Shrija Chavan
A20381511
ITMD 521
Week 02

Analysis of Part 01:
Below is a brief explanation of the amount of time required to run each dataset using the awk script with a graph and chart combined with some screen shots to explain the process in details a compare the time taken by each dataset.

System Configuration:
Virtual Box Used: Ubuntu/trusty64.
Memory allocated: 2048 MB.
CPU Speed: 2699.986 MHz

Observation:

1. Dataset of year 1990 took the least amount of time as compared to the other dataset since the amount of data increased with each dataset. Hence, the time.

2. Processing time of the datasets with memory of 1028 and 2048 was same, concluding that the processing time is not affected by the increase or decrease of memory.

3. The processing time increases while processing huge amount of data.

4. For quicker results and to reduce the processing time, the job can be executed on different processors.

5. The amount Memory free before running the job was 86232 KB, and the Memory free while running the job was 66252 KB (Screen shots attached below is the file)

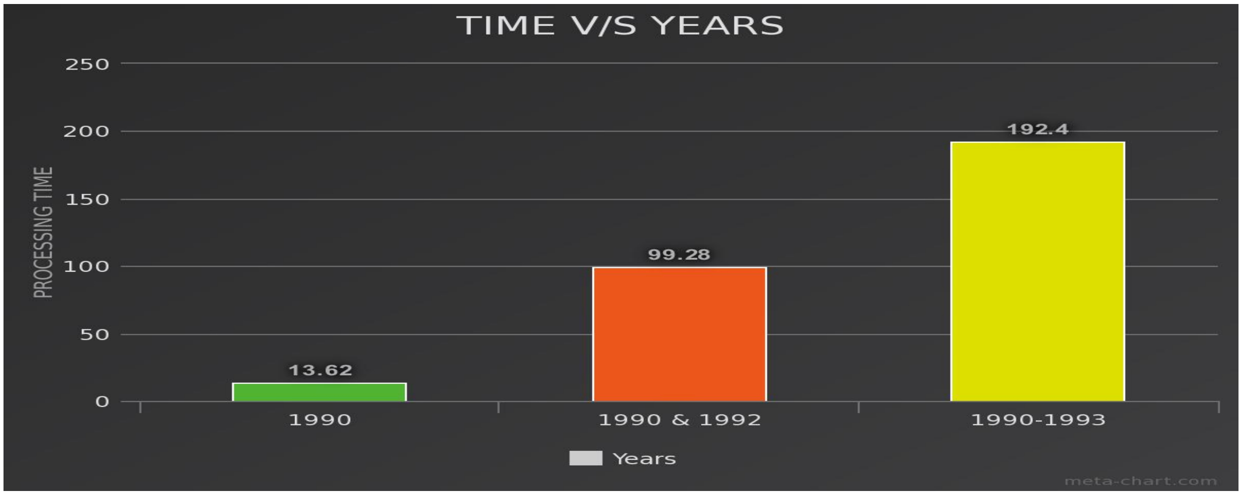
The below Chart and graph explains about the time taken to run the awk script on each dataset while parsing the huge amount of data in seconds and giving us the max temperature for each year.

Chart:

Year	Time taken to run the job	Memory Allocated	CPU Speed
1990	13.62	2048 MB	2699.86
1990 &1992	99.28	2048 MB	2699.86
1990 -1993	192.40	2048 MB	2699.86

Shrija Chavan
A20381511
ITMD 521
Week 02

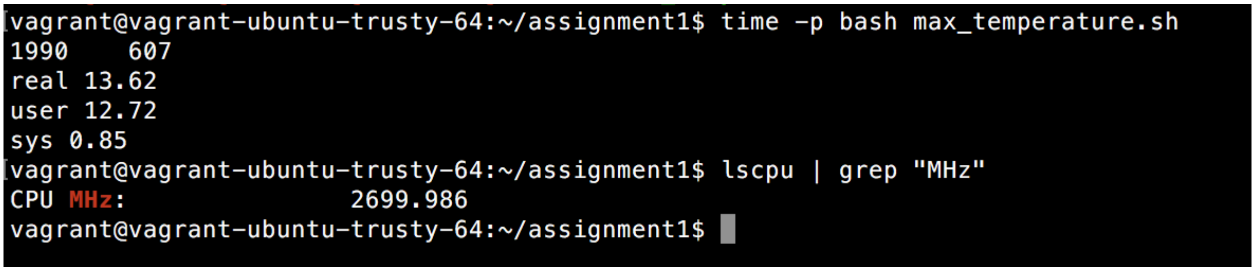
Graph: Using awk script:



We can see the how quick the awk sciprt parses the entire data and the trend where the time is increased with larger dataset in the above graph while using the awk script.

