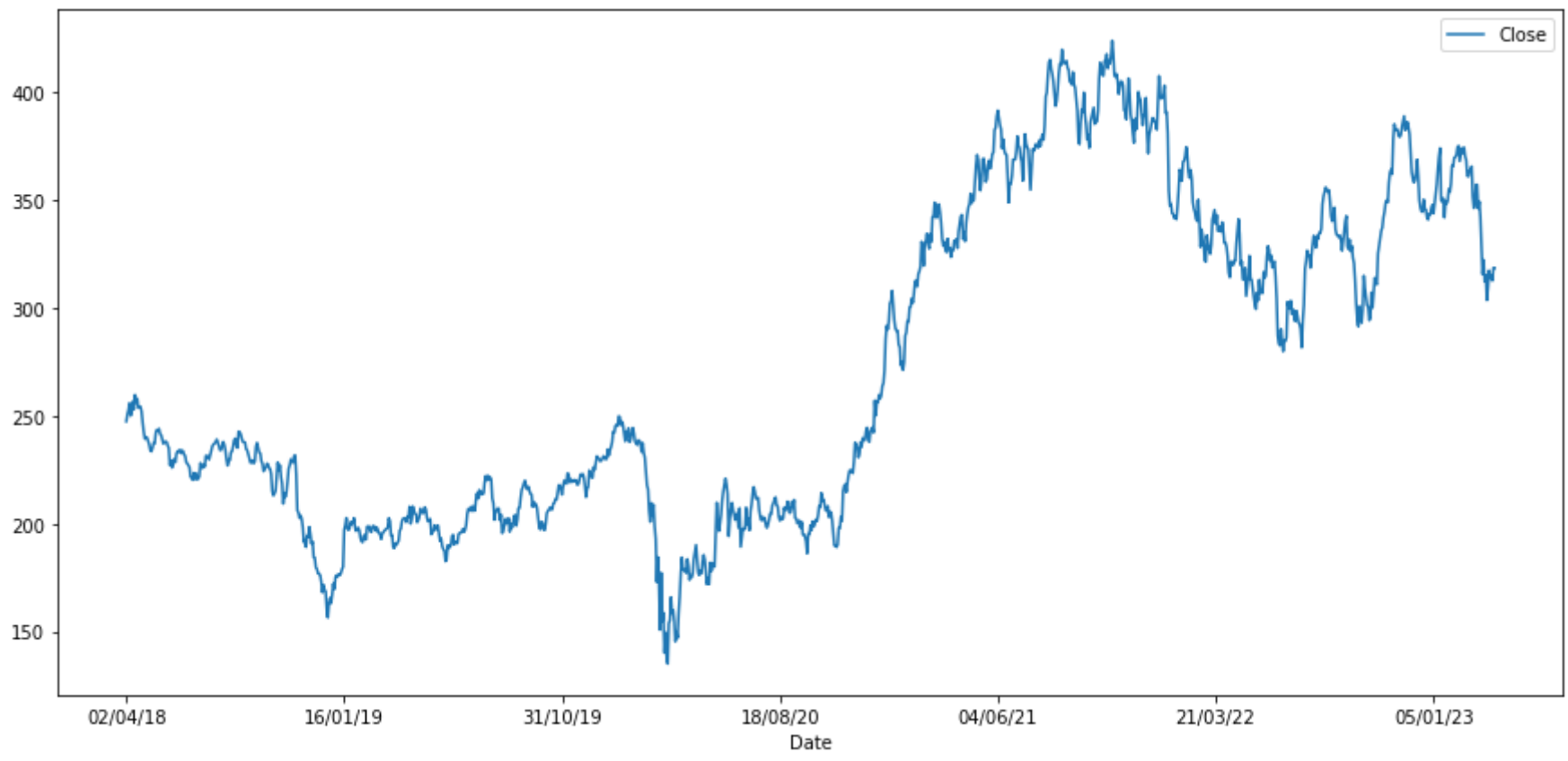
In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import warnings
warnings.filterwarnings("ignore")
```

In [2]:
```
data = pd.read_csv('GS.csv', thousands=',')
data = data[['Date', 'Close']]
data['Close'] = data['Close'].apply(lambda x: float(x))
data.head(n=10)
```

Out[2]:

	Date	Close
0	02/04/18	247.350006
1	03/04/18	250.580002
2	04/04/18	252.619995
3	05/04/18	255.809998
4	06/04/18	249.970001
