

NATIONAL UNIVERSITY OF SINGAPORE**CP3101B – WEB PROGRAMMING AND APPLICATIONS**

(Semester 2: AY2014/15)

Time Allowed : 2 Hours

INSTRUCTIONS TO STUDENTS

1. Please write your Student Number only. Do not write your name.
2. This assessment paper contains **4** questions and comprises **15** printed pages.
3. Students are required to answer **ALL** questions.
4. Students should write the answers for each question in the answer box provided. Do not be long winded.
5. This is an OPEN BOOK assessment.
6. You can use pencil.
7. All the best.

MATRICULATION NO: _____

This portion is for examiner's use only

Question	Maximum Marks	Marks
Q1	30	
Q2	36	
Q3	22	
Q4	12	
Total	100	

Background for Q1 and Q2: In another SoC module that is very competitive: CS3233 – Competitive Programming, Dr Steven Halim needs to maintain a publicly accessible ranklist that is updated at least once a week in order to push his students to the extreme limit of competition. The *simplified version* of this public ranklist for the purpose of this CP3101B final exam is shown in the screenshots below.

The ranklist table has 5 columns. The first column (**Rank**) contains ranking from 1 to N , where N is the class size ($N=22$ in S2 AY14/15). The second column (**Student**) contains students' initials. The third (**Speed**) and fourth (**Diligence**) columns contain students' speed and diligence component scores. The fifth column (**Total**) contains the sum of each student's speed and diligence component scores.

Upon page load, the rows in the table are initially sorted based on decreasing scores in the fifth column (see **Figure 1**); breaking ties using lexicographical order of the second column (see Rank 1 vs Rank 2). However, user can interactively sort the rows based on speed (see **Figure 2**), diligence (see **Figure 3**), or back to default: based on total, by clicking the appropriate table head element `<th>`.

The layout is responsive with the Chart sidebar hidden when the page is loaded on small screen (see **Figure 4**). On large screen, the table takes 2/3 of the available width of its container and the chart takes the other 1/3. Note: The visualization content of the Chart sidebar is *not asked* in this final exam.