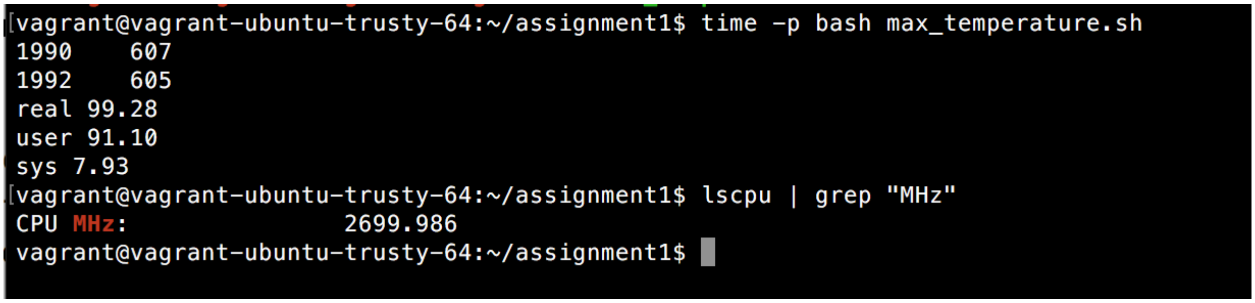
Screen Shots:

Result of the dataset 1990 while running the Awk script:



```
[vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ time -p bash max_temperature.sh
1990    607
real 13.62
user 12.72
sys 0.85
[vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ lscpu | grep "MHz"
CPU MHz:                 2699.986
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ 
```

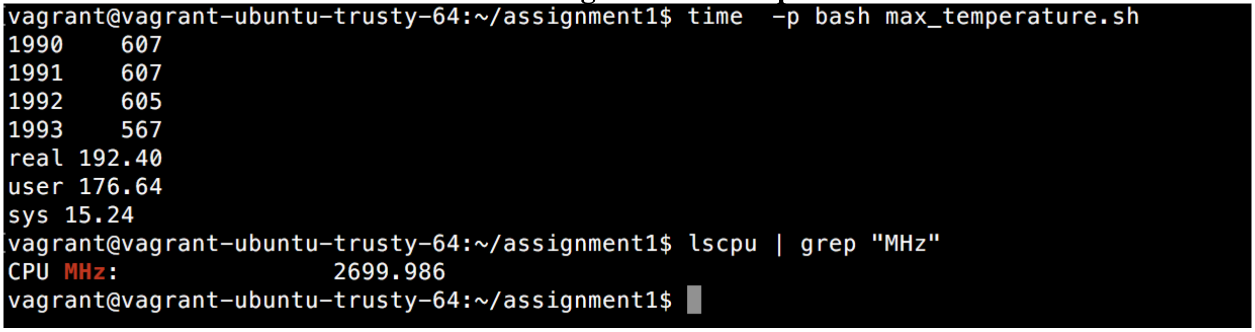
Result of the dataset 1990 and 1992 while running the Awk script:



```
[vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ time -p bash max_temperature.sh
1990    607
1992    605
real 99.28
user 91.10
sys 7.93
[vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ lscpu | grep "MHz"
CPU MHz:                 2699.986
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ 
```

