Q1. Write shell script to read two command line argument(2 numbers) and display the sum of the two number(excluding the script name).It should display an error message if the number of arguments is less than Also write the command on the prompt to run the file:

if [[$# -ne 2]] –OR- if ["$#" –ne 2]

then

echo "2 numbers as command line arguments please"

else

sum=$(($1 + $2))

fi

echo $sum

Running command: ./q1 3 4

Q2. Create a directory PQR. Create a symbolic link in the current user's home directory named PQR to the file /data/hello:

mkdir PQR

ln –s /data/hello PQR

Q3. Write a shell program to accept 10 numbers from the user and count the number of odd and even numbers in the list:

odd=0

even=0

for i in {1..10}

do

echo "Enter Number"

read numberFromUser

number=$(($numberFromUser % 2))

if [$number –eq 0]

then

```
((even++))
else
((odd++))
fi
done
echo “Even counter: $even”
echo “Odd counter: $odd”
```

Q4. Draw and Explain the life cycle of a thread:

There are four states of a thread’s lifecycle:

1. Ready: ready and able to run, but is waiting for a processor.

2. Running: currently running on a multiprocessor.

3. Blocked: not able to run because it is waiting for something. Example: mutex, i/o

4. Terminated: terminated by returning from its start function.

Q5. What is a deadlock and how can it be resolved?

It is a situation in which two thread/process sharing the same resource are effectively preventing each other from accessing the resource, resulting in both programs ceasing to function.

Using mutex and condition variables prevent this issue.

Q6. What are conditional variable?

Sometimes locking or unlocking a variable is less resource-consuming than continuously scanning the variable. Conditional variables work with mutex to provide a way for threads to synchronize.

From https://computing.llnl.gov/tutorials/pthreads/#ConditionVariables:

“While mutexes implement synchronization by controlling thread access to data, condition variables allow threads to synchronize based upon the actual value of data”

based on a run-time condition. Without condition variables, program would have to poll the variable/condition continuously, which indeed is time consuming