5	09/04/18	252.190002
6	10/04/18	256.570007
7	11/04/18	252.940002
8	12/04/18	259.589996
9	13/04/18	255.919998

In [3]:
```
data.plot(y='Close',x='Date', figsize=(15, 7))
plt.show()
```

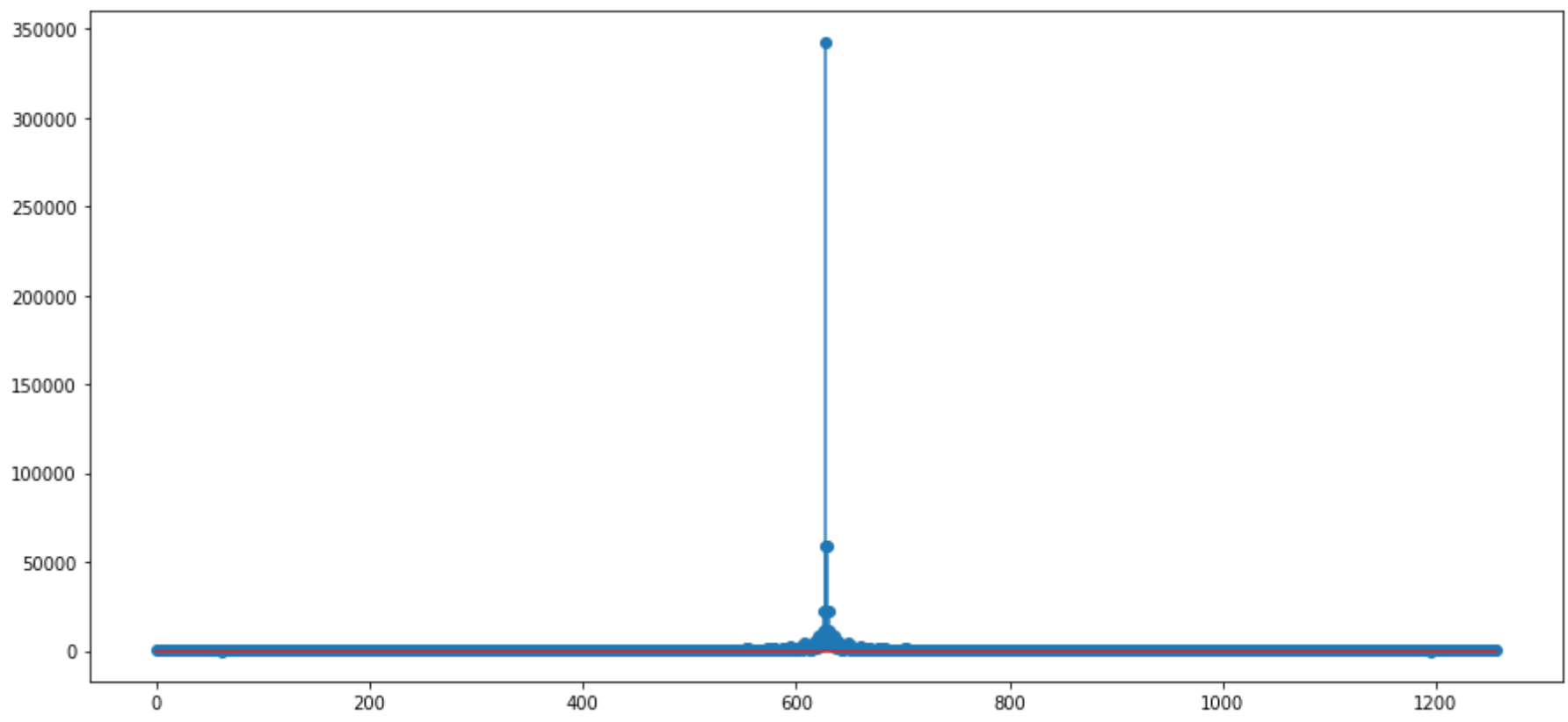


In [4]:
```
close_fft = np.fft.fft(np.asarray(data['Close'].tolist()))
fft_df = pd.DataFrame({'fft':close_fft})
fft_df['absolute'] = fft_df['fft'].apply(lambda x: np.abs(x))