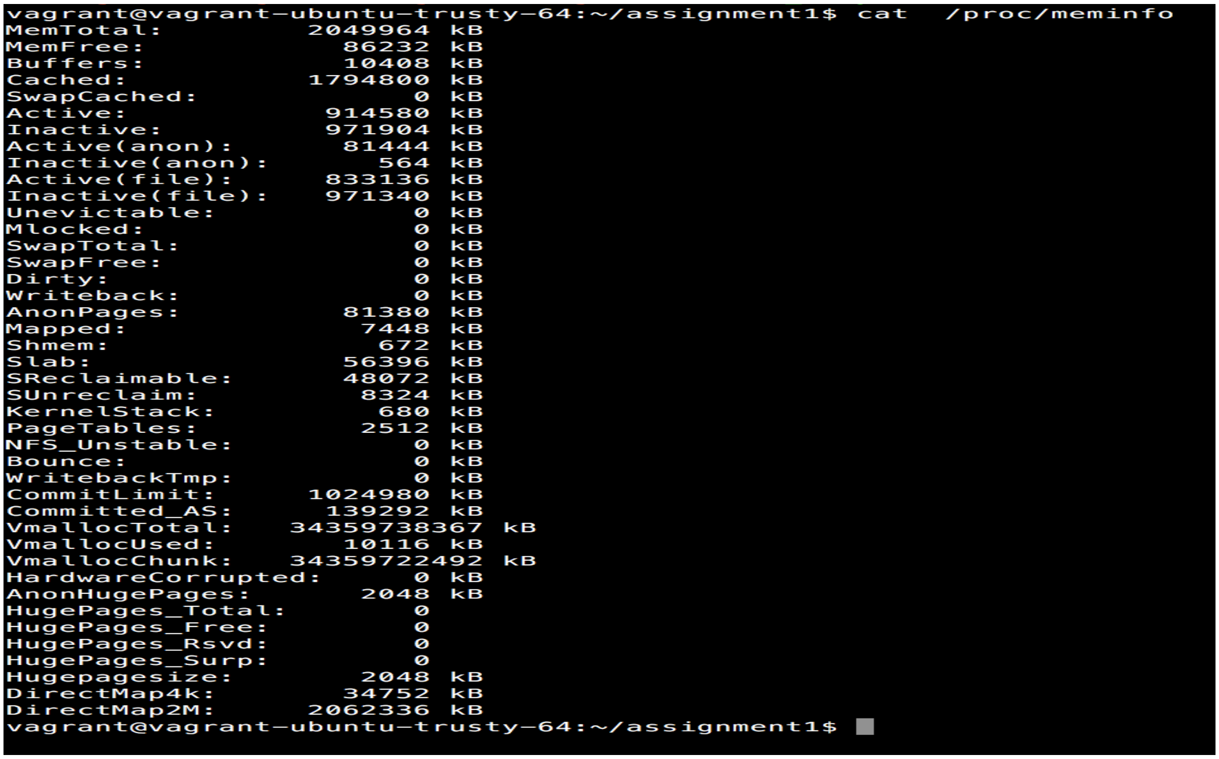
Shrija Chavan
A20381511
ITMD 521
Week 02

Result of the dataset 1990-93 while running the Awk script:



```
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ time  -p bash max_temperature.sh
1990    607
1991    607
1992    605
1993    567
real 192.40
user 176.64
sys 15.24
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ lscpu | grep "MHz"
CPU MHz:                2699.986
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$
```

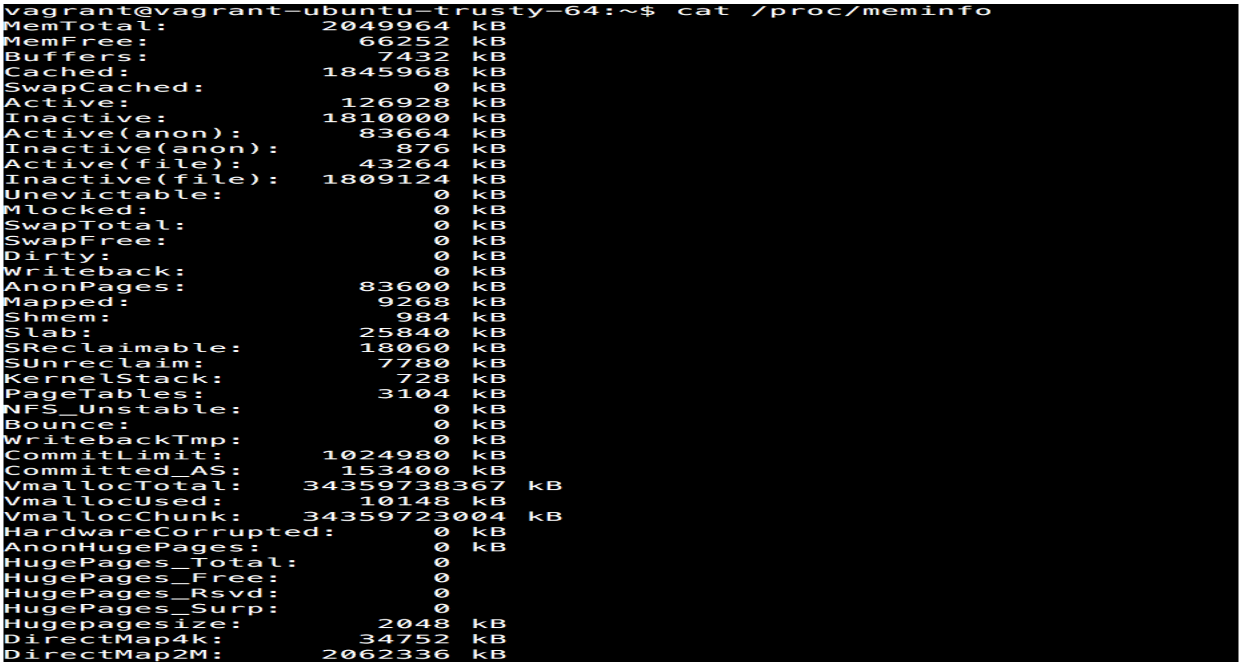
Memory Allocated before running the job:



```
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$ cat  /proc/meminfo
MemTotal:        2049964 kB
MemFree:           86232 kB
Buffers:           10408 kB
Cached:          1794800 kB
SwapCached:            0 kB
Active:           914580 kB
Inactive:         971904 kB
Active(anon):      81444 kB
Inactive(anon):      564 kB
Active(file):     833136 kB
Inactive(file):   971340 kB
Unevictable:           0 kB
Mlocked:               0 kB
SwapTotal:             0 kB
SwapFree:              0 kB
Dirty:                 0 kB
Writeback:             0 kB
AnonPages:         81380 kB
Mapped:             7448 kB
Shmem:               672 kB
Slab:              56396 kB
SReclaimable:      48072 kB
SUnreclaim:         8324 kB
KernelStack:         680 kB
PageTables:         2512 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:     1024980 kB
Committed_AS:     139292 kB
VmallocTotal:   34359738367 kB
VmallocUsed:       10116 kB
VmallocChunk:   34359722492 kB
HardwareCorrupted:     0 kB
AnonHugePages:      2048 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB
DirectMap4k:       34752 kB
DirectMap2M:     2062336 kB
vagrant@vagrant-ubuntu-trusty-64:~/assignment1$
```

