Working with Real and Big Data
BASH SCRIPT #2

INTRODUCTION :

This BASH script is to create a file, testdata.dat, which contains climate data for a specific site over a number of years. My script, called testgen.sh, runs as follows:

$ ./testgen.sh 218 1948 1997

where 218 represents the site location, 1948 the start year, and 1997 the final year.

DESCRIPTION:

For writing a Bash Script, we always start with **#!/bin/bash.**

Defining the text files to a particular variable:

 # temporary file definitions

BTstns="BTemperature_Stations.txt"

alldatafile="BIGDATA8zx2756.txt"

smalldatafile="distilled_datazx47432_$1.dat"

locationsfile="locationszx646332.txt"

tempfile="tempfile.txt”

newfile1=**"testdata.dat"**

First,in this script, we have to print the range of output that user is asking to print in the testdata.dat (example : 1948 to 1997).

For this reason I am using a For loop to display data that the user is asking.

```
for((a=$2;a<=$3;a++))
```

Here $1(first input) is location , $2(second input) is the starting year and $3(3rd input) is the ending year

Then same as in BASH #1 ,similarly, I extracted station IDs from BTemperature_Stations.txt.The actual data starts from the line 5 and ends at line 343.

So using For Loop, I scanned lines 5 to 343 to get the data from that file.For loop is shown below:

```
for x in {5..343}
do
 next=$(head -n $x $BTstns | tail -n 1) # read line x from
BTemperature_Stations.txt
 line=($next)
 stationNUM=(${line[0]}) # station number
stationID=(${line[1]}) # station ID
 stationNAME=(${line[2]}) # station name
nextfile=mm$stationID.txt
       newfile=$stationNAME
       echo "$newfile" >>$tempfile
```

```
 echo "$nextfile" >> $locationsfile # write the data file name first
 y=$(cat $nextfile | tr "," "\n") # remove commas, replace with
newlines
 for z in $y # go through each token in the file $x
 do
 echo $z >> $alldatafile

 done
done
```

Execution of For loop:

For x in 5,

`next=$(head -n $x $BTstns | tail -n 1)`

next= 5th line(data) from the BTemperature_Stations.txt file.

i.e.

next=1 1100120 AGASSIZ BC 1893 1 2012 12 49.25 -121.77 15 N

Now,

`line=($next)` (line is an array that takes the value inside the next variable)

`stationNUM=(${line[0]})` (stationNUM=1)

`stationID=(${line[1]}` (stationID=1100120)

`stationNAME=(${line[2]})` (stationNAME=AGASSIZ)

`nextfile=mm$stationID.txt` (nextfile=mm.1100120.txt)

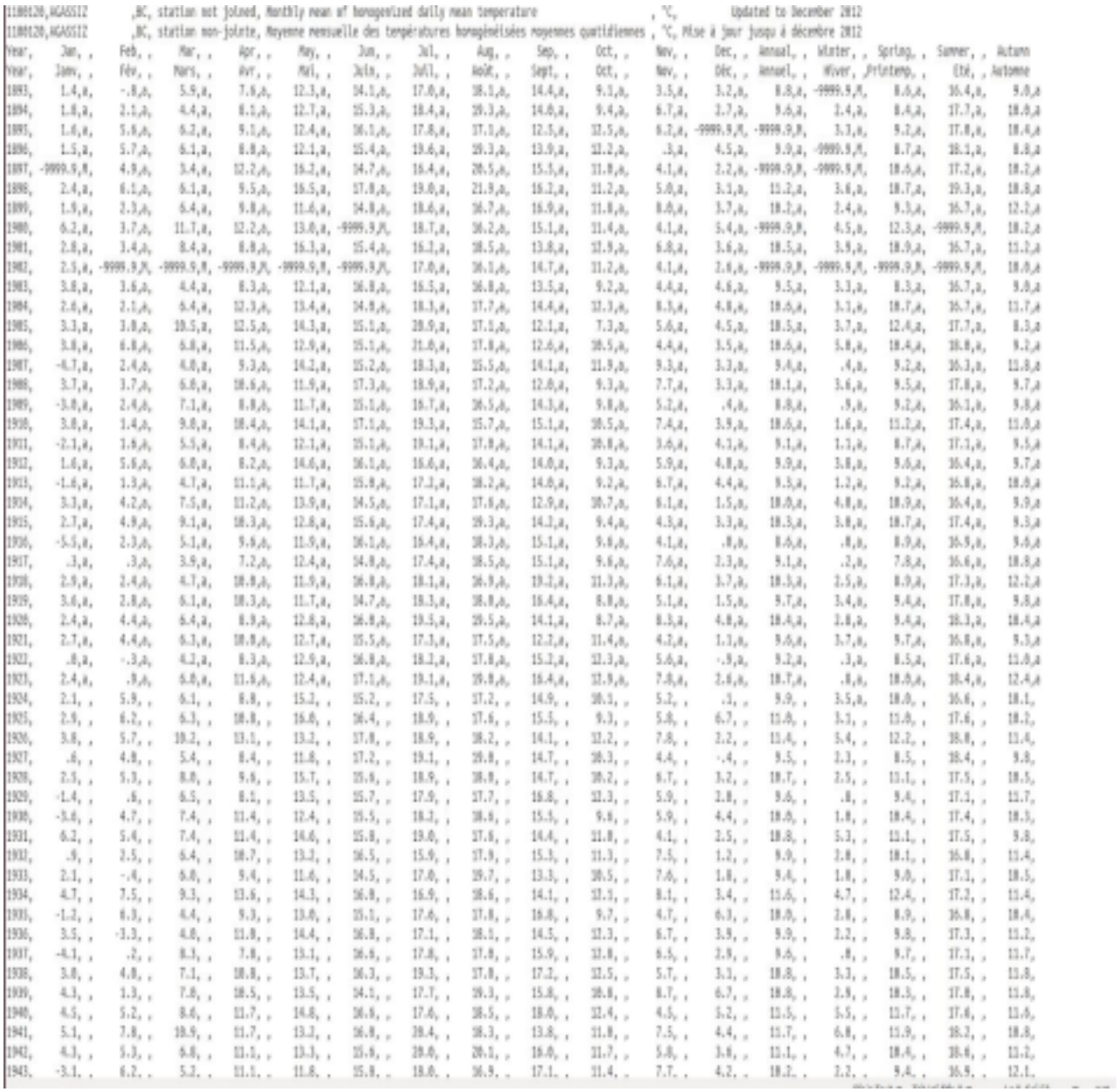
`newfile=$stationNAME` (newfile=AGASSIZ)

