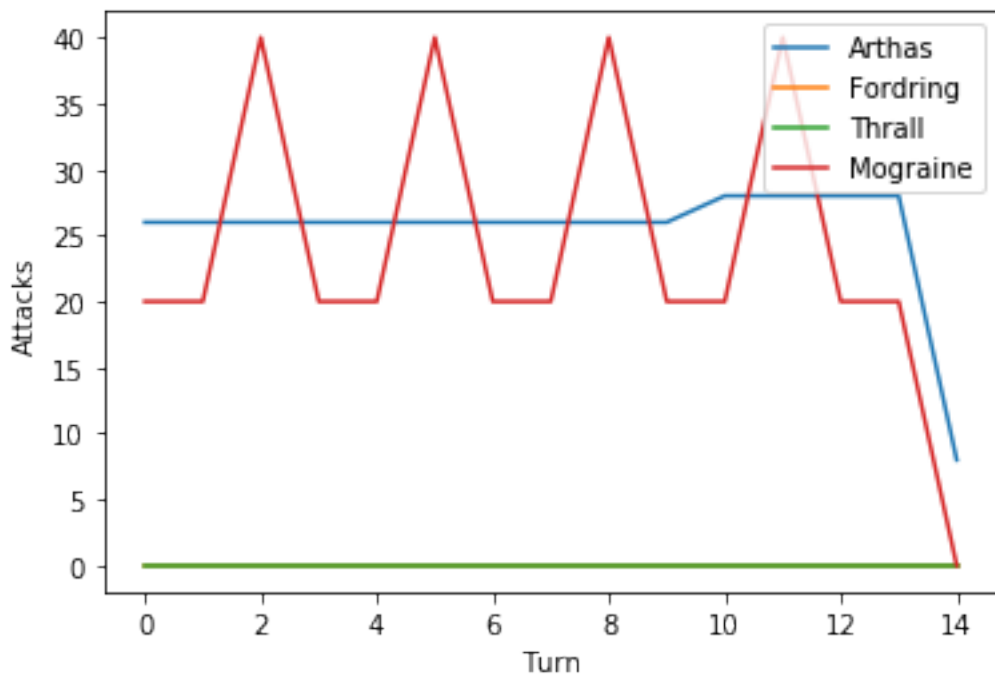
```
npdata = readbattle(infile='problem2b.txt')

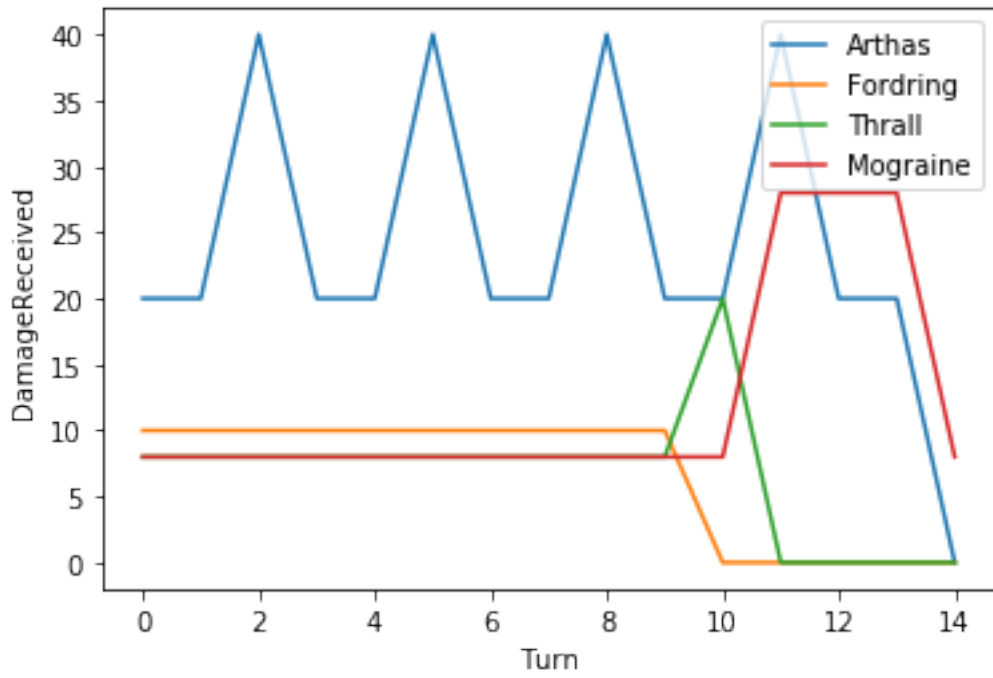
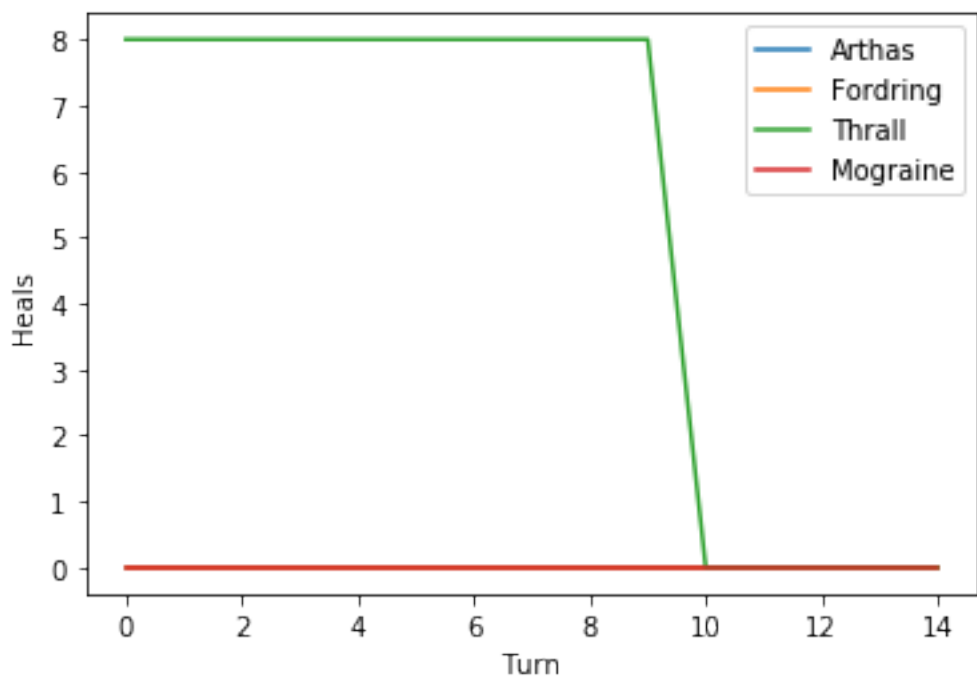
names = ["Arthas", "Fordring", "Thrall", "Mograine"]
actions = ['Attacks', 'Defends', 'Heals', 'DamageReceived', 'HealingRecieved']
```

