

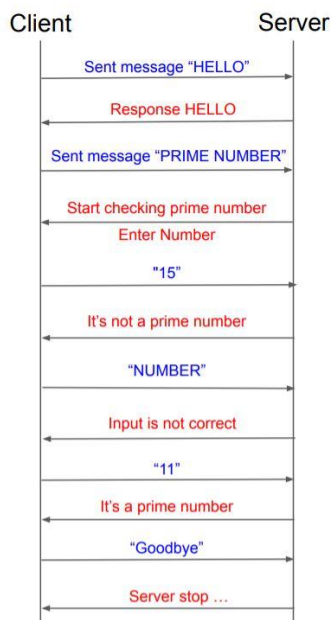
Socket Assignment (Part 2)

Prime Number

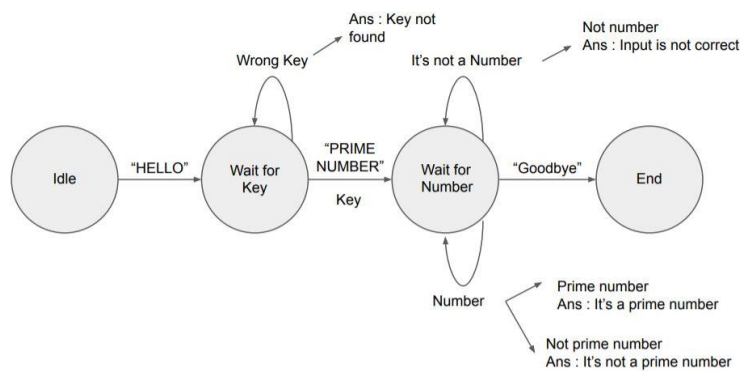
หลักการทำงาน

เริ่มต้นการทำงานเมื่อฝั่ง Client จะทำการส่ง “Hello” ไปฝั่ง Server จะทำการส่ง “Hello” กลับมา โดยจะรอรับคีย์ที่เป็นค่าที่กำหนด ถ้าคีย์ถูกต้องจะตรวจสอบตัวเลขว่าเป็น prime number หรือไม่

Sequence Diagram



State Diagram



Code

Client.js

```
var net = require('net');

const readline = require('readline').createInterface({
  input: process.stdin,
  output: process.stdout
})

var HOST = '127.0.0.1';
var PORT = 6969;
```

- Readline กำหนดค่าอินพุตและเอาต์พุต
- กำหนด Host และ Port ให้กับ client

```
var client = new net.Socket();
client.connect(PORT, HOST, function() {
  console.log('CONNECTED TO: ' + HOST + ':' + PORT);
  readline.question('\n', input => {
    client.write(input);
  })
});
```

- ทำการสร้าง Socket เพื่อเชื่อมต่อ server และส่งค่าจาก client ไปยัง server

```
client.on('data', function(data) {
  readline.question(data + '\n', input => {
    client.write(input);
  })
});

client.on('close', function() {
  console.log('Connection closed');
});
```

- รับค่าข้อความจาก server มาแสดงที่ client และรับค่าจาก client ส่งไปยัง server

Server.js

```

var net = require('net');
var HOST = '127.0.0.1';
var PORT = 6969;

var db = {}
var state = 0 //idle
var current_key = null

const isPrime = num => {
  for (let i = 2; i < num; i++)
    if (num % i === 0) return false;
  return num > 1;
}

```

- กำหนด Host และ Port
- ประกาศตัวแปร state วนลูปและฟังก์ชัน isPrime เพื่อในการหาค่าจำนวนเฉพาะ

```

net.createServer(function (sock) {
  sock.on('data', function (data) {
    switch (state) {
      case 0:
        if (data == 'HELLO') {
          sock.write('HELLO')
          state = 1 //wait for key
        }
        break
      case 1:
        if (data == 'PRIME NUMBER') {
          sock.write('Enter Number')
          state = 2
        }
        else {
          sock.write('ERROR')
        }
        break
    }
  })
})

```

- เมื่อผู้ใช้ input เข้ามาเป็น “HELLO” server ก็จะตอบกลับมาเป็น “HELLO” เช่นกัน
- เมื่อผู้ใช้ input เข้ามาเป็น “PRIME NUMBER” ก็จะรับตัวเลขจากผู้ใช้เข้ามา หากผู้ใช้ input มาเป็นอย่างอื่นที่ไม่ใช่ “PRIME NUMBER” server ก็จะแสดงผลว่า “ERROR”

```

case 2:
    if (data == 'GOODBYE') {
        sock.close()
        state = 3 //end
    } else {
        if (!isNaN(data)) {
            if (isPrime(data))
                sock.write(data + " is a prime number")
            else
                sock.write('It is not prime number')
        }
        else {
            sock.write('Input is not correct')
        }
    }
    break
}
});

```

- If(!isNaN(data)) เป็นเงื่อนไขเพื่อตรวจสอบว่าค่า input ที่รับเข้ามาทั้งหมด เป็นตัวเลข(int) หรือไม่ หากใช่จะไปเข้าเงื่อนไขเพื่อตรวจสอบต่อว่าเป็น Prime Number หรือไม่ หากไม่ใช่ก็จะแสดงผลว่า “Input is not correct”

- หาก input ที่รับมาเป็นคำว่า “GOODBYE” ก็จะจบการทำงานทันที

ผลลัพธ์

```

HELLO
HELLO
PRIME NUMBER
Enter Number
11
11 is a prime number
12
It is not prime number
num
Input is not correct
GOODBYE

```

สมาชิก

- | | |
|---------------------------|------------|
| 1.นายณัฐวัฒน์ ขาวมัน | 6010110185 |
| 2.นางสาวจริญญา ธนะทวี | 6210110459 |
| 3.นายณัฐพัฒน์ สุรปรีชากุล | 6210110653 |