

# CSE 190 M, Spring 2008

## Final Exam Key

### 1. JavaScript/DOM (3 solutions shown)

```
window.onload = function() {
  $("eat").onclick = eatClick;
};

function eatClick() {
  var name = $("foodname").value.toLowerCase();
  var group = $("foodgroup").value;
  var elements = $$("img.food");
  for (var i = 0; i < elements.length; i++) {
    if (elements[i].hasClassName(group)
        && elements[i].alt.toLowerCase() == name) {
      elements[i].remove();
    }
  }
}
```

```
window.onload = function() {
  $("eat").onclick = eatClick;
};

function eatClick() {
  var elements = document.getElementsByTagName("img");
  for (var i = 0; i < elements.length; i++) {
    if (elements[i].hasClassName("food")
        && elements[i].hasClassName($("foodgroup").value)
        && elements[i].alt.toLowerCase() == $("foodname").value.toLowerCase()) {
      elements[i].remove();
    }
  }
}
```

```
window.onload = function() {
  $("eat").onclick = function() {
    $$("img.food." + $("foodgroup").value + "[alt=" + $("foodname").value.toLowerCase() +
      function(item) { item.remove(); });
  }
}
```

## 2. Ajax/XML (2 solutions shown)

```
window.onload = function() {
  new Ajax.Request("personality.php",
    {
      method: "get",
      onSuccess: ajaxSuccess
    }
  );
};

function ajaxSuccess(ajax) {
  var persons = ajax.responseXML.getElementsByTagName("person");
  for (var i = 0; i < persons.length; i++) {
    var name = persons[i].getAttribute("name");
    var type = persons[i].getAttribute("type");
    var answers = persons[i].getElementsByTagName("answers")[0].firstChild.nodeValue.toLo
    var b = 0;
    for (var j = 0; j < answers.length; j++) {
      if (answers[j] == "b") {
        b++;
      }
    }

    var li = document.createElement("li");
    li.innerHTML = name + ": " + type + " (" + b + " Bs)";
    $("output").appendChild(li);
  }
}
```

```
window.onload = function() {
  new Ajax.Request("personality.php", {
    method: "get",
    onSuccess: function(ajax) {
      var persons = ajax.responseXML.getElementsByTagName("person");
      for (var i = 0; i < persons.length; i++) {
        var li = document.createElement("li");
        li.innerHTML = persons[i].getAttribute("name") + ": " + persons[i].getAttribute
          " (" + persons[i].firstChild.nodeValue.replace(/^[^bB]+/g, "").length +
        $("output").appendChild(li);
      }
    }
  });
}
```

### 3. PHP (4 solutions shown)

```
<?php
header("Content-type: text/plain");
if (isset($_REQUEST["name"])) {
    $text = file_get_contents("names.txt");
    $lines = explode("\n", $text);
    foreach ($lines as $line) {
        $tokens = preg_split("/[ ]+/", $line);
        if (strtolower($tokens[0]) == strtolower($_REQUEST["name"])) {
            $min = $tokens[1];
            for ($i = 2; $i < count($tokens); $i++) {
                if ($tokens[$i] < $min) {
                    $min = $tokens[$i];
                }
            }
            print $min;
        }
    }
}

if (!isset($min)) {
    print -1;
}
?>
```

```
<?php
header("Content-type: text/plain");
if (isset($_REQUEST["name"])) {
    $lines = explode("\n", file_get_contents("names.txt"));
    foreach ($lines as $line) {
        $tokens = preg_split("/[ ]+/", $line);
        $name = array_shift($tokens);
        if (strtolower($name) == strtolower($_REQUEST["name"])) {
            sort($tokens);
            die($tokens[0]);
        }
    }
} else {
    print -1;
}
?>
```

```
<?php
header("Content-type: text/plain");
if (isset($_REQUEST["name"])) {
    foreach (file("names.txt") as $line) {
        $tokens = preg_split("/\s+/", trim($line));
        if (!strcasecmp(array_shift($tokens), $_REQUEST["name"])) {
            die(min($tokens));
        }
    }
}
print -1;
?>
```

```
<?php header("Content-type: text/plain");
if (isset($_REQUEST["name"]) &&
    preg_match("/\n{$_REQUEST['name']}([\d ]+\s+)+/i",
        file_get_contents("names.txt"), $matches))
    die(min(preg_split("/\s+/", trim($matches[1]))));
print -1; ?>
```

## 4. SQL (2 solutions shown)

```
SELECT DISTINCT t.name, c.name
FROM   teachers t
JOIN   courses c ON c.teacher_id = t.id
JOIN   grades g1 ON g1.course_id = c.id
JOIN   grades g2 ON g2.course_id = c.id
WHERE  g1.student_id <> g2.student_id
AND    g1.grade <= 'C-' AND g2.grade <= 'C-';
```

```
SELECT DISTINCT t.name, c.name
FROM   teachers t
JOIN   courses c ON c.teacher_id = t.id
JOIN   grades g1 ON g1.course_id = c.id
JOIN   students s1 ON s1.id = g1.student_id
JOIN   grades g2 ON g2.course_id = c.id
JOIN   students s2 ON s2.id = g2.student_id
WHERE  s1.name < s2.name
AND    g1.grade <= 'C-' AND g2.grade <= 'C-';
```