`echo "$newfile" >>$tempfile` (Transfering station names to “tempfile.txt”)

```
echo "$nextfile" >> $locationsfile (In first loop it will print
mm.1100120.txt to the locationszx646332.txt)
```

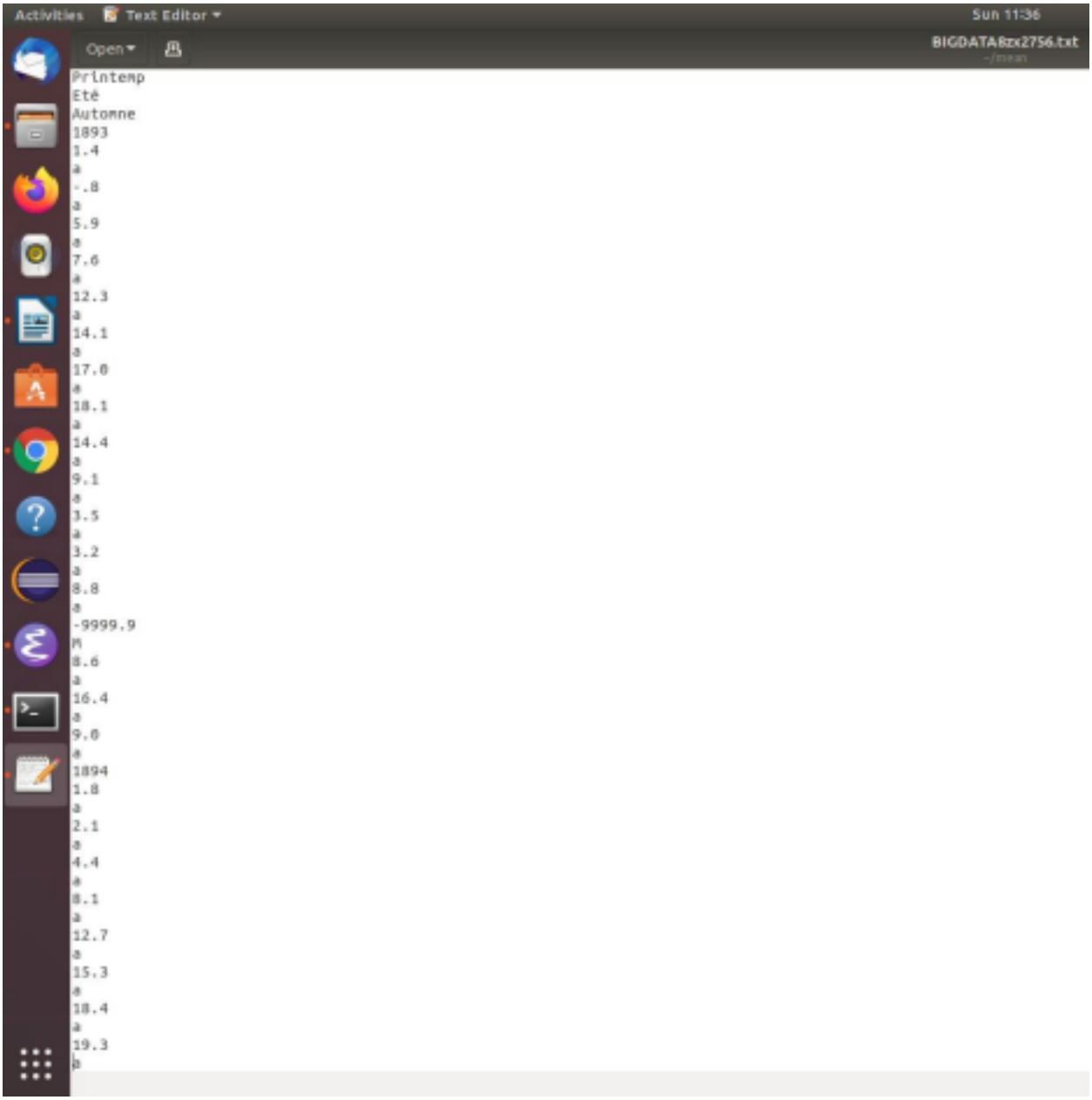
```
 y=$(cat $nextfile | tr "," "\n")
 for z in $y
 do
 echo $z >> $alldatafile
 done
done
```

As you can see in the screenshot below, the mm1100120.txt file consists of the data separated by the commas. In order to remove that we use **tr command** to replace commas by newlines.And after scanning all the data, that data is moved to “BIGDATA8zx2756.txt”.