fft_df['angle'] = fft_df['fft'].apply(lambda x: np.angle(x))
#fft_df.sort_values(by=['abs'], ascending=False, inplace=True)
#fft_df.reset_index(inplace=True)
#fft_df.drop(columns=['index'], inplace=True)
fft_df.head(n=20)
```

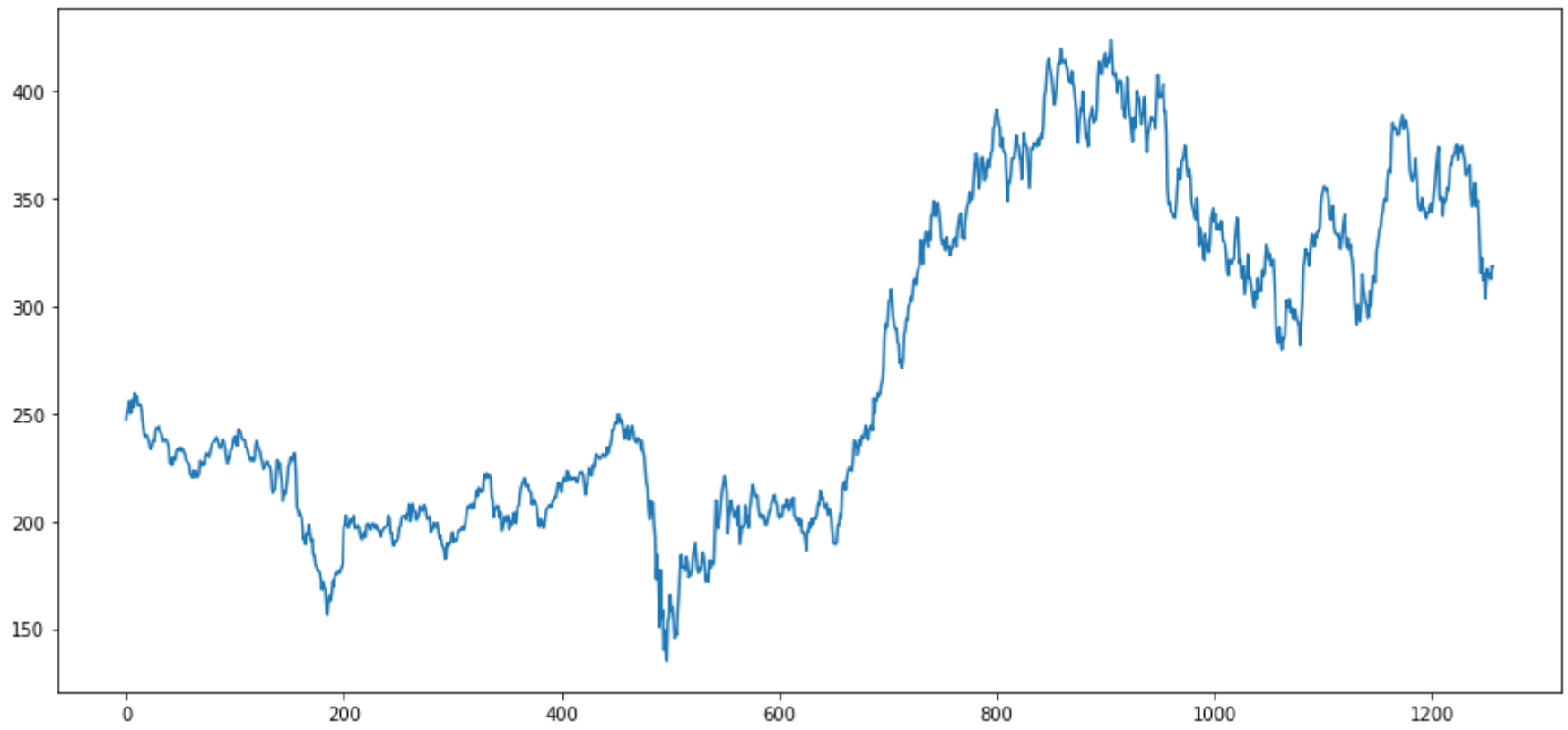
Out[4]:

	fft	absolute	angle
0	342737.2501083-0.0000000j	342737.250183	-2.280455e-17
1	9651.619703+57697.936934j	58499.621275	1.405052e+00
2	-10505.280022-3680.746434j	11131.433091	-2.804587e+00
