Name: Harsh Chaudhari **Class:** BE-10
Batch: P-10 **Roll no:** 43215

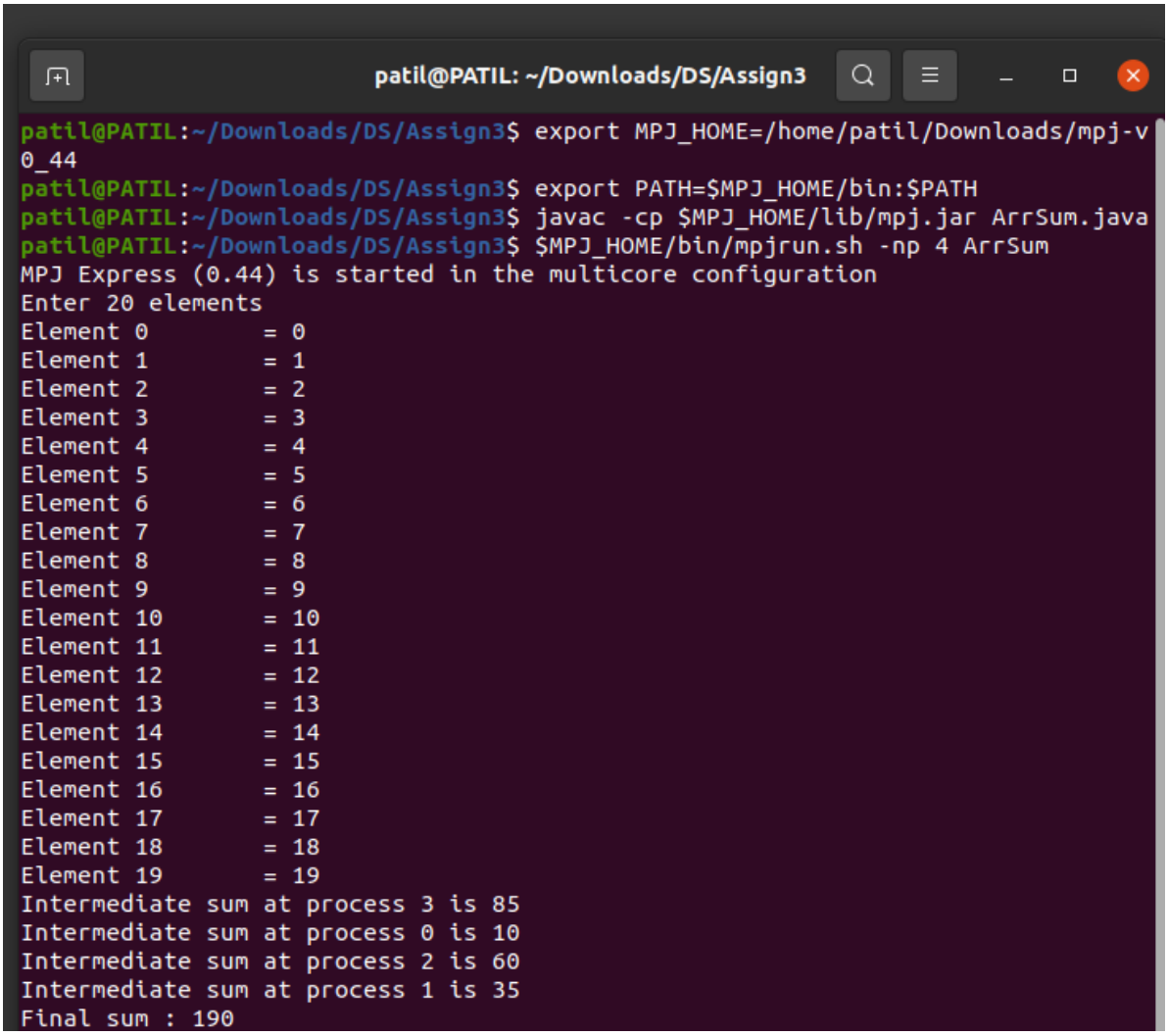
Code:

ArrSum.java

```
// client
import mpi.MPI;
import java.util.Scanner;
import mpi.*;
public class ArrSum {
   public static void main(String[] args) throws Exception{
      MPI.Init(args);
      int rank = MPI.COMM_WORLD.Rank();
      int size = MPI.COMM_WORLD.Size();
      int unitsize = 5;
      int root = 0;
      int send_buffer[] = null;
      //  1 process is expected to handle 4 elements
      send_buffer = new int [unitsize * size];
      int recieve_buffer[] = new int [unitsize];
      int new_recieve_buffer[] = new int [size];
      //  Set data for distribution
      if(rank == root) {
         int total_elements = unitsize * size;
         System.out.println("Enter " + total_elements + " elements");
```

```
    for(int i = 0; i < total_elements; i++) {
        System.out.println("Element " + i + "\t = " + i);
        send_buffer[i] = i;
    }
}
// Scatter data to processes
MPI.COMM_WORLD.Scatter(
    send_buffer,
    0,
    unitsize,
    MPI.INT,
    recieve_buffer,
    0,
    unitsize,
    MPI.INT,
    root
);
// Calculate sum at non root processes
// Store result in first index of array
for(int i = 1; i < unitsize; i++) {
    recieve_buffer[0] += recieve_buffer[i];
}
System.out.println(
    "Intermediate sum at process " + rank + " is " + recieve_buffer[0]
);
```

```
    // Gather data from processes
    MPI.COMM_WORLD.Gather(
        recieve_buffer,
        0,
        1,
        MPI.INT,
        new_recieve_buffer,
        0,
        1,
        MPI.INT,
        root
    );
    // Aggregate output from all non root processes
    if(rank == root) {
     int total_sum = 0;
        for(int i = 0; i < size; i++) {
            total_sum += new_recieve_buffer[i];
        }
        System.out.println("Final sum : " + total_sum);
    }
    MPI.Finalize();
}}
```


patil@PATIL: ~/Downloads/DS/Assign3
patil@PATIL:~/Downloads/DS/Assign3$ export MPJ_HOME=/home/patil/Downloads/mpj-v0_44
patil@PATIL:~/Downloads/DS/Assign3$ export PATH=$MPJ_HOME/bin:$PATH
patil@PATIL:~/Downloads/DS/Assign3$ javac -cp $MPJ_HOME/lib/mpj.jar ArrSum.java
patil@PATIL:~/Downloads/DS/Assign3$ $MPJ_HOME/bin/mpjrun.sh -np 4 ArrSum
MPJ Express (0.44) is started in the multicore configuration
Enter 20 elements
Element 0 = 0
Element 1 = 1
Element 2 = 2
Element 3 = 3
Element 4 = 4
Element 5 = 5
Element 6 = 6
Element 7 = 7
Element 8 = 8
Element 9 = 9
Element 10 = 10
Element 11 = 11
Element 12 = 12
Element 13 = 13
Element 14 = 14
Element 15 = 15
Element 16 = 16
Element 17 = 17
Element 18 = 18
Element 19 = 19
Intermediate sum at process 3 is 85
Intermediate sum at process 0 is 10
Intermediate sum at process 2 is 60
Intermediate sum at process 1 is 35
Final sum : 190