Assignment No 4

Code

A) BerkeleyAlgorithm.java

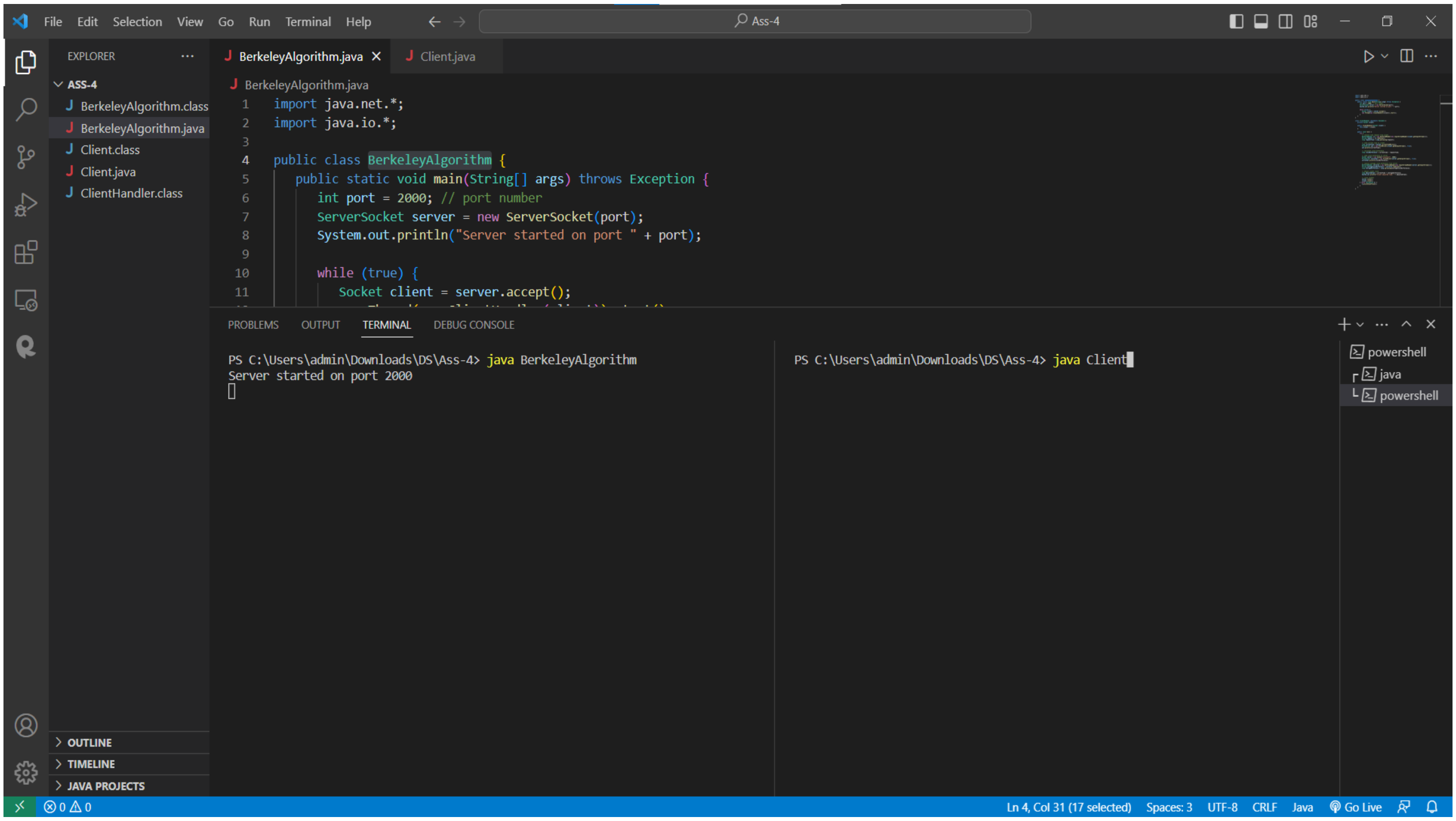
```
import java.net.*;
import java.io.*;

public class BerkeleyAlgorithm {
  public static void main(String[] args) throws Exception {
    int port = 2000; // port number
    ServerSocket server = new ServerSocket(port);
    System.out.println("Server started on port " + port);

    while (true) {
      Socket client = server.accept();
      new Thread(new ClientHandler(client)).start();
    }
  }
}

class ClientHandler implements Runnable {
  private Socket client;

  public ClientHandler(Socket client) {
    this.client = client;
  }

  public void run() {
    try {
      // receive time request from client
      BufferedReader in = new BufferedReader(new
InputStreamReader(client.getInputStream()));
      String request = in.readLine();
      long requestTime = Long.parseLong(request);

      // send current time to client
      long currentTime = System.currentTimeMillis();
      PrintWriter out = new PrintWriter(client.getOutputStream(), true);
      out.println(currentTime);

      // calculate clock difference
      long clockDifference = currentTime - requestTime;
      // send clock difference to server
      Socket server = new Socket("localhost", 2000);
      PrintWriter serverOut = new PrintWriter(server.getOutputStream(), true);
      serverOut.println(clockDifference);

      // receive average clock difference from server
      BufferedReader serverIn = new BufferedReader(new
InputStreamReader(server.getInputStream()));
      String averageClockDifference = serverIn.readLine();
      long averageDifference = Long.parseLong(averageClockDifference);

      // adjust client clock
      long adjustedTime = currentTime + averageDifference;
      System.out.println("Client adjusted time: " + adjustedTime);

      // close sockets
      server.close();
      client.close();
    } catch (Exception e) {
      e.printStackTrace();
    }
  }
}
```

