Exam Policy:

There is no tolerance policy for exams. **You will be asked to leave the exam room immediately without a warning** once you do the following things which mean you'll get **NC**.

1. You are caught cheating or trying to cheat such as look around, speak in your own language not English. You have to look straight and focus on your own exam during exam.
2. Answers should be written with a Pen or Pencil, but if you want to use a pencil please bring your own eraser, sharpener and ruler. You're not allowed to borrow from other students or proctors during exam.
3. All mobile phones should be turned off or put in your bag. All your belongings except stationary must be left in the front of the classroom before the start of the exam.
4. Restroom Policy
    a. Maximum 10 minutes – We'll go to find you to see if you need any help.
    b. No phone in pockets
    c. **Not allowed once a student submit his/her paper.**
5. You're not allowed to go out for water.
6. You're not allowed to ask/get extra papers from other students or proctors. All your notes must be written on the exam paper provided.
7. You're only allowed to provide 1 answer per question. If multiple answers are provided, will use the lowest grade for this question.
8. Programming Question: You must use correct JavaScript or TypeScript syntax to implement the feature match the requirement. Psudo-code or comments to explain the logic don't get credit.
9. During the exam, if you ask questions like: professor, I forgot the syntax of this, could you help me? The answer is NO. or like Professor, is it correct? The answer is I don't know.
10. **Please write down your answer clearly. If I cannot read your answer, you'll not get credit.**
11. **For any question you don't understand, you MUST ask the proctors during the exam. If you claim the question isn't clear when/after we go over the exam solutions, don't accept any excuse.**

Exam time: August 3rd, 2023 10:00 am ~12:30pm. **Arrive at least 5 minutes, so we can start on time.**

Location: V28

Questions in the exam – Total: 40 points

1. True/False questions – 10 questions – 10 points
    a. Lecture 1 – 4 questions – 2 event loop, 2 other part
    b. Lecture 2 – 4 questions – 3 stream, 1 path.join
    c. Lecture 4 – 2 questions – (static, express.static), middleware
2. Short Answer Questions – 3 questions – 15 points
    a. module.exports, exports – 2 questions – 5 points each one
    b. next() – 1 question – 5 points
3. Programming Question – 1 question – 15 points
    a. Like the exercise

Review:

Lecture 1:

Page 23 – 24

Page 28- 36: true/fase, setTimeout, setImmediate, fs.readfile, process.nextTick

Lecture 2:

Page 8 – 12: module.exports, exports – 2 short answer questions

Page 21: path.join() – true/false, find the correct path of a file

Page 22: programming question – you can decide by yourself using sync or async

Page 24 – 28: 3 true/false questions

Lecture 4:

Page 10,11 - middleware, app.use()

Page 26: (static, express.static),

1 Programming question

Midterm Exam Exercise Solns

1. Read the code below and describe what happens in the terminal when run “node app.js”. Assume the two files are in the same folder.

If no error, list the outputs in the terminal in correct order. If there’s error, describe the error and what cause the error.

person.js

```
class Person {
  constructor(name = 'John') {
    this.name = name;
     }
  getName() {
    console.log(this.name);
     }
}
exports = new Person('Bella');
//return module.exports
```

app.js

```
const person = require('./person');
person.getName();
person.name = 'Jack';
person.getName();
```

Error. In app.js, when we require, the variable person is an empty object{}, then when we call person.getName(), person.getName is undefined, we make a function on undefined, which will give us error.

Based on the code below, when user types http://localhost:3000/users in browser bar, then hit “Enter”, what’s the output in the console/terminal? Assume all other parts of the application are correct except the code listed below.

```
app.get('/users', (req, res, next) => {
  console.log(1);
  next();
});
app.post('/users', (req, res, next) => {
  console.log(2);
  next();
});
app.all((req, res, next) => {
  console.log(3);
  res.status(200).send('Try later');
});
```

1

Programming Solution

```
const express = require('express');
const path = require('path');
const app = express();
app.use(express.urlencoded());


app.listen(8080);
```