Announcements

- Check class announcements daily
- You must implement programming projects by yourself
Model-View-Controller (MVC)

- Model for GUI programming (Xerox PARC ’78)
- Separates GUI into 3 components
  1. Model ⇒ application data
  2. View ⇒ visual interface
  3. Controller ⇒ user interaction
MVC Model of GUI Design

- **Model**
  - Should perform actual work
  - Should be independent of the GUI
    - But can provide access methods

- **Controller**
  - Lets user *control* what work the program is doing
  - Design of controller depends on model

- **View**
  - Lets user see what the program is doing
  - Should not display what controller *thinks* is happening (base display on model, not controller)
Model View-Controller

- How should we design our systems using this model?
- **Example:** AverageComputation.html,
- **Example:** AverageComputationMVC.html
- Let’s think about our projects and the use of MVC
Design: 2-Dim Application (Tic Tac Toe)

- How can we implement such a system?
Design: 2-Dim Application (Order Values)

- A board has a blank piece we can move which pushes numbers in the opposite direction
- How can we implement such a system?
Design: 2-Dim Application (Sudoku)

- Sudoku validator
- How can we implement such a validator?
innerHTML property

- Property defining:
  - HTML code
  - Text occurring between the opening and closing tags of the element
- Allows you to modify/define HTML content
- **Example:** GetContent.html
- **Example:** ModifyContent.html
  - What happens if you keep selecting the Modify button?