B) Client.java

```
import java.net.*;

import java.io.*;


public class Client {

    public static void main(String[] args) throws Exception {

        Socket server = new Socket("localhost", 2000);


        // get current time

        long currentTime = System.currentTimeMillis();
        // send time request to server
        PrintWriter out = new PrintWriter(server.getOutputStream(), true);
        out.println(currentTime);

        // receive current time from server
        BufferedReader in = new BufferedReader(new
InputStreamReader(server.getInputStream()));
        String response = in.readLine();
        long serverTime = Long.parseLong(response);

        // calculate clock difference
        long clockDifference = serverTime - currentTime;

        // send clock difference to server
        PrintWriter serverOut = new PrintWriter(server.getOutputStream(), true);
        serverOut.println(clockDifference);

        // receive average clock difference from server
        BufferedReader serverIn = new BufferedReader(new
InputStreamReader(server.getInputStream()));
        String averageClockDifference = serverIn.readLine();
        long averageDifference = Long.parseLong(averageClockDifference);

        // adjust client clock
        long adjustedTime = serverTime + averageDifference;
        System.out.println("Client adjusted time: " + adjustedTime);

        // close sockets
```

```
        server.close();
    }
}
```

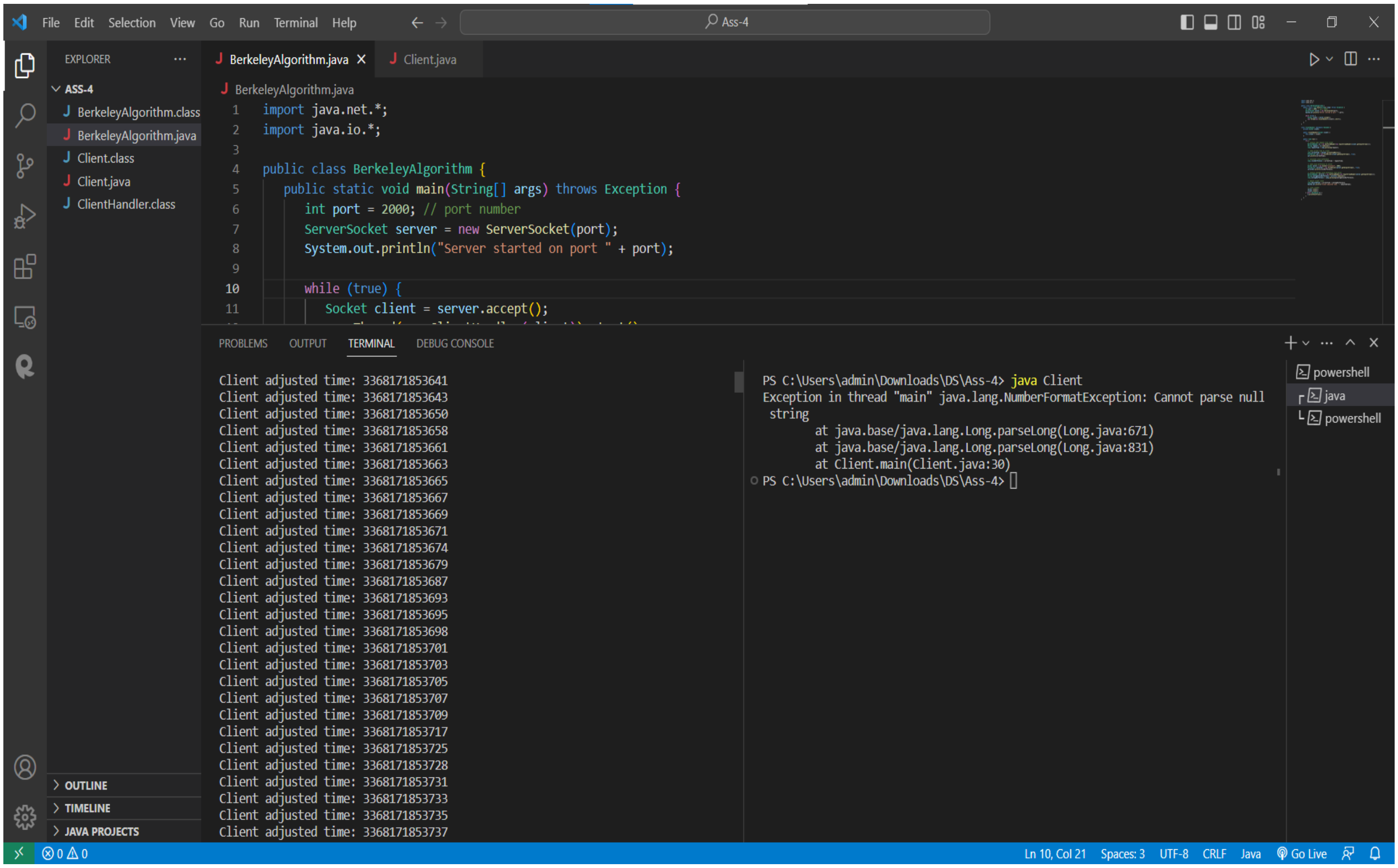
# Output



```java
import java.net.*;
import java.io.*;

public class BerkeleyAlgorithm {
   public static void main(String[] args) throws Exception {
      int port = 2000; // port number
      ServerSocket server = new ServerSocket(port);
      System.out.println("Server started on port " + port);

      while (true) {
         Socket client = server.accept();
```

```