1180120,AGASSIZ	,BC, station not joined, Monthly mean of homogenized daily mean temperature										, °C,	Updated to December 2012				
1180120,AGASSIZ	,BC, station non-jointe, Moyenne mensuelle des températures homogénéisées moyennes quotidiennes										, °C,	Mise à jour jusqu à décembre 2012				
Year,	Jan, ,	Feb, ,	Mar, ,	Apr, ,	May, ,	Jun, ,	Jul, ,	Aug, ,	Sep, ,	Oct, ,	Nov, ,	Dec, ,	Annual, ,	Winter, ,	Spring, ,	Summer, ,	Autumn
Year,	Janv, ,	Fév, ,	Mars, ,	Avr, ,	Mai, ,	Juin, ,	Juil, ,	Août, ,	Sept, ,	Oct, ,	Nov, ,	Déc, ,	Annuel, ,	Hiver, ,	Printemp, ,	Été, ,	Automne
1893,	1.4,a,	-.8,a,	5.9,a,	7.6,a,	12.3,a,	14.1,a,	17.0,a,	16.1,a,	14.4,a,	9.1,a,	3.5,a,	3.2,a,	8.8,a,	-9999.9,M,	8.6,a,	16.4,a,	9.8,a
1894,	1.8,a,	2.1,a,	4.4,a,	8.1,a,	12.7,a,	15.3,a,	18.4,a,	19.3,a,	14.0,a,	9.4,a,	6.7,a,	2.7,a,	9.6,a,	2.4,a,	8.4,a,	17.7,a,	10.0,a
1895,	1.6,a,	5.6,a,	6.2,a,	9.1,a,	12.4,a,	16.1,a,	17.8,a,	17.1,a,	12.5,a,	12.5,a,	6.2,a,	-9999.9,M,	-9999.9,M,	3.1,a,	9.2,a,	17.8,a,	10.4,a
1896,	1.5,a,	5.7,a,	6.1,a,	8.8,a,	12.1,a,	15.4,a,	19.6,a,	19.3,a,	13.9,a,	12.2,a,	.3,a,	4.5,a,	9.9,a,	-9999.9,M,	8.7,a,	18.1,a,	8.8,a
1897,	-9999.9,M,	4.9,a,	5.4,a,	12.2,a,	16.2,a,	14.7,a,	16.4,a,	20.5,a,	15.5,a,	11.8,a,	4.1,a,	2.2,a,	-9999.9,M,	-9999.9,M,	18.6,a,	17.2,a,	10.2,a
1898,	2.4,a,	6.1,a,	6.1,a,	9.5,a,	16.5,a,	17.8,a,	19.0,a,	21.9,a,	16.2,a,	11.2,a,	5.0,a,	3.1,a,	11.2,a,	3.6,a,	10.7,a,	19.3,a,	10.8,a
1899,	1.9,a,	2.3,a,	6.4,a,	9.8,a,	11.6,a,	14.8,a,	18.6,a,	16.7,a,	16.9,a,	11.8,a,	8.0,a,	3.7,a,	10.2,a,	2.4,a,	9.3,a,	16.7,a,	12.1,a
1900,	6.2,a,	3.7,a,	11.7,a,	12.2,a,	13.0,a,	-9999.9,M,	18.7,a,	16.2,a,	15.1,a,	11.4,a,	4.1,a,	5.4,a,	-9999.9,M,	4.5,a,	12.3,a,	-9999.9,M,	10.2,a
1901,	2.8,a,	3.4,a,	8.4,a,	8.8,a,	16.3,a,	15.4,a,	16.3,a,	18.5,a,	13.8,a,	11.9,a,	6.8,a,	3.6,a,	10.5,a,	3.9,a,	10.9,a,	16.7,a,	11.2,a
1902,	2.5,a,	-9999.9,M,	-9999.9,M,	-9999.9,M,	-9999.9,M,	-9999.9,M,	17.0,a,	16.1,a,	14.7,a,	11.2,a,	4.1,a,	2.6,a,	-9999.9,M,	-9999.9,M,	-9999.9,M,	-9999.9,M,	10.0,a
1903,	3.8,a,	3.6,a,	4.4,a,	8.3,a,	12.1,a,	16.8,a,	16.5,a,	16.8,a,	13.5,a,	9.2,a,	4.4,a,	4.6,a,	9.5,a,	3.3,a,	8.3,a,	16.7,a,	9.8,a
1904,	2.6,a,	2.1,a,	6.4,a,	12.3,a,	13.4,a,	14.8,a,	18.3,a,	17.7,a,	14.4,a,	11.3,a,	8.3,a,	4.8,a,	10.6,a,	3.1,a,	10.7,a,	16.7,a,	11.7,a
1905,	3.3,a,	3.8,a,	10.5,a,	12.5,a,	14.3,a,	15.1,a,	20.9,a,	17.5,a,	12.1,a,	7.3,a,	5.6,a,	4.5,a,	10.5,a,	3.7,a,	12.4,a,	17.7,a,	8.3,a
1906,	3.8,a,	6.8,a,	6.8,a,	11.5,a,	12.9,a,	15.1,a,	21.0,a,	17.8,a,	12.6,a,	10.5,a,	4.4,a,	3.5,a,	10.6,a,	5.8,a,	10.4,a,	16.8,a,	9.2,a
1907,	-6.7,a,	2.4,a,	6.8,a,	9.3,a,	14.2,a,	15.2,a,	18.3,a,	15.5,a,	14.1,a,	11.9,a,	9.3,a,	3.3,a,	9.4,a,	.4,a,	9.2,a,	16.3,a,	11.8,a
1908,	3.7,a,	3.7,a,	6.8,a,	10.6,a,	11.9,a,	17.3,a,	18.9,a,	17.2,a,	12.8,a,	9.3,a,	7.7,a,	3.3,a,	10.1,a,	3.6,a,	9.5,a,	17.8,a,	9.7,a
1909,	-3.8,a,	2.4,a,	7.1,a,	8.8,a,	11.7,a,	15.1,a,	16.7,a,	16.5,a,	14.3,a,	9.8,a,	5.2,a,	.4,a,	8.8,a,	.9,a,	9.2,a,	16.1,a,	9.8,a
1910,	3.8,a,	1.4,a,	9.8,a,	10.4,a,	14.1,a,	17.1,a,	19.3,a,	15.7,a,	15.1,a,	10.5,a,	7.4,a,	3.9,a,	10.6,a,	1.6,a,	11.2,a,	17.4,a,	11.0,a
1911,	-2.1,a,	1.6,a,	5.5,a,	8.4,a,	12.1,a,	15.1,a,	19.1,a,	17.8,a,	14.1,a,	10.8,a,	3.6,a,	4.1,a,	9.1,a,	1.1,a,	8.7,a,	17.1,a,	9.5,a
1912,	1.6,a,	5.6,a,	6.8,a,	8.2,a,	14.6,a,	16.1,a,	16.6,a,	16.4,a,	14.0,a,	9.3,a,	5.9,a,	4.8,a,	9.9,a,	3.8,a,	9.6,a,	16.4,a,	9.7,a
1913,	-1.6,a,	1.3,a,	4.7,a,	11.1,a,	11.7,a,	15.8,a,	17.1,a,	18.2,a,	14.0,a,	9.2,a,	6.7,a,	4.4,a,	9.3,a,	1.2,a,	9.2,a,	16.8,a,	10.0,a
1914,	3.3,a,	4.2,a,	7.5,a,	11.2,a,	13.9,a,	14.5,a,	17.1,a,	17.6,a,	12.9,a,	10.7,a,	6.1,a,	1.5,a,	10.0,a,	4.8,a,	10.9,a,	16.4,a,	9.9,a
1915,	2.7,a,	4.9,a,	9.1,a,	10.3,a,	12.8,a,	15.6,a,	17.4,a,	19.3,a,	14.2,a,	9.4,a,	4.3,a,	3.3,a,	10.3,a,	3.8,a,	10.7,a,	17.4,a,	9.3,a
1916,	-5.5,a,	2.3,a,	5.1,a,	9.6,a,	11.9,a,	16.1,a,	16.4,a,	18.3,a,	15.1,a,	9.6,a,	4.1,a,	.8,a,	8.6,a,	.8,a,	8.9,a,	16.9,a,	9.6,a
1917,	.3,a,	.3,a,	3.9,a,	7.2,a,	12.4,a,	14.8,a,	17.4,a,	16.5,a,	15.1,a,	9.6,a,	7.6,a,	2.3,a,	9.1,a,	.2,a,	7.8,a,	16.6,a,	10.8,a
1918,	2.9,a,	2.4,a,	4.7,a,	10.8,a,	11.9,a,	16.8,a,	18.1,a,	16.9,a,	13.2,a,	11.3,a,	6.1,a,	3.7,a,	10.3,a,	2.5,a,	8.9,a,	17.3,a,	12.2,a
1919,	3.6,a,	2.8,a,	6.1,a,	10.3,a,	11.7,a,	14.7,a,	18.3,a,	16.8,a,	16.4,a,	8.8,a,	5.1,a,	1.5,a,	9.7,a,	3.4,a,	9.4,a,	17.8,a,	9.8,a
1920,	2.4,a,	4.4,a,	6.4,a,	8.9,a,	12.8,a,	16.8,a,	19.5,a,	19.5,a,	14.1,a,	8.7,a,	8.3,a,	4.8,a,	10.4,a,	2.8,a,	9.4,a,	18.3,a,	10.4,a
1921,	2.7,a,	4.4,a,	6.3,a,	10.8,a,	12.7,a,	15.5,a,	17.3,a,	17.5,a,	12.2,a,	11.4,a,	4.2,a,	1.3,a,	9.6,a,	3.7,a,	9.7,a,	16.8,a,	9.3,a
1922,	.8,a,	-.3,a,	4.2,a,	8.3,a,	12.9,a,	16.8,a,	18.2,a,	17.8,a,	15.2,a,	12.3,a,	5.6,a,	-.9,a,	9.2,a,	.3,a,	8.5,a,	17.6,a,	11.0,a
1923,	2.4,a,	.8,a,	6.8,a,	11.6,a,	12.4,a,	17.1,a,	19.1,a,	19.8,a,	16.4,a,	12.9,a,	7.8,a,	2.6,a,	10.7,a,	.8,a,	10.0,a,	18.4,a,	12.4,a
1924,	2.1, ,	5.9, ,	6.1, ,	8.8, ,	15.2, ,	15.2, ,	17.5, ,	17.2, ,	14.9, ,	10.1, ,	5.2, ,	.1, ,	9.9, ,	3.5,a,	10.0, ,	16.6, ,	10.1,
1925,	2.9, ,	6.2, ,	6.3, ,	10.8, ,	16.0, ,	16.4, ,	18.9, ,	17.6, ,	15.5, ,	9.3, ,	5.8, ,	6.7, ,	11.0, ,	3.1, ,	11.0, ,	17.6, ,	10.2,
1926,	3.8, ,	5.7, ,	10.2, ,	13.1, ,	13.2, ,	17.8, ,	18.9, ,	18.2, ,	14.1, ,	11.2, ,	7.8, ,	2.2, ,	11.4, ,	5.4, ,	12.2, ,	18.8, ,	11.4,
1927,	.6, ,	4.8, ,	5.4, ,	8.4, ,	11.8, ,	17.2, ,	19.1, ,	19.8, ,	14.7, ,	10.3, ,	4.4, ,	-.4, ,	9.5, ,	2.3, ,	8.5, ,	18.4, ,	9.8,
1928,	2.5, ,	5.3, ,	8.8, ,	9.6, ,	15.7, ,	15.8, ,	18.9, ,	18.8, ,	14.7, ,	10.2, ,	6.7, ,	3.2, ,	10.7, ,	2.5, ,	11.1, ,	17.5, ,	10.5,
1929,	-1.4, ,	.6, ,	6.5, ,	8.1, ,	13.5, ,	15.7, ,	17.9, ,	17.7, ,	16.8, ,	11.3, ,	5.9, ,	2.8, ,	9.6, ,	.8, ,	9.6, ,	17.1, ,	11.7,
1930,	-3.6, ,	4.7, ,	7.4, ,	11.4, ,	12.4, ,	15.5, ,	18.2, ,	18.6, ,	15.5, ,	9.6, ,	5.9, ,	4.4, ,	10.0, ,	1.8, ,	10.4, ,	17.4, ,	10.3,
1931,	6.2, ,	5.6, ,	7.4, ,	11.4, ,	14.6, ,	15.8, ,	19.0, ,	17.8, ,	14.4, ,	11.8, ,	6.1, ,	2.5, ,	10.8, ,	5.3, ,	11.1, ,	17.5, ,	9.8,
1932,	.9, ,	2.5, ,	6.4, ,	10.7, ,	13.2, ,	16.5, ,	15.9, ,	17.9, ,	15.3, ,	11.3, ,	7.5, ,	1.2, ,	9.9, ,	2.8, ,	10.1, ,	16.8, ,	11.4,
1933,	2.1, ,	-.4, ,	6.8, ,	9.4, ,	11.6, ,	14.5, ,	17.0, ,	19.7, ,	13.3, ,	10.5, ,	7.6, ,	1.8, ,	9.4, ,	1.8, ,	9.0, ,	17.1, ,	10.5,
1934,	4.7, ,	7.5, ,	9.3, ,	13.6, ,	14.3, ,	16.8, ,	16.9, ,	18.6, ,	14.1, ,	11.5, ,	8.1, ,	3.4, ,	11.6, ,	4.7, ,	12.4, ,	17.2, ,	11.4,
1935,	-1.2, ,	6.3, ,	6.4, ,	9.3, ,	13.0, ,	15.1, ,	17.6, ,	17.8, ,	16.8, ,	9.7, ,	4.7, ,	6.3, ,	10.0, ,	2.8, ,	8.9, ,	16.8, ,	10.6,
1936,	3.5, ,	-3.3, ,	4.8, ,	11.8, ,	14.4, ,	16.8, ,	17.1, ,	18.5, ,	14.5, ,	12.3, ,	6.7, ,	3.9, ,	9.9, ,	2.2, ,	9.8, ,	17.3, ,	11.2,
1937,	-4.1, ,	.2, ,	8.3, ,	7.8, ,	13.1, ,	16.8, ,	17.8, ,	17.8, ,	15.9, ,	11.8, ,	6.5, ,	2.9, ,	9.6, ,	.8, ,	9.7, ,	17.1, ,	11.7,
1938,	3.8, ,	4.8, ,	7.1, ,	10.8, ,	13.7, ,	16.3, ,	19.1, ,	17.8, ,	17.2, ,	12.5, ,	5.7, ,	3.1, ,	10.8, ,	3.3, ,	10.5, ,	17.5, ,	11.8,
1939,	4.3, ,	1.3, ,	7.8, ,	10.5, ,	13.5, ,	14.1, ,	17.7, ,	19.3, ,	15.8, ,	10.8, ,	8.7, ,	6.7, ,	10.8, ,	2.9, ,	10.5, ,	17.8, ,	11.8,
1940,	4.5, ,	5.2, ,	8.6, ,	11.7, ,	14.8, ,	16.8, ,	17.6, ,	16.5, ,	18.0, ,	12.4, ,	4.5, ,	5.2, ,	11.5, ,	5.5, ,	11.7, ,	17.8, ,	12.0,
1941,	5.1, ,	7.8, ,	10.9, ,	11.7, ,	13.2, ,	16.8, ,	18.4, ,	18.3, ,	13.8, ,	11.8, ,	7.5, ,	4.4, ,	11.7, ,	6.8, ,	11.9, ,	18.2, ,	11.8,
1942,	4.3, ,	5.3, ,	6.8, ,	11.1, ,	13.3, ,	15.8, ,	18.0, ,	20.1, ,	16.0, ,	11.7, ,	5.8, ,	3.8, ,	11.1, ,	4.7, ,	10.4, ,	18.6, ,	11.2,
1943,	-3.1, ,	6.2, ,	5.2, ,	11.1, ,	11.8, ,	15.8, ,	18.0, ,	16.9, ,	17.1, ,	11.4, ,	7.7, ,	4.2, ,	10.2, ,	2.2, ,	9.4, ,	16.9, ,	12.1,