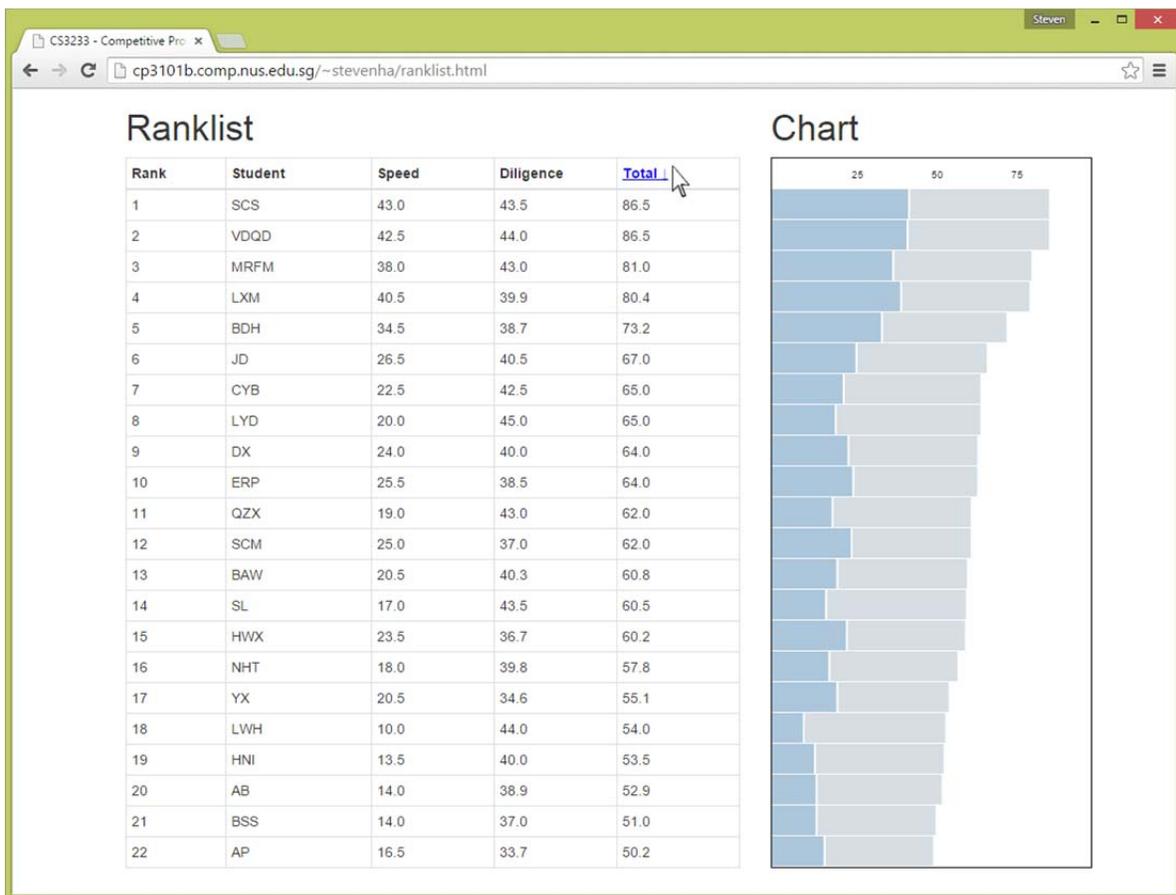


Figure 1. The client-side screen shot on large screen (Bootstrap `md` or `lg`).

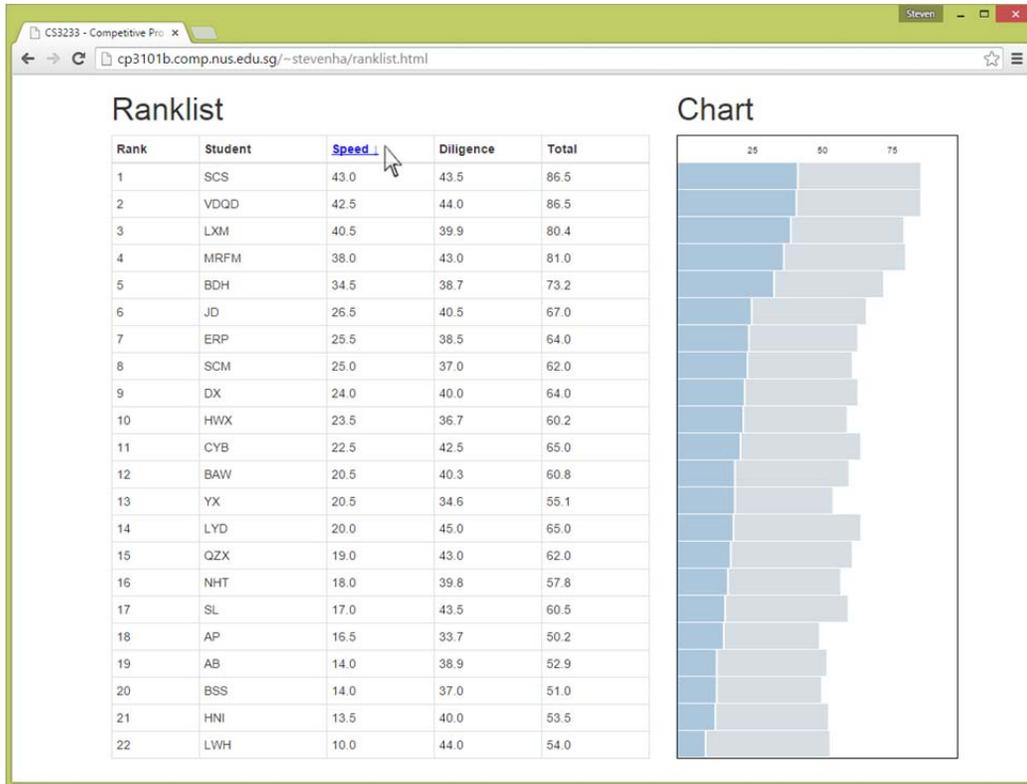


Figure 2. Clicking the table header “Speed” sorts the rows based on descending speed scores

