Q5. Explain the usage of the following bash script.

The given script is as follows.

```
#!/bin/bash
limit=5 # seconds to wait for timeout
file=$1
timeout=$(awk "BEGIN{srand(); print srand() + $limit}")
echo $timeout
until [ -s "$file" ] || [ $(awk ’BEGIN{srand();print srand()}’) -gt $timeout ]
do
sleep 1
done
if ! [ -s "$file" ]
then
# timed-out
exit 1
else
cat $file
fi
```

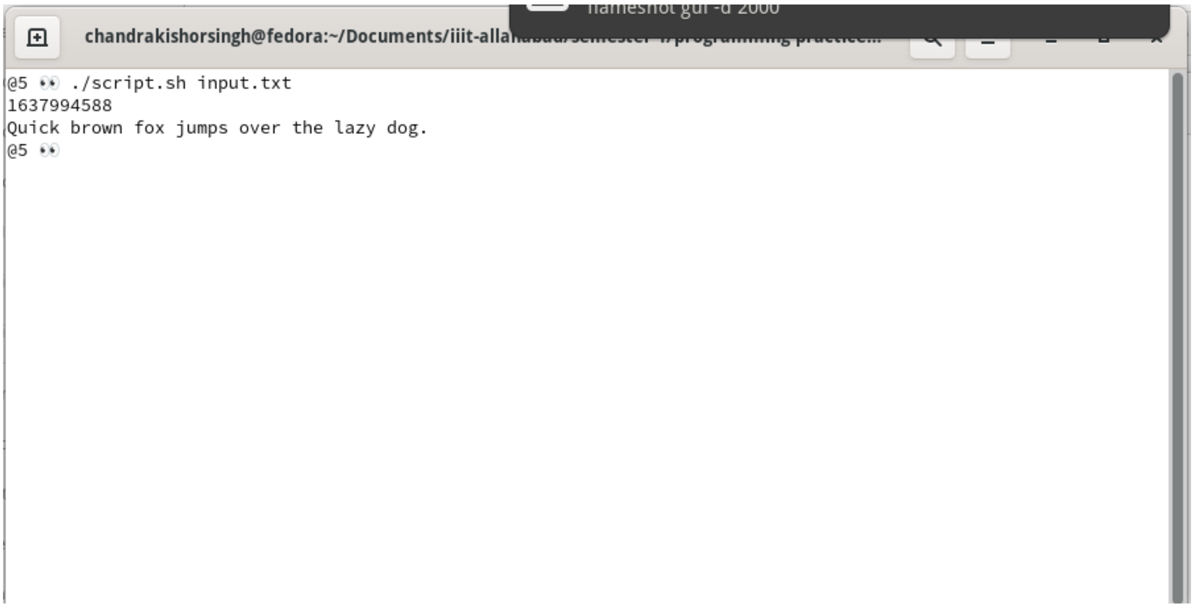
The given script will print the current time + 5 and the content of the file(provided as first argument to the script) if the given path represents a valid file and is not empty. Otherwise, it will just print the timeout and wait for 5 seconds to give back the prompt.

I have commented the given code to explain the main logic of the program. The commented code is as follows.

```
#!/bin/bash

# assign `5` to `limit` variable
limit=5 # seconds to wait for timeout
# assign the first argument to `file` variable
file=$1

# initialize the srand() function by current time and assign a `current time + 5` to the `timeout` variable
timeout=$(awk "BEGIN{srand(); print srand() + $limit}")
# print the `timeout` variable
echo $timeout

# until the current time is not greater than `timeout` variable or the file(whose path is assigned to `file` variable) is exists and is not empty, sleep for 1 second
until [ -s "$file" ] || [ $(awk "BEGIN{srand();print srand()}") -gt $timeout ]
do
    sleep 1
done
```

```
# if file does not exists or if it is empty, then exit with status of 1.
otherwise show the contents of file using cat command.
if ! [ -s "$file" ]
then
    # timed-out
    exit 1
else
    cat $file
fi
```

Its output is as follows.


flameshot gui -d 2000
chandrakishorsingh@fedora:~/Documents/iiit-allahabad/semester-7/programming practice...

```
@5 👀 ./script.sh input.txt
1637994588
Quick brown fox jumps over the lazy dog.
@5 👀
```