```python
#Importing all the required libraries
import pandas as pd
from pandas import DataFrame
import numpy as np

import matplotlib.pyplot as plt
plt.rcParams["figure.figsize"] = (15,7)

import seaborn as sns
from datetime import datetime, timedelta

from statsmodels.tsa.arima_model import ARIMA
from statsmodels.tsa.statespace.sarimax import SARIMAX
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from statsmodels.tsa.stattools import adfuller
from statsmodels.tsa.seasonal import seasonal_decompose

from scipy import stats
import statsmodels.api as sm
from itertools import product

import warnings
warnings.filterwarnings('ignore')
```

```python
#Loading the dataset

dateparse = lambda dates: pd.datetime.strptime(dates, '%Y-%m-%d')

df = pd.read_csv('../input/crypto-markets.csv', parse_dates=['date'],

index_col='date', date_parser=dateparse)

df.head()
```

```
# Extract the bitcoin data only
btc=df[df['symbol']=='BTC']
# Drop some columns
btc.drop(['slug', 'volume','symbol','name','ranknow','market',
'close_ratio', 'spread'],axis=1,inplace=True)
```

Note:- For all next steps you have to consider monthly variation of bitcoin prices.

Q1. (6 marks)

Plot rolling average and rolling standard deviation and see if the time series looks stationary to you?

Q2. (6 marks)

Look at Autocorrelation and see if the same observations are made regarding the stationarity of the series.

Q3. (6 marks)

How will you remove this seasonality and make the series stationary, write the processing steps and see the rolling average and standard deviation if it gets improved?

HInt:- You can use the Box-Cox transformation to suppress some of the seasonality. The Box-Cox transformation is a family of power transformations indexed by a parameter lambda. Whenever you use it the parameter needs to be estimated from the data.

Refer to:- https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.boxcox.html

Q4. (6 marks)

AutoRegressive Integrated Moving Average. ARIMA models are denoted with the notation ARIMA(p, d, q). These parameters account for seasonality, trend, and noise in datasets:

p - the number of lag observations to include in the model, or lag order. (AR)

d - the number of times that the raw observations are different, or the degree of differencing. (I)

q - the size of the moving average window, also called the order of moving average.(MA)

Try to fit an ARIMA model with given 4 sets of hyperparameters, and print their summary. Out of these 4 models select the best one.

(1,1,0)

(2,1,0)

(0,1,2)

(1,1,1)

Q.5. (6 marks)

Try to plot the prediction prices by the ARIMA model selected in the previous question, along with the true prices and explain how your model performs, and try to list down reasons why your model might be not performing well.

Hint:-

In order to do predictions you need to reverse transform the data

Use this in invboxcox function in order to transform the data back which was transformed in question 3.

```python
# Inverse Box-Cox Transformation Function
def invboxcox(y,lmbda):
   if lmbda == 0:
      return(np.exp(y))
   else:
      return(np.exp(np.log(lmbda*y+1)/lmbda))
```