3	20762.071923+7522.670761j	22082.893966	3.476146e-01
4	-721.406412+07386.4045630j	7421.549675	1.668154e+00
5	-1879.077310+7288.671460j	7526.995628	1.823110e+00
6	-3132.007886+3880.382065j	4986.666058	2.249875e+00
7	-2022.2288810+757.1632780j	2159.329960	2.783330e+00
8	-900.635607+08370.3841130j	8418.697921	1.677982e+00
9	1585.8885410+650.2794350j	1714.032033	3.891324e-01
10	-2025.731117+4602.262911j	5028.360614	1.985437e+00
11	2569.408552+1209.027239j	2839.649128	4.398087e-01
12	-1712.907614-1327.497930j	2167.095579	-2.482285e+00
13	-1858.529514+1173.049519j	2197.766396	2.578568e+00
14	679.9797600-431.9087850j	805.554265	-5.658855e-01
15	-457.6643560+188.7867460j	495.072821	2.750357e+00
16	3136.119054+2320.456805j	3901.251402	6.370144e-01
17	628.642861+01629.2218310j	1746.297690	1.202543e+00
18	-769.774410+03158.8670120j	3251.306421	1.809824e+00
19	372.900162+01584.6788940j	1627.962447	1.339685e+00

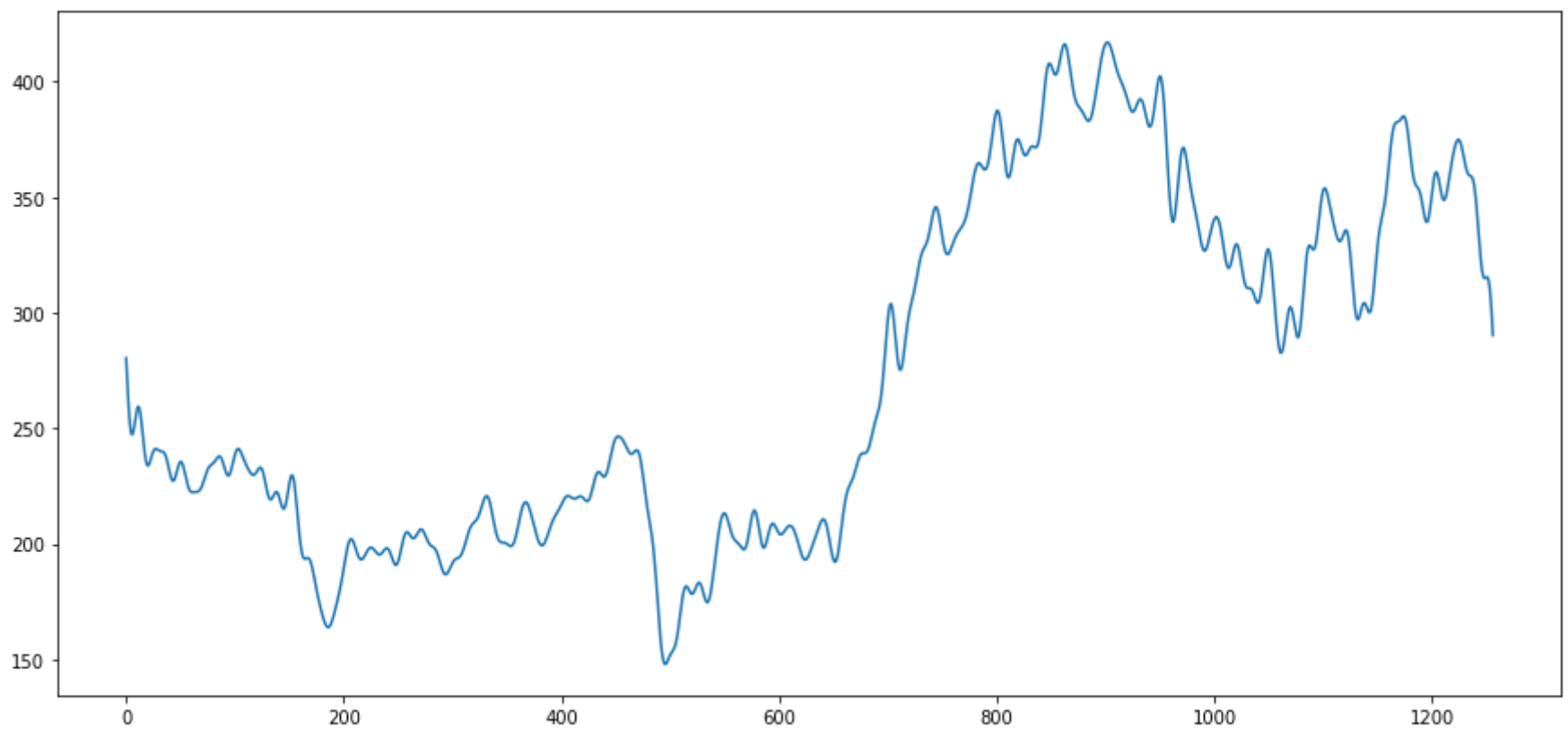
In [5]:
```
from collections import deque
items = deque(np.asarray(fft_df['absolute'].tolist()))
items.rotate(int(np.floor(len(fft_df)/2)))
plt.figure(figsize=(15, 7))
plt.stem(items)
plt.show()
```



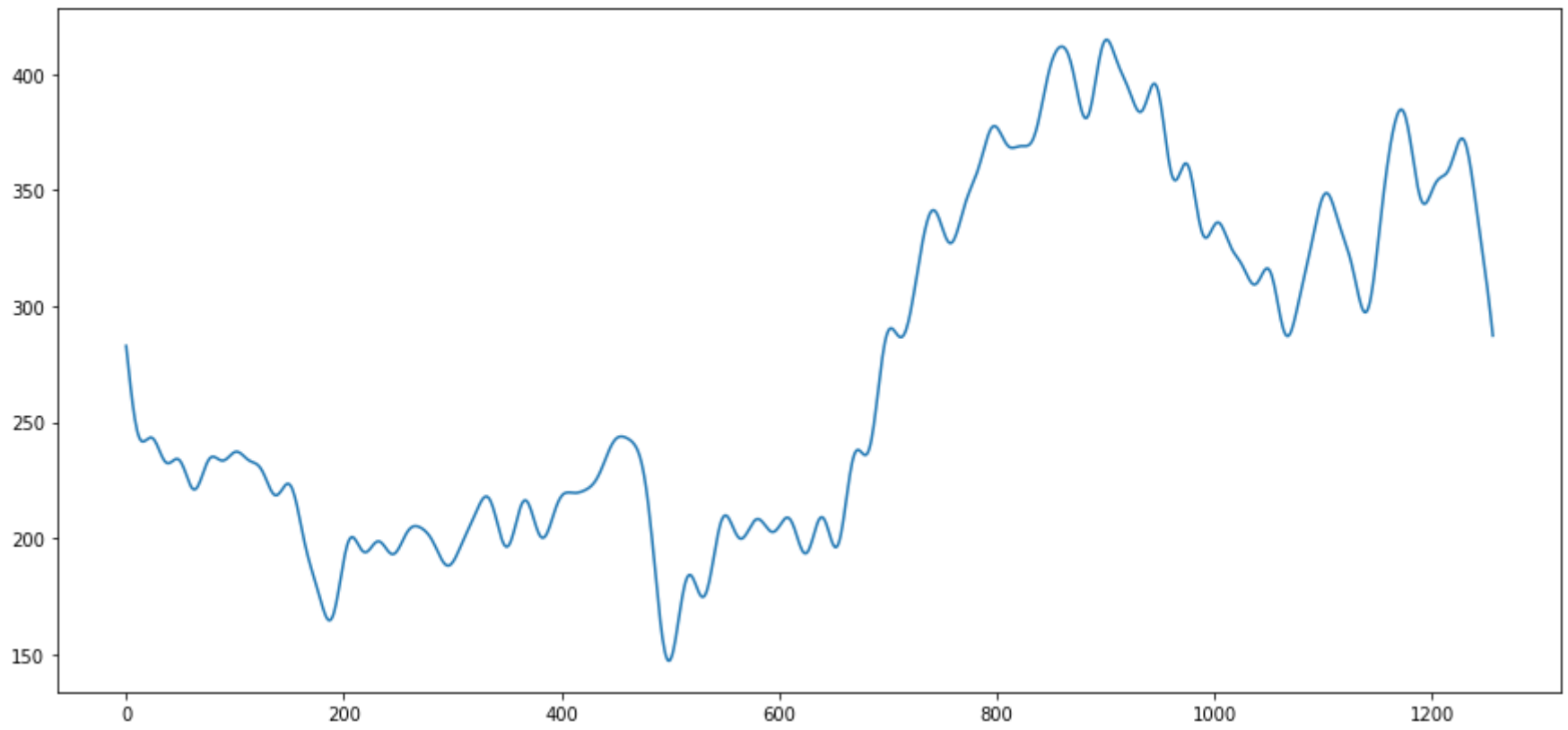
In [6]:
```
# take all components

plt.figure(figsize=(15, 7))
plt.plot(np.fft.ifft(np.asarray(fft_df['fft'].tolist())))
plt.show()
```



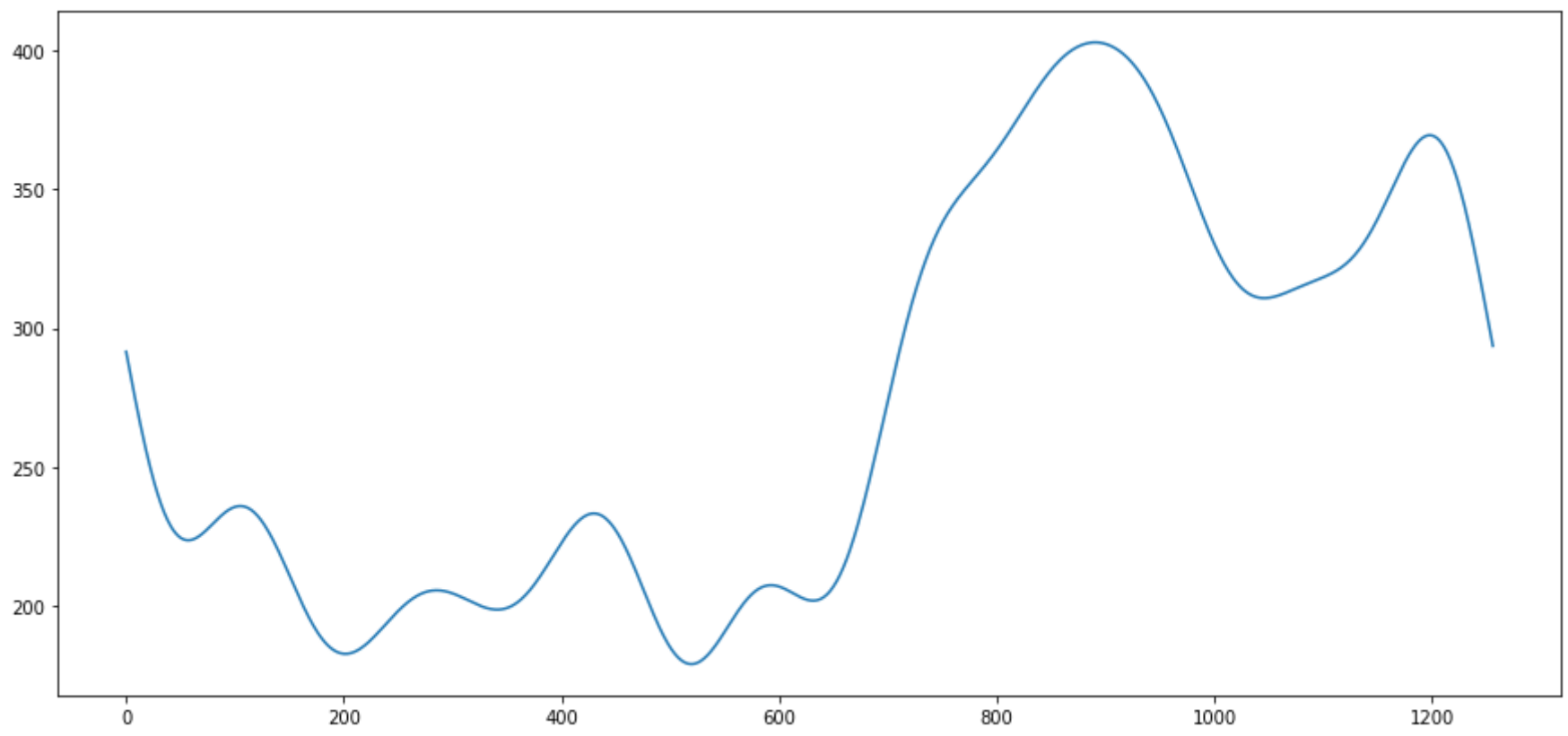
In [7]:
```
# take only 200 components

fft_list = np.asarray(fft_df['fft'].tolist())
fft_list[100:-100] = 0

plt.figure(figsize=(15, 7))
plt.plot(np.fft.ifft(fft_list))
plt.show()
```



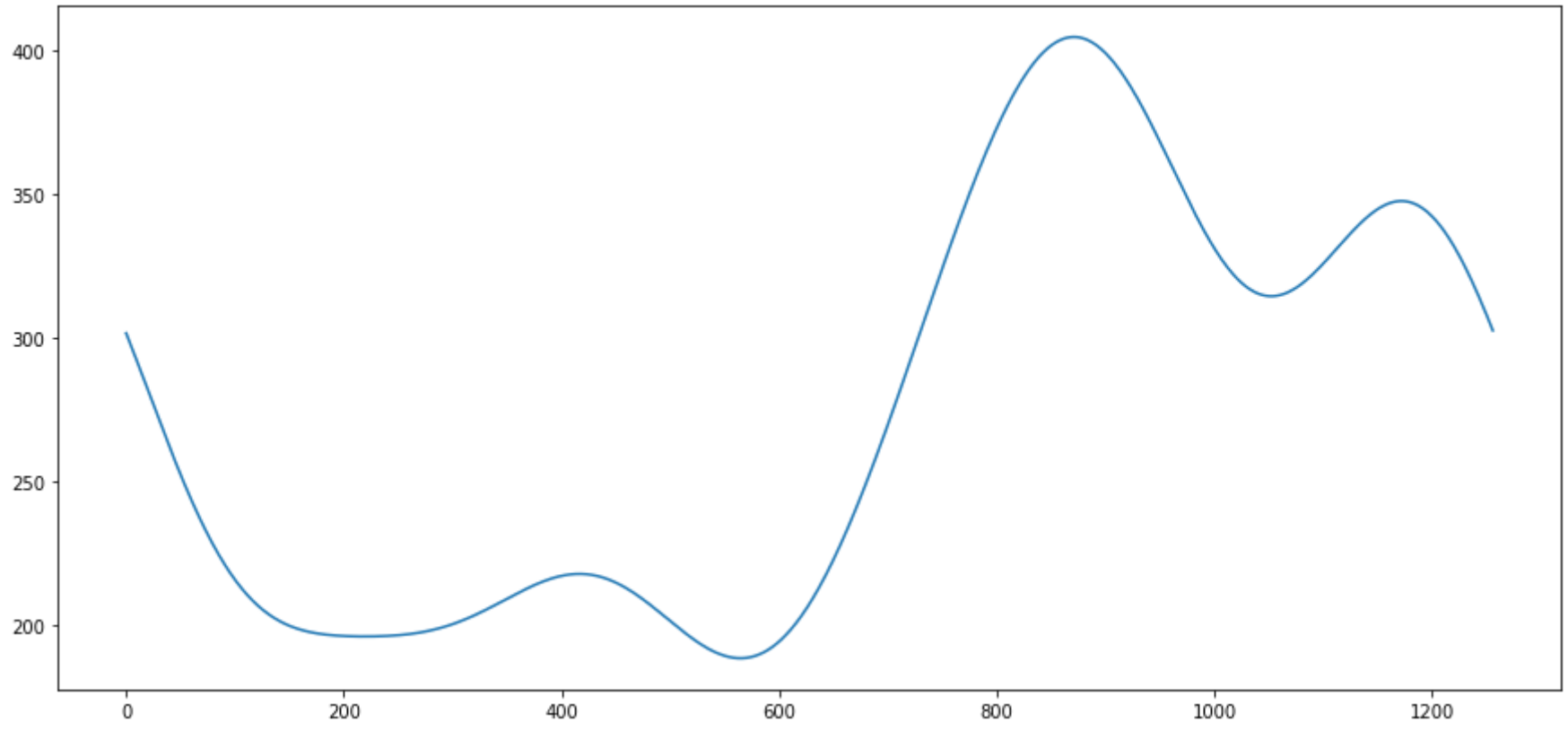
In [8]:
```
# take only 100 components

fft_list = np.asarray(fft_df['fft'].tolist())
fft_list[50:-50] = 0

plt.figure(figsize=(15, 7))
plt.plot(np.fft.ifft(fft_list))
plt.show()
```



In [9]:
```
# take only 20 components

fft_list = np.asarray(fft_df['fft'].tolist())
fft_list[10:-10] = 0

plt.figure(figsize=(15, 7))
plt.plot(np.fft.ifft(fft_list))
plt.show()
```



In [10]:
```
# take only 10 components

fft_list = np.asarray(fft_df['fft'].tolist())
fft_list[5:-5] = 0

plt.figure(figsize=(15, 7))
plt.plot(np.fft.ifft(fft_list))
plt.show()
```



In [ ]: