In [1]:

```
import numpy as np
import pandas as pd

large_eq = pd.read_csv('./HW9/Large_Eq.csv')
large_eq['Lat'] = large_eq['Lat'].round().astype(int)
large_eq['Lon'] = large_eq['Lon'].round().astype(int)
lat = large_eq.iloc[:,5]
lon = large_eq.iloc[:,6]
print(lat)
print(lon)

LL_csv = pd.concat([large_eq['Lat'], large_eq['Lon']], axis=1)
print(LL_csv)

W_lon_csv = []
W_lat_csv = []

E_lon_csv = []
E_lat_csv = []

PM_csv = []
for i, row in large_eq.iterrows():
    lon = row['Lon']
    lat = row['Lat']
    if lon < 0:
        W_lon_csv.append(lon)
        W_lat_csv.append(lat)
    elif lon>0:
        E_lon_csv.append(lon)
        E_lat_csv.append(lat)
    elif lon==0:
        PM_csv.append(lat)

W_LL_csv = pd.DataFrame({'Lat': W_lat_csv, 'Lon': W_lon_csv})
E_LL_csv = pd.DataFrame({'Lat': E_lat_csv, 'Lon': E_lon_csv})
PM_LL_csv = pd.DataFrame({'Lat': PM_csv, 'Lon': 0})

print("West Longitude and Latitude:")
print(W_LL_csv)
print("\nEast Longitude and Latitude:")
print(E_LL_csv)
print("\nPrime Meridian:")
print(PM_LL_csv)
```

```
W_lon_dat = []
W_lat_dat = []

E_lon_dat = []
E_lat_dat = []

W_elevation = []
E_elevation = []
topo = open('./HW9/topo (1).dat','r')
for i in topo:
    data = i.strip().split('\t')
    lon = float(data[0])
    lat = float(data[1])
    ele = float(data[2])
    if lon<180:
        E_lon_dat.append(lon)
        E_lat_dat.append(lat)
        E_elevation.append(ele)
    elif lon>=180:
        W_lon_dat.append(lon)
        W_lat_dat.append(lat)
        W_elevation.append(ele)

W_LL_dat = pd.DataFrame({'Lat': W_lat_dat, 'Lon': W_lon_dat, 'Elevation': W_elevation})
E_LL_dat = pd.DataFrame({'Lat': E_lat_dat, 'Lon': E_lon_dat, 'Elevation': E_elevation})

print("West Longitude and Latitude:")
print(W_LL_dat)
print("\nEast Longitude and Latitude:")
print(E_LL_dat)

W_LL_csv['Lon']=W_LL_csv['Lon']+360
merged_data_W = pd.merge(W_LL_csv, W_LL_dat, on=['Lon', 'Lat'], how='left')
merged_data_E = pd.merge(E_LL_csv, E_LL_dat, on=['Lon', 'Lat'], how='left')
merged_data_PM = pd.merge(PM_LL_csv, E_LL_dat, on=['Lon', 'Lat'], how='left')
print(merged_data_W[['Lon', 'Lat', 'Elevation']])
print(merged_data_E[['Lon', 'Lat', 'Elevation']])
print(merged_data_PM[['Lon','Lat','Elevation']])

total = max_elevation-min_elevation
interval_difference = total/10
print(interval_difference)

import matplotlib.pyplot as plt
```

```
merged_data_TOTAL = pd.concat([merged_data_W, merged_data_E, merged_data_PM], ignore_index=True)
hist, bins = np.histogram(merged_data['Elevation'], bins=10, range=(min_elevation, max_elevation))
plt.hist(merged_data_TOTAL['Elevation'], bins=bins, color='purple',edgecolor='black')
plt.title('Elevation Histogram')
plt.xlabel('Topograph (km)')
plt.ylabel('Number of Earthquakes')
plt.show()
```