Shrija Chavan
A20381511
ITMD 521
Week 02

Memory Allocated while running the job:



```
vagrant@vagrant-ubuntu-trusty-64:~$ cat /proc/meminfo
MemTotal:        2049964 kB
MemFree:           66252 kB
Buffers:            7432 kB
Cached:          1845968 kB
SwapCached:            0 kB
Active:           126928 kB
Inactive:        1810000 kB
Active(anon):      83664 kB
Inactive(anon):      876 kB
Active(file):      43264 kB
Inactive(file):  1809124 kB
Unevictable:           0 kB
Mlocked:               0 kB
SwapTotal:             0 kB
SwapFree:              0 kB
Dirty:                 0 kB
Writeback:             0 kB
AnonPages:         83600 kB
Mapped:             9268 kB
Shmem:               984 kB
Slab:              25840 kB
SReclaimable:      18060 kB
SUnreclaim:         7780 kB
KernelStack:         728 kB
PageTables:         3104 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:     1024980 kB
Committed_AS:     153400 kB
VmallocTotal:   34359738367 kB
VmallocUsed:       10148 kB
VmallocChunk:   34359723004 kB
HardwareCorrupted:     0 kB
AnonHugePages:         0 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB
DirectMap4k:       34752 kB
DirectMap2M:     2062336 kB
```

Analysis of Part 02:

In part02, I wrote a java class which will parse each record from the dataset, write it into the CSV file and then load the CSV file into the database executing a SQL query which will return the highest temperature for each year in the database.

Observation:

1. Java class was written to parse each dataset and create a csv file. The csv file was then loaded into the DB and then execute a sql query to give me the highest temperature per year. But, the time taken to carry out the entire process took a lot of time as compared to running the awk sciprt on the dataset.

2. But, by improving the code and making it more efficient, I was able to carry out the same operation in less than 50 secs.

3. However, the awk script still seems to be much more efficient to retrieve the data as compared to the Java program.

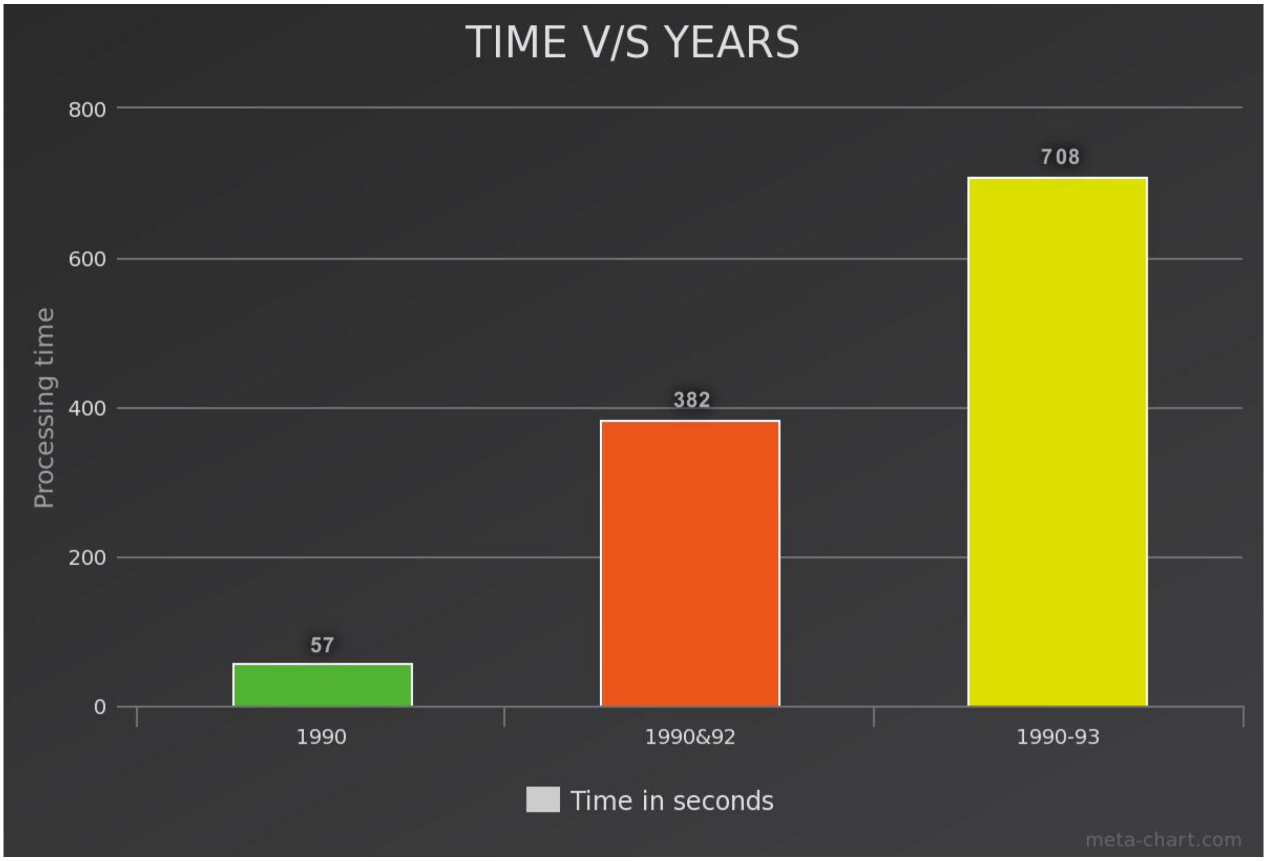
Shrija Chavan
A20381511
ITMD 521
Week 02

Chart:

Year	Time taken to run the job in secs	Memory Allocated	CPU Speed
1990	57	2048 MB	2698.86
1990 &1992	382	2048 MB	2698.86
1990 -1993	708	2048 MB	2698.86

Graph: Using java:

