CSC309 Winter 2016
Lecture 11

Larry Zhang

Announcements

- Showcase order posted on Piazza
- Your game should be done by the time of showcase, the three days before the final submission is for code organization, bug fixes, etc.
- This week's tutorial is: A3 problem solving
- If you got very low mark for A2, show the working version to your TA in this week's tutorial.
- This week's tutorial is the last one.

TA’s feedback’s about submission, may apply to A3

- must include a live version URL in their readme file
- must include a schema.sql script in their submission
- it would be a good idea to double-check their MarkUs submission by unzipping it and setting up everything, including the database tables, from scratch (i.e. go through the steps that the TA would take when marking it)
- use a relative path for the session directory when calling session_save_path

Showcase Location: **DH-2010**

Tuesday March 29, 3pm

Final Exam Review

what we have learned
what questions will be on the exam

Types of Questions

- Conceptual questions: things you should remember after taking this course
- Reading code: given a piece of code, analyze it, find bugs
- Writing code: given some requirements, write a small piece of code, or complete a partially completed piece of code

The Topics

HTTP

- What is the work flow of HTTP request and response
- What is the format of HTTP request / response
- What is URL, what is domain name, what is IP, what is DNS
- HTTP/1.0 and HTTP/1.1 what’s the difference

HTML

- Basically, you are supposed to know how to write HTML
- Some basic concepts
  - difference between div and span
  - invalid nesting of elements
  - etc

CSS

- In general, you need to be able to write CSS code
- How selectors work
  - how to group selectors, how select children, sibling, etc.
  - how to select id, class…
  - how to select hover…
  - play that selector game

CSS cont.

- How does cascading work in CSS
- color, background-color, text-align, etc
- The Box Model: padding, margin, border
- Float, Position
- CSS media query
- Know very well how these work
- Know very well what you did in A1

CSS: Exam Questions

- Given a piece of HTML, write a selector to select certain elements
- Given a selector, point out which elements are selected
- Requiring a style, write the CSS for it
- Given CSS, analyze what style is displayed
- Write / Read position / float CSS
- etc

JavaScript / JQuery

- DOM
- Find an element, getElementById, etc.
- How to decouple HTML, CSS, JavaScript
- Basic Syntax: Variables, Functions, Operators
  - the really basic things you should know without reference sheet

AJAX

- The asynchronous communication model
- How to use JQuery to implement AJAX calls
- Not required: raw AJAX implementation
- JSON

JavaScript / JQuery / AJAX Exam Questions

- Given a requirement, write a piece of JavaScript
- Read JavaScript and analyze what it does
- Given an ugly piece of code, redo it in better way
- Fixing bugs
- You may be asked to write in raw JavaScript / JQuery, you need to know both

Example: what’s popped up?

```
<head>
<script>
    var x = document.getElementById(“ID”).innerHTML;
    alert(x);
</script>
<body>
    <div id=“ID”>HAHA</div>
</body>
</head>
```

PHP

- Server-side programming basics: basic concepts
- Basic Syntax
  - Arrays
- Parsing query
- Sessions
- All the examples we did

PHP: Exam Questions

- Read code
- Write code
- Code with simple PostgreSQL usage
- Based on all the demos we have done, tutorials and assignments

MVC, REST, OAuth

- Mostly conceptual things
- Be able to identify what is model, view, and control
- Be able to draw the finite state machine
- Know what is REST, what is not REST
- Understand how OAuth works

OO JavaScript

- Know how to implement class, methods, member variables in JavaScript

Canvas

- Given a required shape, describe how to draw it
- Write JavaScript code to create certain shape or animation
- Read a piece of code and describe what it is drawing

Mobile API, Touch

- Just basic concept, not much can be asked here

Node.js

- Understand it’s model of computation
  - non-blocking I/O
  - Event Loop
  - single thread
  - difference from other models

Node.js cont.

- Write server side code in JavaScript
- In general, you need to be able to implement the same functionality using different approaches
  - PHP
  - JavaScript / JQuery
  - Node.js
- Like what we did with the guess number game

WebSocket

- Know how to code on client side and server side, similar to what we did in class.
- The event driven programming model

Performance and Scalability

- The concepts that we mentioned in the lecture.

Summary

- Be familiar with what we talked about in the lecture
- Be familiar with the demos we did in class
- Be familiar with the tutorial exercises.
- Be familiar with what you did in the assignments
- Past exam can be found in UofT library's old exam repository. Arnold's past exams are more relevant.

Time and Location

- April 18th, 9am ~ 12pm
- IB-120
- Aid allowed: 1 page of double-sided Letter (8-1/2 x 11) sheet
- Necessary reference information will be provided within each question.

Showcase in DH-2010!