PS C:\Users\admin\Downloads\DS\Ass-4> java BerkeleyAlgorithm
Server started on port 2000
```

```
PS C:\Users\admin\Downloads\DS\Ass-4> java Client
```



```java
import java.net.*;
import java.io.*;

public class BerkeleyAlgorithm {
   public static void main(String[] args) throws Exception {
      int port = 2000; // port number
      ServerSocket server = new ServerSocket(port);
      System.out.println("Server started on port " + port);

      while (true) {
         Socket client = server.accept();
```

```
Client adjusted time: 3368171853641
Client adjusted time: 3368171853643
Client adjusted time: 3368171853650
Client adjusted time: 3368171853658
Client adjusted time: 3368171853661
Client adjusted time: 3368171853663
Client adjusted time: 3368171853665
Client adjusted time: 3368171853667
Client adjusted time: 3368171853669
Client adjusted time: 3368171853671
Client adjusted time: 3368171853674
Client adjusted time: 3368171853679
Client adjusted time: 3368171853687
Client adjusted time: 3368171853693
Client adjusted time: 3368171853695
Client adjusted time: 3368171853698
Client adjusted time: 3368171853701
Client adjusted time: 3368171853703
Client adjusted time: 3368171853705
Client adjusted time: 3368171853707
Client adjusted time: 3368171853709
Client adjusted time: 3368171853717
Client adjusted time: 3368171853725
Client adjusted time: 3368171853728
Client adjusted time: 3368171853731
Client adjusted time: 3368171853733
Client adjusted time: 3368171853735
Client adjusted time: 3368171853737
```

```
PS C:\Users\admin\Downloads\DS\Ass-4> java Client
Exception in thread "main" java.lang.NumberFormatException: Cannot parse null
 string
        at java.base/java.lang.Long.parseLong(Long.java:671)
        at java.base/java.lang.Long.parseLong(Long.java:831)
        at Client.main(Client.java:30)
PS C:\Users\admin\Downloads\DS\Ass-4>
```