# CS 142 Final Examination

Spring Quarter 2016

You have 3 hours (180 minutes) for this examination; the number of points for each question indicates roughly how many minutes you should spend on that question. Make sure you print your name and sign the Honor Code below. During the examination you may consult two double-sided pages of notes; all other sources of information, including laptops, cell phones, etc. are prohibited.

I acknowledge and accept the Stanford University Honor Code. I have neither given nor received aid in answering the questions on this examination.

____________________________________________

(Signature)

____________________________________________

(Print your name, legibly!)

____________________________________________

(SUID - stanford email account for grading database key)

| Problem | #1 | #2 | #3 | #4 | #5 | #6 | #7 | #8 | #9 | |
|---|---|---|---|---|---|---|---|---|---|---|
| Score | | | | | | | | | | |
| Max | 12 | 8 | 8 | 10 | 12 | 12 | 10 | 12 | 10 | |
| | | | | | | | | | | |
| Problem | #10 | #11 | #12 | #13 | #14 | #15 | #16 | #17 | #18 | Total |
| Score | | | | | | | | | | |
| Max | 10 | 10 | 10 | 8 | 10 | 10 | 10 | 6 | 12 | 180 |

## Problem #1 (12 points)

(6 points) HTTP is an acronym for **H**yper**T**ext **T**ransport **P**rotocol and is used by browsers to transfer information encoded in HTML (**H**yper**T**ext **M**arkup **L**anguage). Browsers also can use HTTP to transfer information not encoded in HTML. Describe three different types of information (other than HTML) regularly fetched using HTTP by browsers running modern web applications. (Note: This question is asking about what is transported by the protocol, not what is in the protocol headers.)

(6 points) You observe an web application that receives a HTTP response header from an HTTP GET request that contains the following line:

```
Cache-Control: max-age=2628000
```

1. Describe the benefit a web application would get from having that line in a HTTP GET response header. Give an example of a HTTP GET request done by your photo app that it would be appropriate to apply this line.

2. Describe the disadvantage of having this line on a HTTP GET response. Given an example of a HTTP GET response done by your photo app that it would be inappropriate?

## Problem #2 (8 points)

(4 points) Although our CS142 projects used the XMLHttpRequest DOM interface, we didn't use XML. Instead of XML, what format did we use?

(4 points) When designing a REST API what is the guiding principle in determining the URLs that are exported by the web server (i.e. what principle determines what the URLs look like)?

## Problem #3 (8 points)

(4 points) Is it possible for a single page application to have multiple outstanding XMLHttpRequests? By outstanding requests we mean that a single page has multiple HTTP request sent out to a web server and is waiting for their HTTP responses. Explain your answer.

(4 points) Although it is not a requirement of HTTP, most web applications have backend APIs that make HTTP requests that are independent.  Being independent means the HTTP requests issued by the application don't depend on one another. Describe why making HTTP requests independent is considered a best practice.

## Problem #4 (10 points)

(6 points) Provide a brief description of functioning of the low-level innermost loop of a web server? An answer in pseudocode is acceptable.

(4 points) JavaScript frameworks such as Angular end up using less computing resources on the backend delivering view components than older frameworks such as Ruby on Rails.  Explain the key computation that moved from the backend to frontend in this move to JavaScript frameworks.

## Problem #5 (12 points)

(4 points) Node.js is described as supporting high concurrency in request processing yet only one JavaScript function is ever running at a time. Describe how Node.js can support many requests concurrently yet does not ever technically run multiple requests at the same time. Give an example of multiple requests being handled “concurrently” by the server.

(8 points) What does the following JavaScript code using the Node.js async model output:
Hint: `async.each` calls its "iteratee" function (the 2nd argument) synchronously.

```
console.log('Before async.each');
async.each([10, 1],
   function eachFunc(arg, callback) {
       console.log('eachFunc start', arg);
       setTimeout(function () {
          console.log('eachFunc done', arg);
          callback();
       }, arg);
   },
   function doneFunc(arg) {
          console.log('doneFunc');
   }
);
console.log('After async.each');
```

## Problem #6 (12 points)

(4 points) A file can be read using two different APIs in Node.js. `fs.createReadStream` supports reading a file using the Node.js stream API. `fs.readFile` reads the file and returns it in a Node.js Buffer abstraction. Describe a key advantage of using `fs.createReadStream` that allows it to read files that might not work with the `fs.readFile` interface.

(8 points) Consider the way the listener/emitter pattern is used in Node.js. For each of the following scenarios state if the scenario is part of normally operating program or likely an indication of a programming error. Justify your answer by giving an example of a normal operation or why the scenario would indicate an error.

A. The program installed a listener on an event that is never emitted during a particular run of the program.

B. The program emits an event that never has an listener installed during a particular run of the program.

C. The program installs a listener on an event that can never be emitted by a program.

D. The program emits an event that no corresponding listener is ever installed by the program.

## Problem #7 (10 points)

When defining routes in Express.js we use a call of the form:

```
      app.get(urlPath, callbackFunction);
```

so that the `callbackFunction` will be invoked when an HTTP GET request with a path of `urlPath` comes in.

(6 points) The number of arguments in the definition of the `callbackFunction` varies. Some route definitions have three arguments:

```
      app.get(urlPath,  function (request, response, next) { …
```

while others only use two arguments:

```
      app.get(urlPath,  function (request, response) { …
```

When should the `next` parameter be used by the callback?

(4 points) Would it make sense to have callback with only the request parameter:

```
      app.get(urlPath,  function (request) { …
```

Justify your answer.

## Problem #8 (12 points)

(4 points) Describe the advantages a web application receives by using an index in a database system.

(4 points) State the disadvantages of putting an index on a property in database.

(4 points) What does it mean to do a projection (e.g. select() in Mongoose) of an object in a database?

## Problem #9 (10 points)

(5 points) A relational database system (e.g. MySQL) can work well if the data of a web application can naturally be organized as a collection of ______.

Describe what single word would go in the blank and explain your answer.

(5 points) How did our Photo App benefit from using the Mongoose Object Definition Language instead of directly accessing the MongoDB and using the MongoDB object definition?

## Problem #10 (10 points)

(5 points) Describe how cookies are used by the express session management middleware.

(5 points) Describe a primary use of session state in web applications.

## Problem #11 (10 points)

(6 points) Explain why web apps need to validate user input both in the frontend and in the backend.  Give the reason for having it in both places.

(4 points) You see some JavaScript code that uses a strange convention of calling a method named `then` on the values returned by API calls.  State what you think is going on here.

## Problem #12 (10 points)

(6 points) It is easy to have the Express.js session management middleware store session state in MongoDB rather than in the default memory store we used in the assignment code.

A. What advantage would our photo app backend have if we do this?

B. Explain why a web app might not want to do this.

(4 points) Explain why the HTML5 Web Storage API (`sessionStorage` and `localStorage`) are not useful for storing the user's data of a web application. Look for an explanation that is fundamental rather than limits of some current browser implementations.

## Problem #13 (8 points)

(3 points) What does the Same-Origin Policy in the browser do for our web application?

(5 points) Explain how a security flaw in our web applications can help an attacker gain access to systems other than our web app.

## Problem #14 (10 points)

(3 points) Describe the role of a Certificate authority in HTTPS.  What is the key piece of knowledge that a browser gets from a trustworthy certificate authority.

(4 points) What good does it do to encrypt something with your private key?

(3 points) When using HTTPS, the browser generates the key used to encrypt the connection to the web server.  How does the browser ensure a man-in-the-middle attacker cannot view the key?

## Problem #15 (10 points)

(4 points) If an attacker could get access to the browser running your web application, describe how the attacker could do a Session Hijacking Attack.

(6 points) Which of the below properties can be obtained using Message Authentication Codes (MACs). Explain your answer.

1) Authentication
2) Integrity
3) Confidentiality

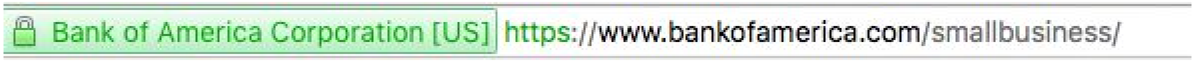
## Problem #16 (10 points)

(6 points) An attacker could perform a code injection attack on both the browser and the web server. Give an example of an attack in each place.

(4 points) Describe why it is not safe to take model data and assign it to innerHTML.

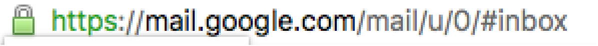
## Problem #17 (6 points)

What does it mean to the user of a web applications if the browser shows an extended certificate such as:



Bank of America Corporation [US] https://www.bankofamerica.com/smallbusiness/

rather than a regular certificate:



https://mail.google.com/mail/u/0/#inbox

## Problem #18 (12 points)

(6 points) Scale out architectures require some mechanism to distribute the request across the multiple instances. Describe the mechanism used in scale out architectures for:

A. Web servers

B. Database servers

(6 points) Explain how a content distribution network (CDN) can improve the performance of your web applications for users in locations distant from your app's web servers. What gets faster?