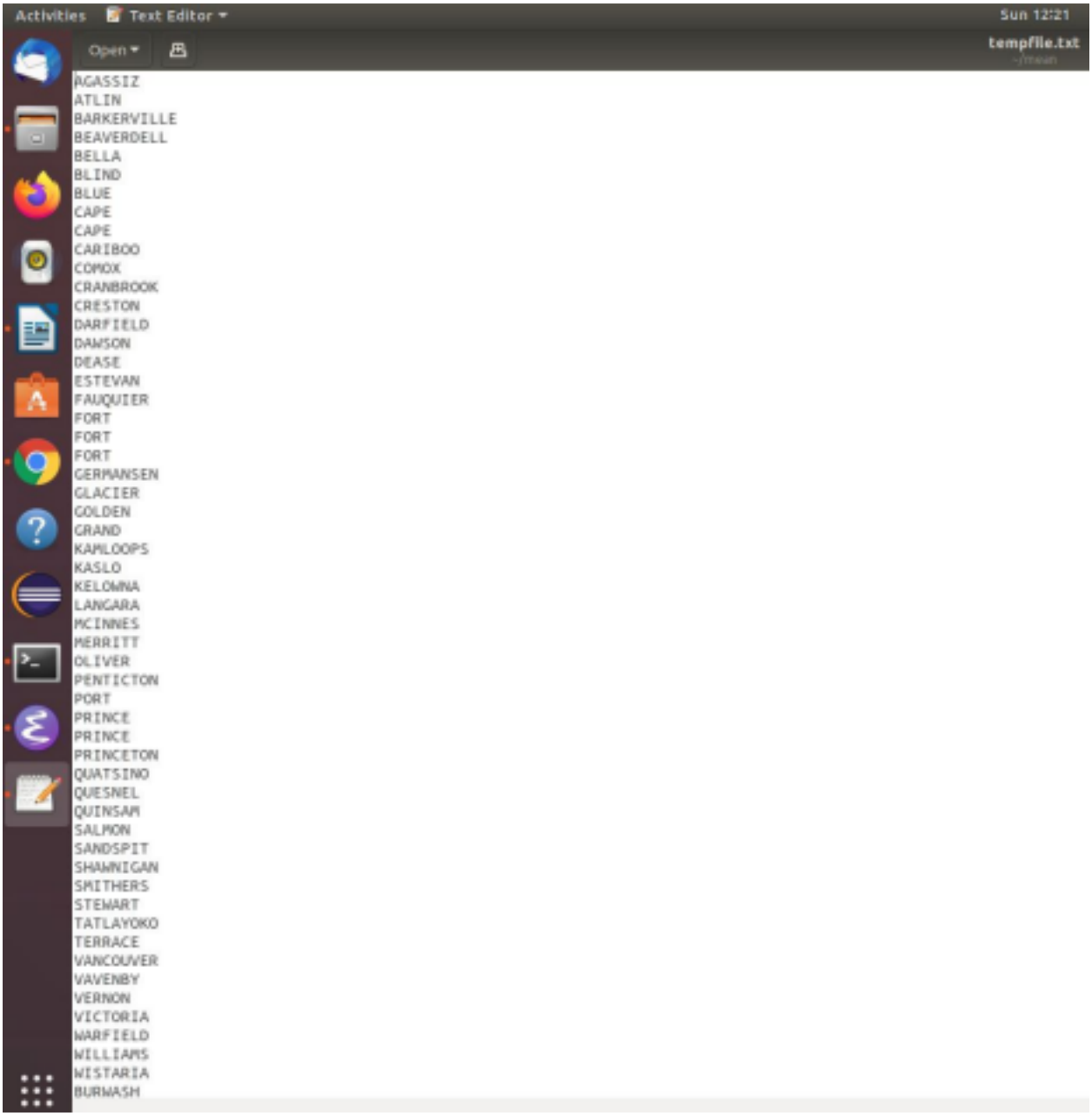
After replacing **“,”** with **“\n” (“BIGDATA8zx2756.txt” file)**