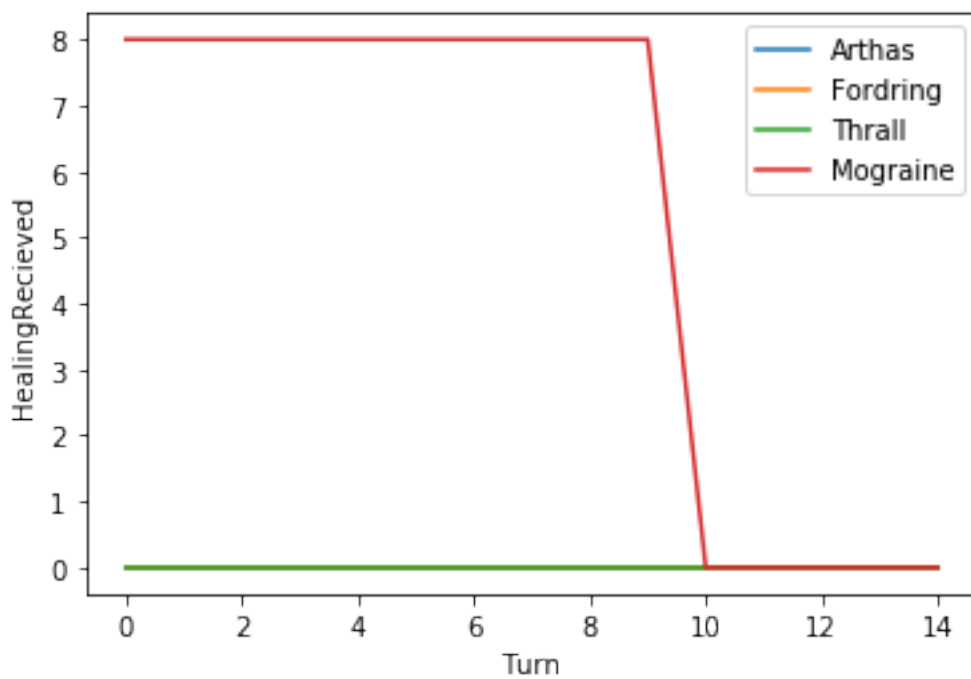
Make a plot of the attacks per turn

In [3]:

```
# Make plot of attacks.
for iact,act in enumerate(actions):
    for iname,name in enumerate(names):
        plt.plot(npdata[iname,:,iact],label=name)
plt.legend()
plt.xlabel("Turn")
plt.ylabel(act)
plt.show()
```







Part a

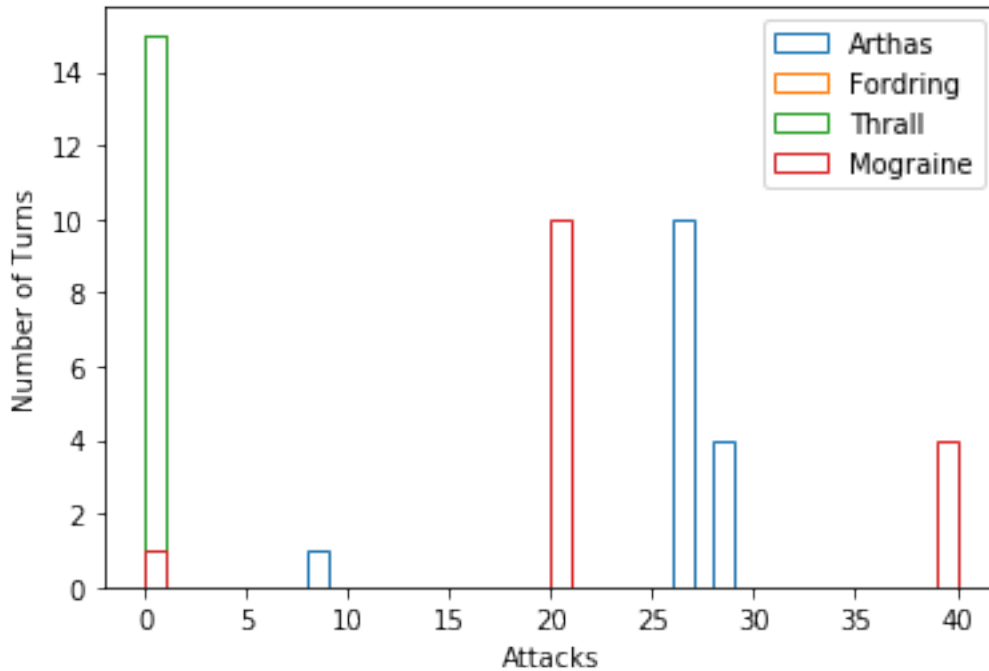
Make plots of all action types for each character: ['Attacks', 'Defends', 'Heals', 'DamageReceived', 'HealingReceived']. (The Attacks are done in the last cell as an example). Label your axes and make a legend.

Part b

Using the [matplotlib hist](https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.hist.html) (https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.hist.html) function, create a single plot showing histograms of the Attacks for each character. Label your axes and make a legend.

In [4]:

```
for iname,name in enumerate(names):
    plt.hist(npdata[iname,:,0],bins=np.linspace(0,40,41),histtype='step',label=name)
plt.legend()
plt.xlabel("Attacks")
plt.ylabel("Number of Turns")
plt.show()
```



Part c

Repeat part b, but this time plot the `Attacks` only for turns where the character was healed.

In [5]:

```
print(npdata.shape)
healedturns = npdata[:,:,:,-1] > 0
print(healedturns)
```

```
(4, 15, 5)
[[False False False False False False False False Fa
lse False False False
  False False False]
 [False False False False False False False False Fa
lse False False False
  False False False]
 [False False False False False False False False Fa
lse False False False
  False False False]
 [ True  True  True  True  True  True  True  True  T
rue  True False False
  False False False]]
```

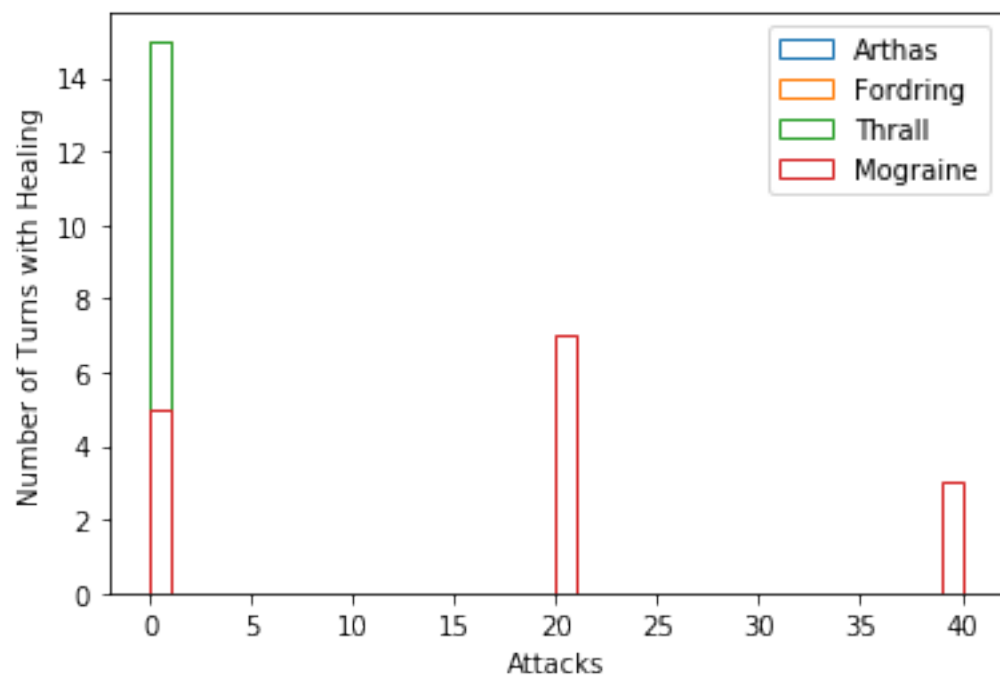
In [6]:

```
attacks_when_healed = np.where( healedturns, npdata[:,:,:0], 0)
print(attacks_when_healed)
print(attacks_when_healed.shape)
```

```
[[ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0
.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0
.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0.  0
.  0.  0.]
 [20. 20. 40. 20. 20. 40. 20. 20. 40. 20.  0.  0.  0
.  0.  0.]]
(4, 15)
```

In [7]:

```
for iname,name in enumerate(names):  
    plt.hist(attacks_when_healed[iname],bins=np.linspace(0,40,41  
) ,histtype='step',label=name)  
plt.legend()  
plt.xlabel("Attacks")  
plt.ylabel("Number of Turns with Healing")  
plt.show()
```



In []: