```r
# THIS SCRIPT IS USED FOR THE STATISTICAL COMPUTING OF THE FORECAST FOR THE SETPOINTS
# BASED UPON THE HISTORICAL DATA

args <- commandArgs()
headFilePath <- args[2]
tailFilePath <- args[3]
outputFilePath <- args[4]

library("TTR")
library("forecast")

## The script will need the L2 Sp for the Head and Tail, as well as
## the corresponding value of the grade.

# SCANNING THE PREVIOUS VALUES OF HEAD AND TAIL SETPOINTS
gd_h <- scan(headFilePath, skip=1);
gd_t <- scan(tailFilePath, skip=1);


# FORMULATION OF THE TIME SERIES
gd_hts <- ts(gd_h);
gd_tts <- ts(gd_t);

#PLOTTING THE TIME SERIES CHARTS AND SAVING IT
png(file = "C:/Users/Dewal Agarwal/Desktop/Intern Project/CropShearSetPoint/Resources/Dynamic
Plots/gd_hts_plot.jpg");
plot(gd_hts, type="o", col= "black", main = "Time Series Head");
dev.off();
png(file = "C:/Users/Dewal Agarwal/Desktop/Intern Project/CropShearSetPoint/Resources/Dynamic
Plots/gd_tts_plot.jpg");
plot(gd_tts, type="o", col= "black", main = "Time Series Tail");
dev.off();

# #[OP] CALCULATION OF MOVING AVERAGES
# gd_htsSMA5 <- SMA(gd_hts, n=5);
# gd_ttsSMA5 <- SMA(gd_tts, n=5);

# Smoothing using Holt-Winters function (SES & DES)

# HEAD & TAIL SES smoothing
# gd_hSES <- HoltWinters(gd_hts, beta =  FALSE, gamma = FALSE)
```

```
# gd_tSES <- HoltWinters(gd_tts, beta = FALSE, gamma = FALSE)

# HEAD & TAIL DES smoothing
gd_hDES <- HoltWinters(gd_hts, gamma = FALSE, l.start = gd_h[1], b.start = gd_h[2]-gd_h[1]);
gd_tDES <- HoltWinters(gd_tts, gamma = FALSE, l.start = gd_t[1], b.start = gd_t[2]-gd_t[1]);

# [op] Printing the value of the smooth parameters.
# print(gd_hDES);
# print(gd_tDES);

# PLOTTING INPUT V/S THE FITTED VALUES
# This part is working fine
png(file = "C:/Users/Dewal Agarwal/Desktop/Intern Project/CropShearSetPoint/Resources/Dynamic
Plots/gd_hDES_plot.jpg");
plot(gd_hDES, type="o", main = "Observed & Fitted Head");
dev.off();
png(file = "C:/Users/Dewal Agarwal/Desktop/Intern Project/CropShearSetPoint/Resources/Dynamic
Plots/gd_tDES_plot.jpg");
plot(gd_tDES, type="o", main = "Observed & Fitted Tail");
dev.off();

# FORECASTING THE NEXT 'h' VALUES FOR THE HEAD AND THE TAIL
gd_hDESfc = forecast.HoltWinters(gd_hDES, h=1);
gd_tDESfc = forecast.HoltWinters(gd_tDES, h=1);

# [op] Print the forecast for the head and tail
# print("The Head setpoint is");
# print(gd_hDESfc)
# print("The Tail setpoint is");
# print(gd_tDESfc)


forecastH <- as.numeric(gd_tDESfc$mean)
forecastT <- as.numeric(gd_hDESfc$mean)

# OUTPUTTING THE FORECASTED VALUE IN THE TEXT FILE
sink(outputFilePath)
print(forecastH)
print(forecastT)
sink()
```

```
## The output is read from the text file by the C# code
# print("The head setpoint value is:")
# print(forecastH)
# print("The tail setpoint value is:")
# print(forecastT)

## Functions for adding the bias.
#Final_SPH <- function(L2Hinput)
#{ sp = (forecastH + L2Hinput)/2
# There should be a provision for changing the entry from the exixting data base.
#}
#print("Displaying the final setpoint with offset");
#print(Final_SPH(500))

##Final_SPT <- function(L2Tinput)
#{ sp = (forecastT + L2Tinput)/2
# There should be a provision for changing the entry from the exixting data base.
#}
#print("Displaying the final setpoint with offset");
#print(Final_SPT(500))
```