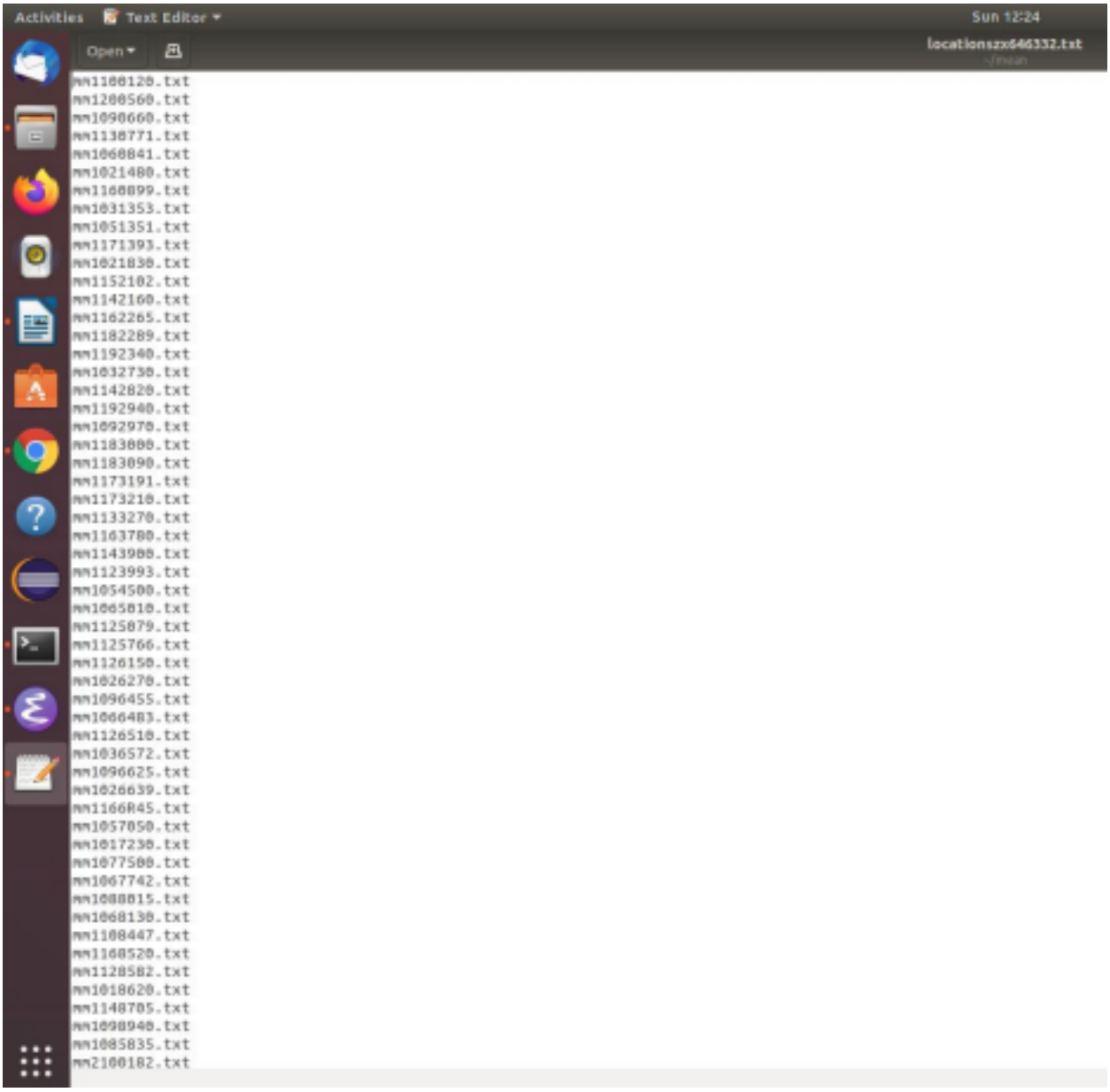
Similarly For Loop will execute for other lines in "BTemperature_Stations.txt".

After execution of the FOR LOOP:

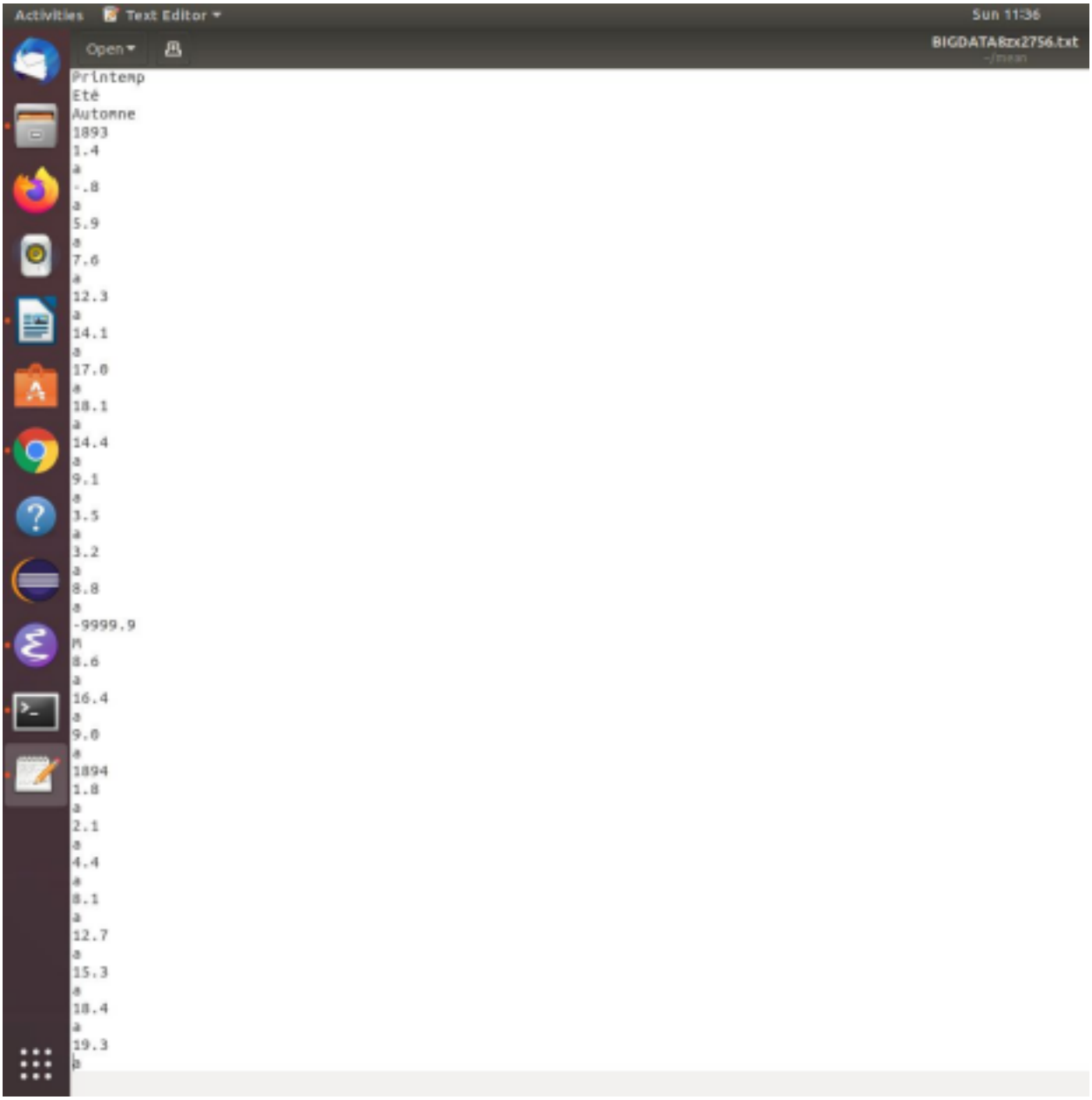
“tempfile.txt”



“locationszx646332.txt”



“BIGDATA8zx2756.txt”



After execution of the for loop now we have all the data, locations and names of the particular location in order.

Continuing further

```
cat $alldatafile | grep -A 17 $a >> $smalldatafile
```

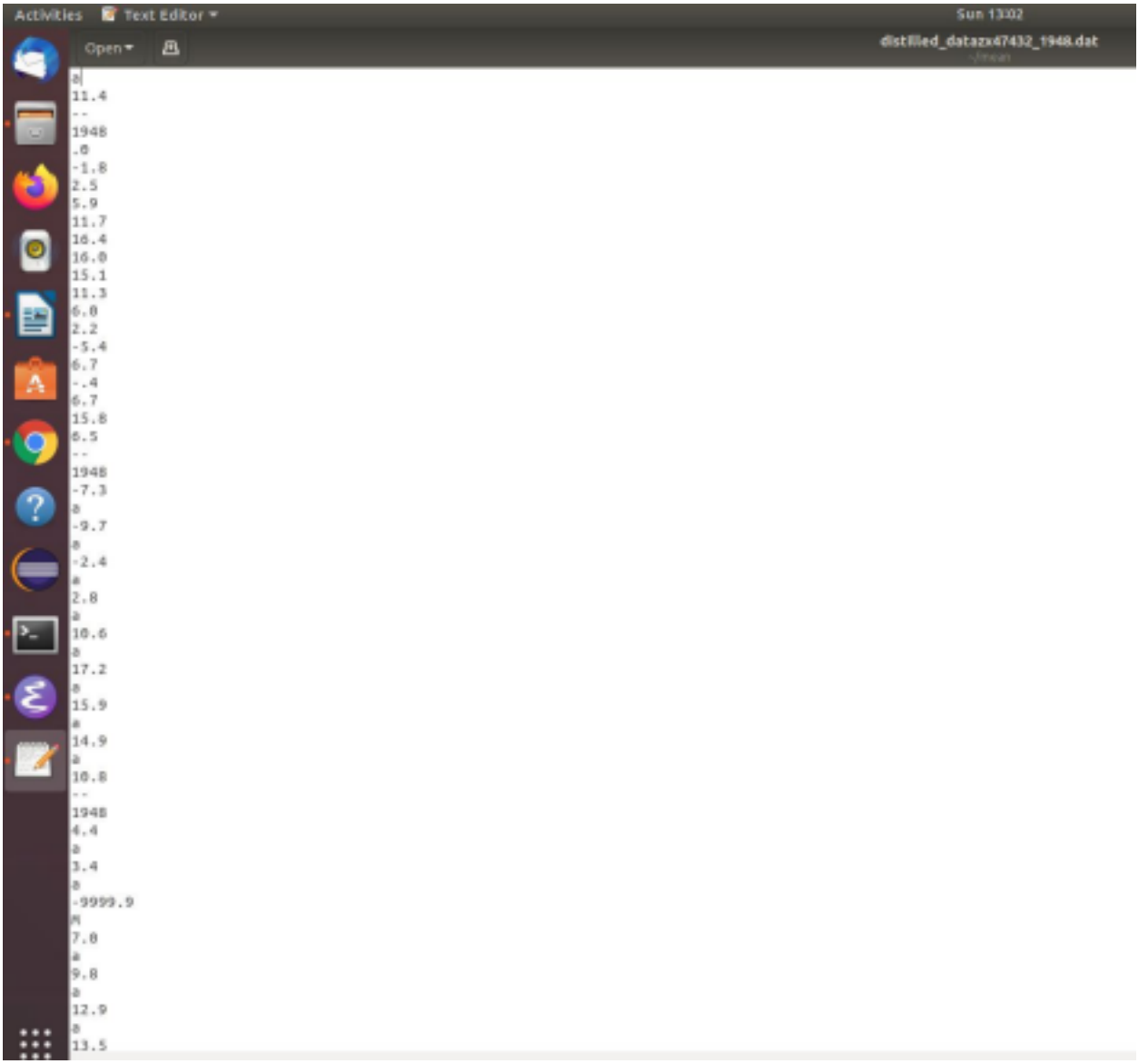
For the first loop(example : `$a=1948`)

cat sends data from the **“BIGDATA8zx2756.txt”** to STDOUT
grep -A 17 searches for 17 lines after $a(i.e the input year from user (e.g 1948)

Piping is used to get the data from STDOUT and then search for 17 lines after a particular year entered by the user.

Then that data is sent to "distilled_datazx47432_$1.dat"

.
.
.
.
.
.
.
"distilled_datazx47432_$1.dat"



THEN

```
head -n $1 $tempfile | tail -n 1 >>$newfile1
```

“$1”(example user enters 200)
head -n $1 $tempfile will display 200 names to STDOUT from “tempfile.txt”.--------->piped to **tail -n 1**

So it will print the 200th name in the “testdata.dat”

--->

```
cat $smalldatafile | grep -A $NUMCONTEXTLINES -m $1 $a | tail -n
$NCLPLUSONE >> $newfile1
```

Here,

NUMCONTEXTLINES=1
NCLPLUSONE=NUMCONTEXTLINES+1

cat $smalldatafile will send data from **"distilled_datazx47432_$1.dat"** to the **STDOUT**

grep -A 1 -m $1 $a will search for 1 line at the 200th position of that particular year entered by the user

tail -n 2
This will display last 2 line to the STDOUT (i.e **year and temperature**) and then we are printing that to “testdata.dat”

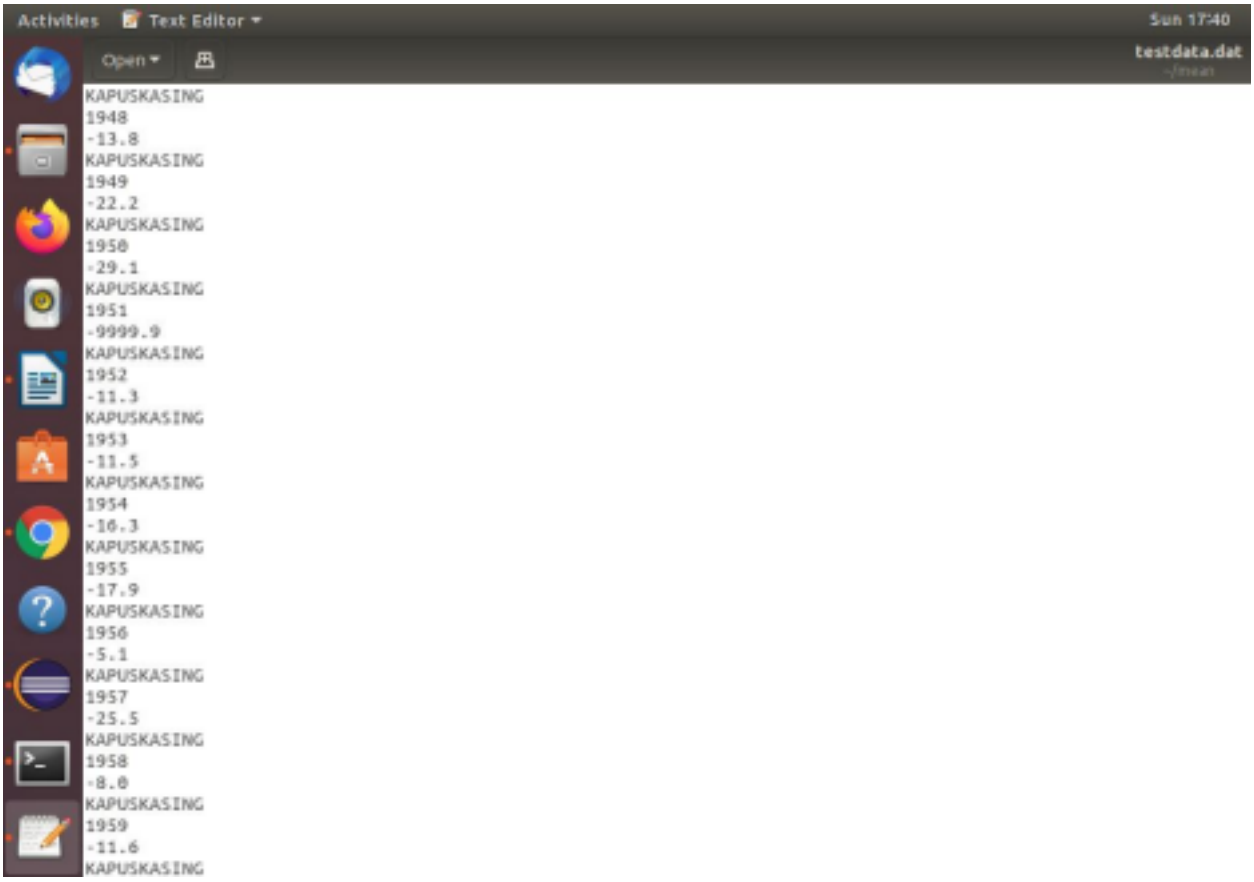
```
echo "-9999" >> $newfile1
echo "-9999" >> $newfile1
echo "-9999" >> $newfile1
```

This will add -9999 (3 times at the end of the testdata.dat)

```
rm $alldatafile; rm $smalldatafile; rm $locationsfile;rm $tempfile
```

rm will remove all temporary files created.

Suppose user entered(1948 1997)
So the for loop executes from 1948 till 1997.
OUTPUT:



CONCLUSION :

BASH SCRIPT#2 is helpful in printing out data of differrent years of a particular location It will be easy for users to use this script instead of searching manually each and every file.

APPENDIX:

```
#!/bin/bash
NUMCONTEXTLINES=1 # works up to 17
let NCLPLUSONE=NUMCONTEXTLINES+1
```

```
# temporary file definitionsBTstns="BTemperature_Stations.txt"
alldatafile="BIGDATA8zx2756.txt"
smalldatafile="distilled_datazx47432_$1.dat"
locationsfile="locationszx646332.txt"
tempfile="tempfile.txt"
newfile1="testdata.dat"
# extract station IDs from BTemperature_Stations.txt
for((a=$2;a<=$3;a++))
do
# scan lines from line 5 to 343
for x in {5..343}
do
 next=$(head -n $x $BTstns | tail -n 1) # read line x from
BTemperature_Stations.txt
 line=($next)
 stationNUM=(${line[0]}) # station number
stationID=(${line[1]}) # station ID
 stationNAME=(${line[2]}) # station name
nextfile=mm$stationID.txt
 newfile=$stationNAME
 echo "$newfile" >>$tempfile
 echo "$nextfile" >> $locationsfile # write the data file name first
 y=$(cat $nextfile | tr "," "\n") # remove commas, replace with
newlines
 for z in $y # go through each token in the file $x
 do
 echo $z >> $alldatafile
 done
done
# scan for a particular year; there should be as many of a given year
# as there are geographical locations
yearsearch=$2 # year provided as argument

cat $alldatafile | grep -A 17 $a >> $smalldatafile
head -n $1 $tempfile | tail -n 1 >>$newfile1
```

```
# extract year's info for location specified in $2

cat $smalldatafile | grep -A $NUMCONTEXTLINES -m $1 $a | tail -n
$NCLPLUSONE >> $newfile1 # send the target year's temp data to
STDOUT
rm $alldatafile; rm $smalldatafile; rm $locationsfile;rm $tempfile;

done

echo "-9999" >> $newfile1
echo "-9999" >> $newfile1
echo "-9999" >> $newfile1

exit 0
```