IS 445/545 Exam
(Total 300 points)

1. The Setup

Please read https://github.com/csulbying/web-app-dev/blob/master/01-setup/setup.md for detail instructions to setup your development tools, create a Github repository, and how to deploy a git repository to Netlify (https://app.netlify.com/).

2. The Submission

The detail web site requirements are given in the next section. Once you complete it, please create a text file (a file with ".txt" postfix) that has two links, one to your github repository and one to your netlify website. Please submit the text file to the beachboard final exam dropbox.

Wrong file format, invalid URL, or un-working application is completed and gets 0 point. For example, below is the sample content of your hw.txt (the links are sample links, please change them to your github repository and your netlify site)

https://github.com/your-github-name/your-repository.git
https://user-specific-part.netlify.com/

The deployment in Netlify will be counted as 10% (30 points).

3. Grading by GitHub Commits

Please complete it step by step, having a functional program is more important than more features that don't work. It also easy to debug.

Each step must be in a separate GitHub commits with commit messages such as "step 1…" , "step 2…". You can have more commits but each step should have at least one commit. The grading will be based on the commits. If there is only a single commit, the total grade cannot be more than 50% of total grade.

Please don't commit any change after due time. Commit after due time will fail the exam with at least a 50% penalty of 150 points.

4. Tasks

Three are three subtasks that you should code in three steps. It doesn't include deployment. The deployment to Netlify will be counted as 10% (30 points), you should deploy to Netlify when you create the initial simple page.

Step 1 of 3: Basic layout and styles (130 points)

All GitHub commit messages in this step should be prefix with "step 1", for example: "step 1: init page", "step 1: add buttons", ... At least one commit in this step.

When open your html file, it should display the following UI:
It's styled with bootstrap 4. The title block has a color a "bg-info" background. Text color are "text-danger" and "text-light". Texts are centers.

Below the title block are two labelled input fields. Both label and input has a color of "text-success".

Then a submit button with a style of "btn-primary-warning".

Below the submit button is the result area.

Final Exam

Find all even numbers between two numbers

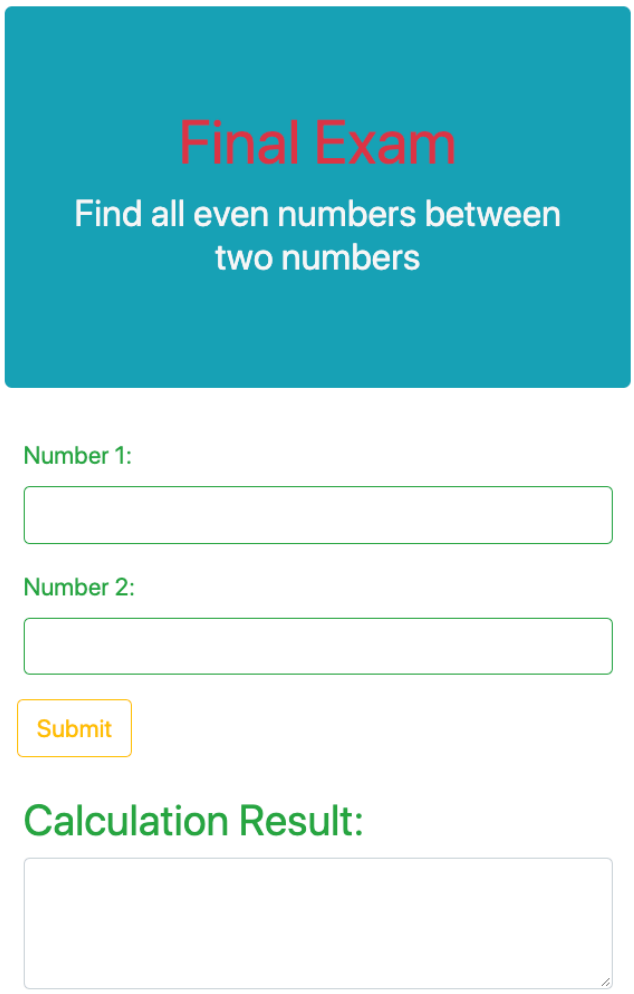
Number 1:

Number 2:

Submit

Calculation Result:

The page should be responsive (weight 30 out of 130 points), if the screen is small than medium, it displays as the following:



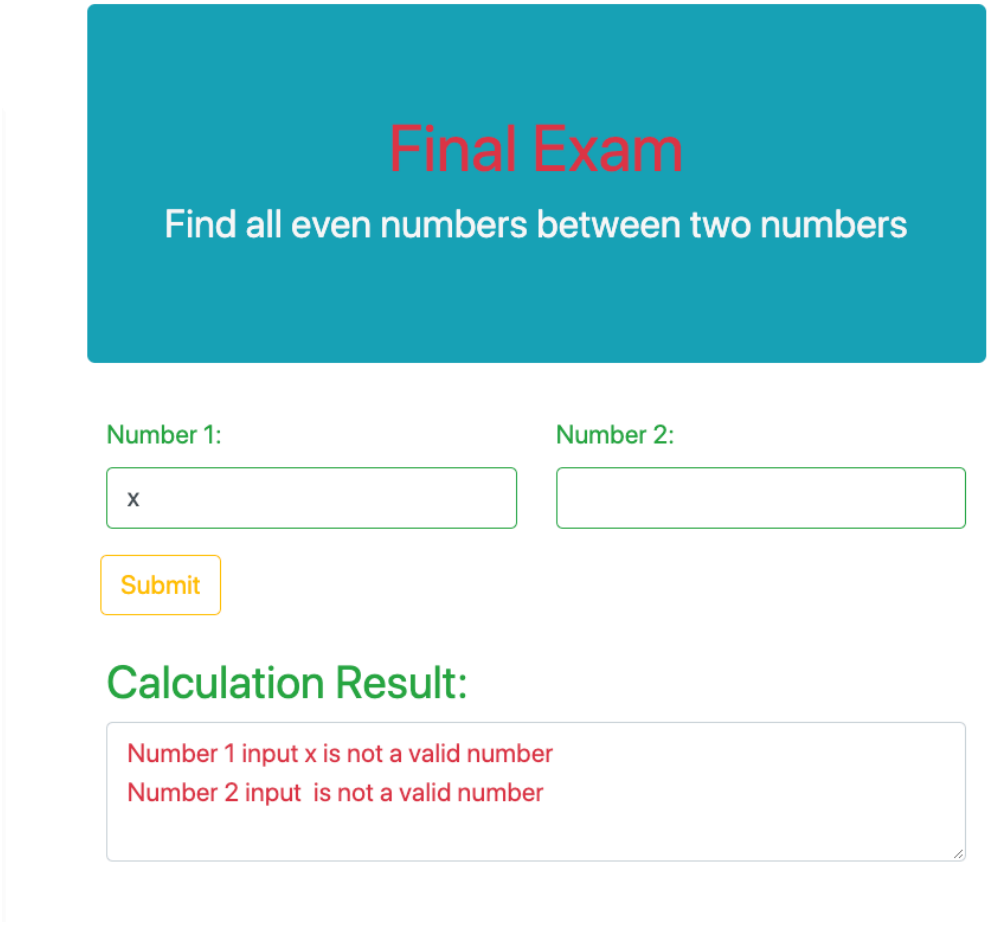
Step 2 of 3: Valid inputs (90)

All GitHub commit messages in this step should be prefix with “step 2”, for example: “step 2: valid number 1”, “step 2: complete valid function”, ... At least one commit in this step.

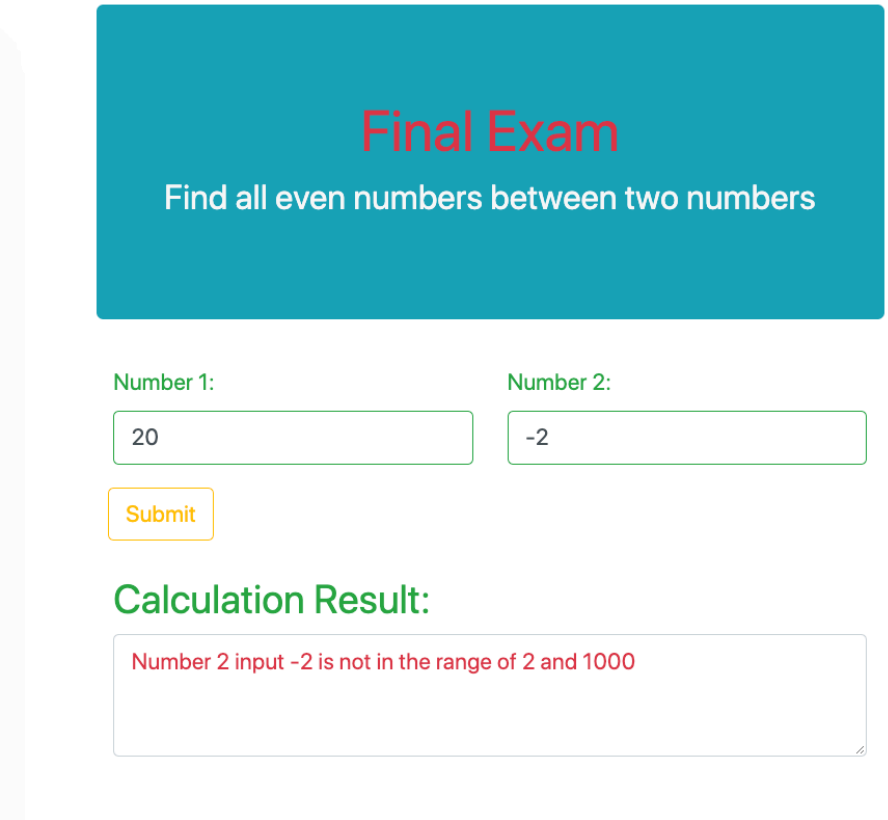
User should be allowed to input two numbers in the range of 2 and 100 (including 2 and 100). The error message should show which input is wrong (-20 points if not specific).

There are at least three possible error cases:

Error case 1: The input is not a valid number. As show in the following diagram, Number 1 is “x” and Number 2 is empty. Tip: you can use “parseInt(val, 10)” to find out, an invalid number returns a “NaN” that can be checked by using “isNaN” function.



Error case 2: input number are out of range. The following input -2 is an error.



Error case 3: input 1 is out of range and input 2 is not a valid number.

Final Exam

Find all even numbers between two numbers

Number 1:

x

Number 2:

200

Submit

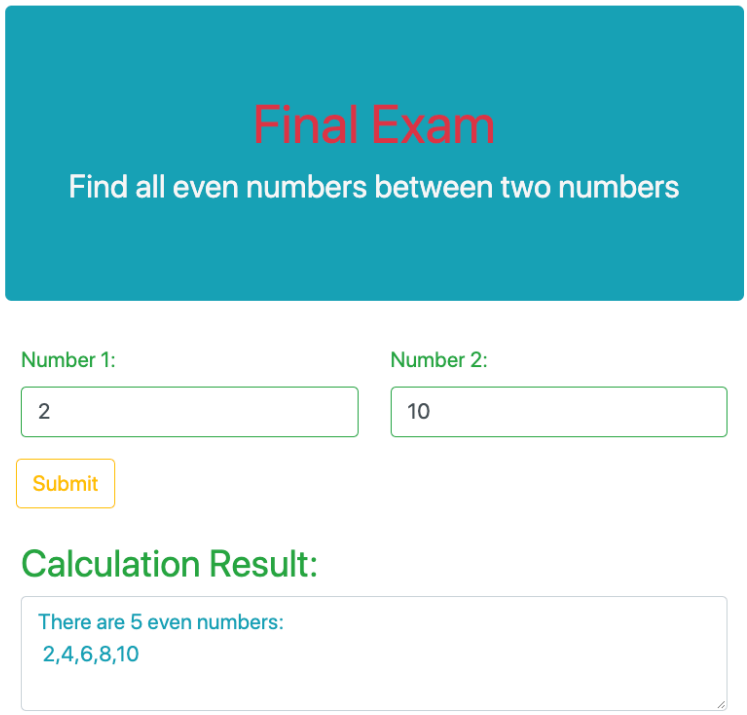
Calculation Result:

Number 1 input x is not a valid number
Number 2 input 200 is not in the range of 2 and 1000

Step 3 of 3: find all even numbers and print the result (50 points)

All GitHub commit messages in this step should be prefix with “step 3”, for example: “step 3: add find even number”, “step 3: fix finding numbers”, ... At least one commit in this step.

Find all even numbers between the two inputs (including the inputs) and report the count (using “text-info” color) and list of all prime numbers (using “text-success” color). The inputs can be in any order, i.e., number 1 could be bigger or smaller than number 2. Following are three examples for input pairs of {2, 10}, {20, 20}, and {90, 77}. In the third case, the first number of 90 is bigger than the second number of 77.



Final Exam

Find all even numbers between two numbers

Number 1:

19

Number 2:

19

Submit

Calculation Result:

There are 0 even numbers:

Final Exam

Find all even numbers between two numbers

Number 1:

90

Number 2:

77

Submit

Calculation Result:

There are 7 even numbers:
78,80,82,84,86,88,90