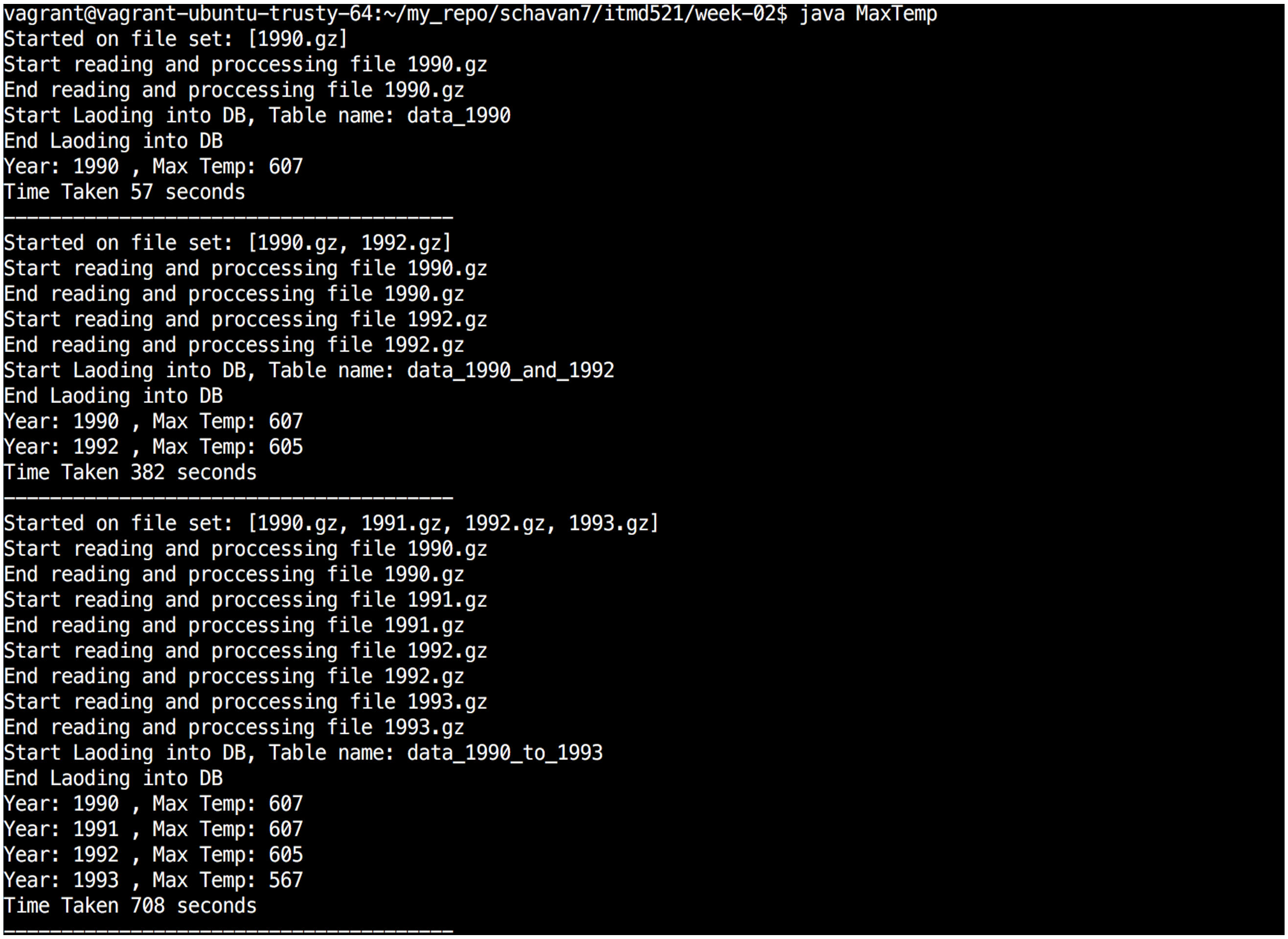
We can see the how the java code parses the data and the trend where the time is increased with larger dataset in the above graph while using the java code.



Shrija Chavan
A20381511
ITMD 521
Week 02

Below is the screen shots of Part02:

Output in the console showing the highest temperature per year and time taken to execute each dataset.



```
vagrant@vagrant-ubuntu-trusty-64:~/my_repo/schavan7/itmd521/week-02$ java MaxTemp
Started on file set: [1990.gz]
Start reading and proccessing file 1990.gz
End reading and proccessing file 1990.gz
Start Laoding into DB, Table name: data_1990
End Laoding into DB
Year: 1990 , Max Temp: 607
Time Taken 57 seconds
---------------------------------------
Started on file set: [1990.gz, 1992.gz]
Start reading and proccessing file 1990.gz
End reading and proccessing file 1990.gz
Start reading and proccessing file 1992.gz
End reading and proccessing file 1992.gz
Start Laoding into DB, Table name: data_1990_and_1992
End Laoding into DB
Year: 1990 , Max Temp: 607
Year: 1992 , Max Temp: 605
Time Taken 382 seconds
---------------------------------------
Started on file set: [1990.gz, 1991.gz, 1992.gz, 1993.gz]
Start reading and proccessing file 1990.gz
End reading and proccessing file 1990.gz
Start reading and proccessing file 1991.gz
End reading and proccessing file 1991.gz
Start reading and proccessing file 1992.gz
End reading and proccessing file 1992.gz
Start reading and proccessing file 1993.gz
End reading and proccessing file 1993.gz
Start Laoding into DB, Table name: data_1990_to_1993
End Laoding into DB
Year: 1990 , Max Temp: 607
Year: 1991 , Max Temp: 607
Year: 1992 , Max Temp: 605
Year: 1993 , Max Temp: 567
Time Taken 708 seconds
---------------------------------------
```