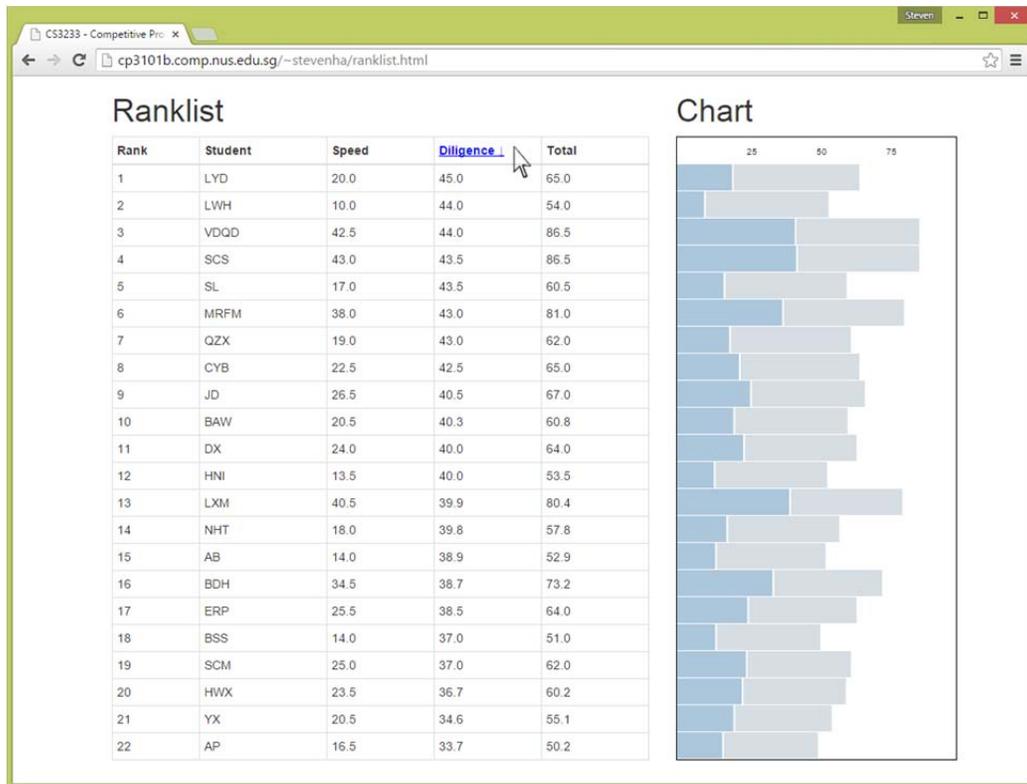


Figure 3. Clicking the table header “Diligence” sorts the rows based on descending diligence scores

The figure shows three browser windows displaying a 'Ranklist' table. Each window shows a different sorting order for the 'Total' column. The first window is sorted by 'Speed', the second by 'Diligence', and the third by 'Total'. The table has 22 rows and 5 columns: Rank, Student, Speed, Diligence, and Total.

Rank	Student	Speed	Diligence	Total
1	SCS	43.0	43.5	86.5
2	VDQD	42.5	44.0	86.5
3	LXM	40.5	39.9	80.4
4	MRFM	38.0	43.0	81.0
5	BDH	34.5	38.7	73.2
6	JD	26.5	40.5	67.0
7	ERP	25.5	38.5	64.0
8	SCM	25.0	37.0	62.0
9	DX	24.0	40.0	64.0
10	HWX	23.5	36.7	60.2
11	CYB	22.5	42.5	65.0
12	BAW	20.5	40.3	60.8
13	YX	20.5	34.6	55.1
14	LYD	20.0	45.0	65.0
15	QZX	19.0	43.0	62.0
16	NHT	18.0	39.8	57.8
17	SL	17.0	43.5	60.5
18	AP	16.5	33.7	50.2
19	AB	14.0	38.9	52.9
20	BSS	14.0	37.0	51.0
21	HNI	13.5	40.0	53.5
22	LWH	10.0	44.0	54.0

Figure 4. Views on smaller screen (Bootstrap sm or xs)

Q1 (30 marks). Front-end

Please complete the given skeleton files: **ranklist.html**, **ranklist.css**, and **ranklist.js** in order to obtain webpages that look like **Figure 1, 2, 3, and 4** above. Use client-side technologies that you have learned in CP3101B (only HTML5, CSS3, JS, jQuery, and Bootstrap answers are accepted).

a. ranklist.html (10x1 = 10 marks, each blank line worth 1 mark)

```

<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset="utf-8">
<title>CS3233 - Competitive Programming</title>
<!-- jQuery -->
<script src="http://code.jquery.com/jquery-2.1.3.min.js"></script>
<!-- Bootstrap -->
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.2/css/bootstrap.min.css">
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.2/css/bootstrap-theme.min.css">
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.2/js/bootstrap.min.js">
</script>
<!-- home-made -->
<link rel="stylesheet" href="ranklist.css"> <!-- see Q1.b-->
<script src="ranklist.js"></script> <!-- see Q1.c-->
</head>

