

Names \_\_\_\_\_

**Instructions**

1. Collaboration is highly encouraged. Though you may work on this assignment alone, you are encouraged to complete this assignment with a single partner according to the following guidelines:
  - You must work on *all* problems *together*. You must both put in an equivalent effort.
  - You are encouraged to discuss (at a high level) any problems or issues with other individuals or pairs, but you may not directly share code with them.
  - To foster true collaboration, you should use one computer: the person typing is the “driver” while the other person is the “navigator” and should direct the driver. Trade off these roles every 15 minutes or at least every other exercise.
2. Hand in all your source code files through webhandin
  - For convenience, place each program into its own folder with appropriate file name(s)
  - Place all of your files into one folder/directory and zip it up. Turn in the ZIP archive file
3. Be sure to rigorously test your programs with sufficient input examples and test cases.
4. Be sure to thoroughly read this rubric and understand how we will evaluate your program.
5. Printout and hand in a hardcopy of this rubric (first page is sufficient) with you name(s) on it.

Question	Points	Score
1	25	
2	25	
3	25	
Total:	75	

## Programs

In this assignment, you will adapt all of your JavaScript programs from the previous assignment to work in an HTML/CSS webpage. Specifically, instead of prompting for user input using the JavaScript functions `prompt` or `alert` you will create simple webpages with web forms that will serve as inputs instead. Each webpage will have a button that executes a JavaScript function that retrieves the input from the web form, computes the result, and then displays the result to the user.

The details of how your webpage looks (style) and operates are left for you to design. However, you are *highly encouraged* to:

- Use Bootstrap as a base for styling
  - Use jQuery to retrieve and set values in your web form
1. (25 points) Create a webpage interface for the gas cost program.
  2. (25 points) Create a webpage interface the loan cost program.
  3. (25 points) Create a webpage interface for the cell minute usage app.

## Rubric

When we grade your assignment, we will be assessing it based on the items below.

### Following Instructions

- All required soft-copy files handed in via webhandin
- Correct file name(s) and organization
- Any required hardcopies (this rubric) handed in
- Programs successfully execute and there are no formatting or syntax errors

### Style

- Appropriate variable and function/method identifiers
- Style and naming conventions are consistent
- Good use of whitespace; proper indentation
- Clean, readable code
- Code is well-organized

## **Documentation**

- Well written comments that clearly explain the purpose of each non-trivial piece of code
- Comments explain the "what" and "why"
- Comments are not overly verbose or overly terse
- Code itself is "self-documenting"; explains the "how"

## **Program Design**

- Code is well-organized and efficient
- Code is modular; substantial pieces of it could be reused; few redundancies
- Code is easily understood and maintainable
- It is clear that sufficient testing has been performed
- Corner cases and bad input have been anticipated and handled appropriately

## **Program Correctness**

- Source code compiles, executes as expected
- Program runs as specified: correctly reads any input; correctly formatted output
- Test cases successfully execute