

## EECS 448 - Lab 09

In your language of choice, implement a calculator that uses the Model-View-Controller pattern. It needs to be able to do the following:

1. Allow the user to select between
  - Addition
  - Subtraction
  - Multiplication
  - Division
  - Exit
2. Once the user selects one of the operations,
  - (a) Ask the user if they would like to use the previous result for the first number, second number, or neither.
  - (b) Allow the user to enter one or two numbers (depending on their response to previous question).
3. Perform the operation the user requested.
4. Ask the user if they would like to perform another operation.
  - (a) Quit if no.
  - (b) Loop back if yes.

### Notes:

- The model class should only ever process data. It shouldn't print anything to console. It shouldn't call functions in other classes.
- The view class should only ever print stuff and ask the user for input. It shouldn't ever process data. It can perform data validation on menu choices. If you need to check if, for instance, a number is valid, offload that on the model by way of the controller.
- The controller class shouldn't ever process data or print stuff to the user. Use the view and model to do that. You can deal with data validation here but rely on the model to actually check if the data is valid. For data validation, only use the controller to deal with the result of the check if the data is valid (eg your loops will be here).
- You may work in teams of not more than 2 people.

**Deliverables:** In a pdf document,

- Place your source code or a link to your source code (eg on github or something similar). Make sure that however you decide to include your source code, it has syntax highlighting.
- Include a class diagram. Use it to illustrate how your software actually does follow the MVC pattern.
- Include a communication diagram of your software performing one of the four operations. In the communication diagram, illustrate how you rely on the view exclusively for user interaction and the model exclusively for data processing.
- Include screenshots of your program performing each of the operations. For + and /, enter both numbers. For \*, use a previous result for the first number. For -, use a previous result for the second number.
- If you worked in a team, briefly mention who worked on which parts of the project.