```

```

<body>
<div class=" " >
  <div class=" " >
    <div class=" " >
      <h1> </h1>
      <table border="1"="table table-bordered table-condensed">
        <thead>
          <tr>
            <th>Rank</th>
            <th> </th>
            <th id="speed" class="sortable">Speed</th>
            <th id="diligence" class="sortable">Diligence</th>
            <th id="total" class="sortable">Total &downarrow;</th>
          </tr>
        </thead>
        <tbody id="dynamictablehere">
        </tbody>
      </table>
    </div>
    <div class=" " hidden-xs hidden-sm">
      <h1>Chart</h1>
      <canvas id="chart" width="320px" height="713px">
        Your browser does not support HTML5 canvas...
      </canvas>
    </div>
  </div>
</div>
</body>
</html>

```

b. ranklist.css (3x2 = 6 marks, each CSS rule worth 2 marks)

```

/* Write these 3 CSS3 rules in the given space below */
/* 1. Put a 1 pixel solid and black border around HTML5 <canvas> element with id="chart" */

/* 2. If mouse pointer is over a <th> element with class "sortable",
change the text color inside this <th> element to blue and add underline */

/* 3. If mouse pointer is over <tr> element that is a descendant of a <tbody> element,
change the background color of this <tr> element to HotPink */

```

c. ranklist.js (14 marks)

This JavaScript question relies on the back-end that is designed in Q2.

Please attempt Q2 first before returning to answer Q1.c.

```
var roster; // this will be populated via call to ranklist.php
var sort_key = 2; // this is the default sort order (descending total)

$(function() { // wait until the DOM is ready
  loadData();
});

function displayTable() { // 4 marks
  // write a code to generate HTML table row (tr) elements based on
  // the content of global variable "roster" and assign it to the
  // tbody element with id "dynamictablehere" in ranklist.html
  //
  // for example, the first tr should be: <tr id='SCS'><td>1</td>
  //   <td>SCS</td><td>43.0</td><td>43.5</td><td>86.5</td></tr>
  // if the sort_key = 2
  // your job is to do the same for the other 21 tr elements

}

function drawChart() {
  // you can assume that this function, if called, will draw the content
  // of variable "roster" as seen in Figure 1, 2, 3 in the canvas chart.
  ... details omitted as it is not important for this final exam...
}
```

```
function registerEventHandlers() { // 5 marks
    // add event handler to allow user to change the sort key interactively
    // by clicking table header (<th>) element that has class 'sortable'
    // add small visual cue: a &downarrow; symbol for the selected sort_key
    // Please scrutinize Figure 1, 2, 3, 4 again for this question

}

function loadData() { // 5 marks
    // write an AJAX call to execute "ranklist.php" with appropriate sort_key
    // to receive a JSON object that contains the roster data
    // that is sorted according to the requirements

}
```

Q2 (36 marks). Back-end**a. Create a table in database (3 marks)**

Suppose the following database table has been designed to support the front-end built in Q1 above.

```
mysql> DESCRIBE roster;
```

```
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null  | Key  | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| name       | varchar(30)   | NO    | PRI  | NULL    |      |
| speed      | float         | YES   |      | NULL    |      |
| diligence  | float         | YES   |      | NULL    |      |
| total      | float         | YES   |      | NULL    |      |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Please write (My)SQL command(s) that was used to create a table according to the description above:

b. Populate the table with some rows (3 marks)

Suppose the content of the table 'roster' is as follows (notice that total = speed + diligence values):

