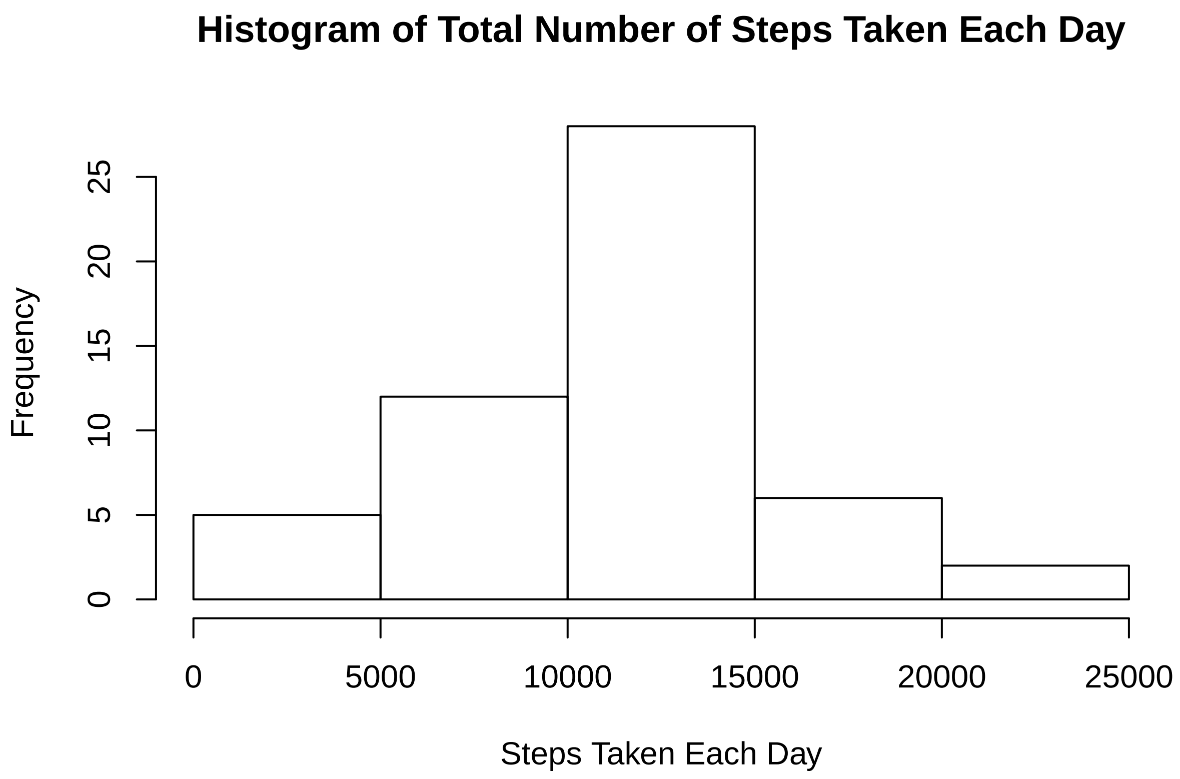
Course Project 1

1. The code for reading and processing our data is:

```
data<-read.csv("project 1 data.csv")
```

2. Generate a histogram of the total number of steps taken each day:

```
hist(tapply(data$steps, data$date, sum), xlab="Steps Taken Each Day", main="Histogram of Total Number o
```



3. The mean number of steps taken is:

```
mean(tapply(data$steps, data$date, sum), na.rm=TRUE)
```

```
## [1] 10766.19
```

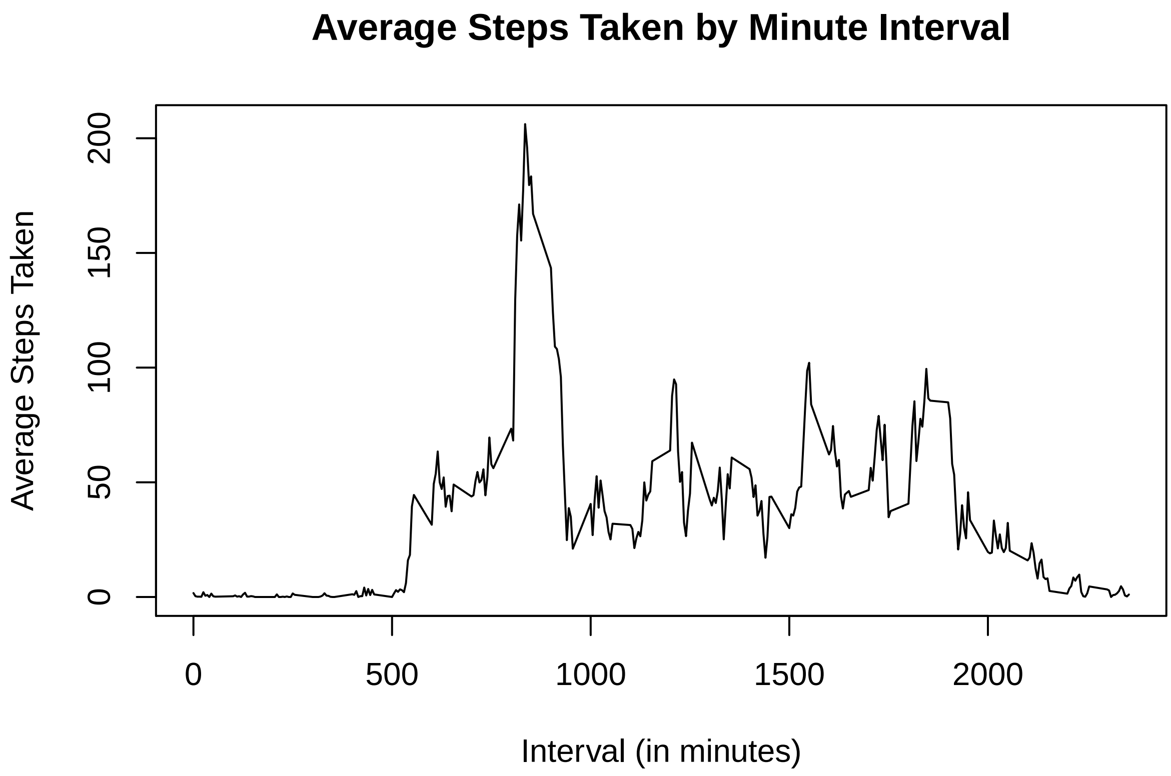
and the median number of steps taken is:

```
median(tapply(data$steps, data$date, sum), na.rm=TRUE)
```

```
## [1] 10765
```

4. Here is a time-series plot of the average number of steps:

```
plot(unique(data$interval), tapply(data$steps, data$interval, mean, na.rm=TRUE), type="l", xlab="Interva
```



5. The five-minute interval that, on average, contains the maximum number of steps is:

```
##find which interval has the maximum average steps
max<-which.max(tapply(data$steps, data$interval, mean, na.rm=TRUE))
##extract the name of the interval which corresponds to a length in feet
names(max)
```

```
## [1] "835"
```

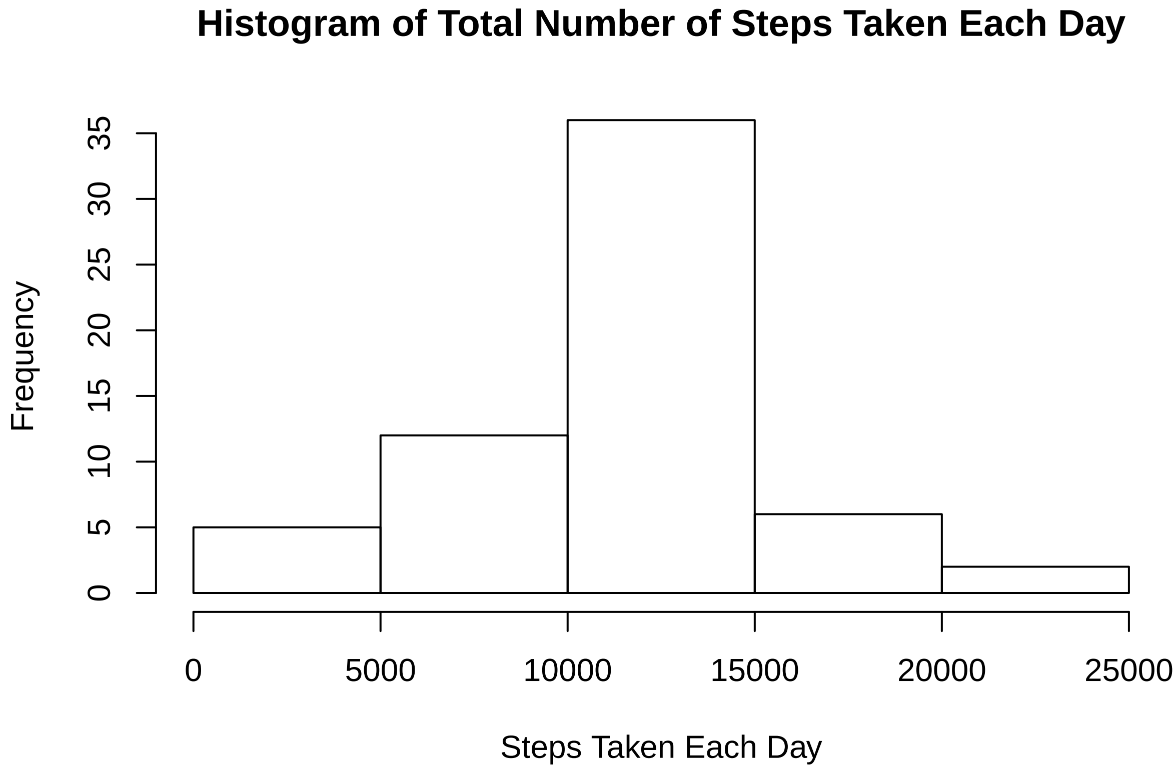
6. The code to impute values for the NAs in the data is:

```
##get average steps for each interval
avg_steps<-tapply(data$steps, data$interval, mean, na.rm=TRUE)
##make data frame of invtervals and average steps
df1<-data.frame(interval=names(avg_steps), avg_steps)

##load qdap package for lookup function
library(qdapTools)
## replaces NAs with average steps from corresponding interval in df1. Must convert interval column in
data$steps<-ifelse(is.na(data$steps), lookup(as.factor(data$interval), df1), data$steps)
```

7. Using this newly refined data, here is a histogram of the total number of steps taken each day:

```
hist(tapply(data$steps, data$date, sum), xlab="Steps Taken Each Day", main="Histogram of Total Number o
```



8. And here is a panel plot comparing the average number of steps taken across intervals for both weekdays and weekends:

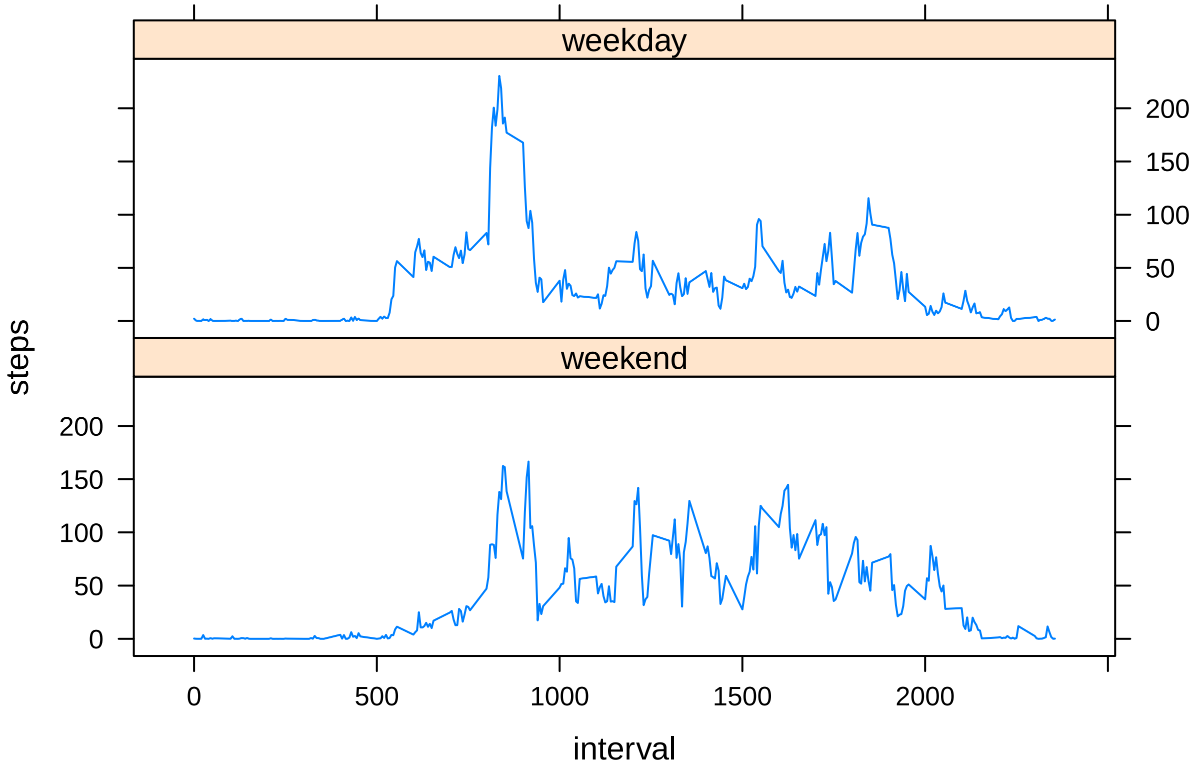
```
##load lubridate package and convert date column from factor to date
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

```
##convert to date format
data$date<-ymd(data$date)
##create a vector of days of the week for each date in data
day<-weekdays(data$date)
##overwrite day vector based on whether day is week or weekend
day<-ifelse((day=="Sunday" | day=="Saturday"), "weekend", "weekday")
##insert day vector into data
data<-data.frame(steps=data[,1], date=data[ ,2], day, interval=data[ ,3])

##create a subset data frame where day=weekend
sub_end<-subset(data, data$day=="weekend")
##create a new data frame of average steps per interval, interval, and day from weekend days
end<-data.frame(steps=tapply(sub_end$steps, sub_end$interval, mean), interval=unique(sub_end$interval),
##create a subset data frame where day=weekday
sub_day<-subset(data, data$day=="weekday")
##create a new data frame of average steps per interval, interval, and day from weekdays
day<-data.frame(steps=tapply(sub_day$steps, sub_day$interval, mean), interval=unique(sub_day$interval),
##put df of avg. steps per interval for weekend and df for avg. steps per interval for weekday together
new_data<-rbind(end, day)
##make lattice panel plot of interval on x-axis, avg. steps taken on y-axis, and separated by weekend/we
library(lattice)
xyplot(steps~interval|day, data=new_data, type="l", layout=c(1,2))
```


weekday
weekend
steps
interval
0
50
100
150
200
0
500
1000
1500
2000