```
mysql> SELECT * FROM roster;
```

```
+-----+-----+-----+-----+
| name | speed | diligence | total |
+-----+-----+-----+-----+
| AB   | 14    | 38.9     | 52.9 |
...19 other lines are not shown for brevity...
| VDQD | 42.5  | 44       | 86.5 |
| YX   | 20.5  | 34.6     | 55.1 |
+-----+-----+-----+-----+
22 rows in set (0.00 sec)
```

Please write (MySQL) command(s) that was used to insert the 3 visible rows of data shown above (ignore the other 19 rows) into the initially empty table 'roster':

c. Back-end PHP (10 marks)

Now please write **ranklist.php** that:

1. Opens connection to MySQL database (assume that this has been done by '**connect.php**' that has been discussed in class and you can simply use variable **\$db** to access the **mysqli object**).
2. Accepts one parameter **sort_key** that can have value 0, 1, or 2 which means sort by speed/diligence/total column, respectively. In case a value other than 0, 1 or 2 is entered, the script will revert to the default sort key, which is to sort by total.
3. Retrieves the rows of the database table 'roster' (that have been described in Q2.a and populated in Q2.b above) and have the rows sorted according to the **sort_key** parameter.
4. Sends the entire content of the database table as a JSON string to the caller.

The JSON object, when parsed (by client), will produce the following JavaScript array of objects:

```
var roster = [ // there are 22 students, i.e. roster.length = 22
  {name:"AB", speed:14, diligence:38.9, total:52.9},
  ...19 other lines are not shown for brevity...
  {name:"VDQD", speed:42.5, diligence:44, total:86.5},
  {name:"YX", speed:20.5, diligence:34.6, total:55.1},
];
```

5. Closes the connection to MySQL database

Use server-side technologies that you have learned in CP3101B to answer this question!
(only PHP and MySQL code are accepted).

```
<?php  
require_once 'connect.php'; // now you can use variable $db below
```

```
?>
```

d. update.php (5 marks)

Assume that a script **update.php** has been written that accepts three parameters: (name, newspeed, newdiligence), opens the database, checks if name exists in table 'roster', and if yes, updates the table accordingly (or ignore the update command otherwise). For example, executing:

update.php?name=VDQD&newspeed=46.5&newdiligence=48

will update one row in the table 'roster' into (notice the changes in total column too):

VDQD	46.5	48	94.5
------	------	----	------

Please complete the following PHP code:

```
<?php
require_once 'connect.php'; // now you can use variable $db below

$name = $db->escape_string($_REQUEST['name']);
$newspeed = $_REQUEST['newspeed'];
$newdiligence = $db->escape_string(_____);

$query = "_____";
_____";
$res = $db->query($query);
if (!$res)
    exit("MySQL error");
?>
```

e. Database design concerns (5 marks)

Is there any design concern with the database table shown in Q2.a above?

If there is/are concern(s), what should we do to improve the design?

Hint: Consider the total column?

f. Security concerns (5 marks)

Elaborate what can possibly go wrong with the answer for question Q2.d above and provide suggestions on how to make it more secure.

One hint (there are others): A1 and A2.

g. Performance concerns (5 marks)

Elaborate what can possibly be performance concern(s) with the simple web application that we build in Q1+Q2. Provide suggestions on how to make it faster and/or more scalable.

Q3 (22 marks). Miscellaneous topics in Web Development

1. (5 marks). Someone from the marketing division of your company ('UWishWCode' from CP3101B lab assignments) suggests the following URL¹ for a new web application that your company has just recently developed: http://www.uwishwcode.com/awesomeapp/launchapp.php?app_id=007.

What is your response to such suggestion?

2. (5 marks). You built a website about a special project event in School of Computing approximately one year ago. You have put Google Analytics tracker code in all your web application files. Within this one year, the number of average sessions/day steadily declining from about a few hundreds visitors/day one year ago to a mere 1-digit visitors/day today.

What can you conclude?

¹ At this point of writing (06 April 2015, the domain name *uwishwcode.com* is still available).

3. (5 marks). You believe that the new web application that you build is “no 1 in the world”. List down five Search Engine Optimization (SEO) techniques that you will use in your application to improve its chance to be listed as top 10 in search results of relevant keywords in major search engines.

1.
2.
3.
4.
5.

4. (7 marks). <http://nusmods.com> is an example of a Single-Page Application (SPA). Mention at least two benefits and two challenges of using SPA technique in your web application.

Benefits:
1.
2.
Challenges:
1.
2.

Q4 (12 marks). Course Project Website Maintenance

Please state what is your team’s consensus of the current state/weaknesses of your team’s course project website as of Wednesday, 29 April 2015? What will your team will do to maintain/fix bugs/add feature to your team’s course project website by Wednesday, 06 May 2015, 23.59?

This question will be graded on Thursday, 07 May 2015 based on the *consistencies of the answers from team members* and the actual updates done to your team website by the stipulated deadline.

The current state/weaknesses of our team’s course project website is/are:

Our team’s plan to maintain the website is/are: