

**CH IV Domino Constructorpalooza Redux && Executable Preconditions**  
**COSC 2329 Component-Based Programming**  
**Assigned: Wednesday, March 21, 2018**  
**Due: Tuesday, March 27, 2018 11:59 PM**  
**Late Deadline: Wednesday, March 28, 2018 11:59 PM**

**Assignment**

Same general goal as Challenge III (Domino Constructorpalooza). In addition,

- take test case feedback from Challenge I and fix your code
- add in executable preconditions for each constructor
- note that I might change the values of MINIMUM\_PIP\_COUNT and MAXIMUM\_PIP\_COUNT

**Deliverables**

- A .zip file uploaded to Canvas that contains the following files  
(Look for "Domino" assignment or similar):
  - DominoHighLowImpl\_LastName.java (e.g., DominoHighLowImpl\_Kart.java)
  - DominoHighLowSetImpl\_LastName.java (e.g., DominoHighLowSetImpl\_Kart.java)
  - DominoLowDifferenceStringImpl\_LastName.java (e.g., DominoLowDifferenceStringImpl\_Kart.java)
  - Any supporting Utils/classes/interfaces that you created (note that the filename suffix on these files must be \_LastName)

**Rules**

- My test cases do not change based on your submission.
- **USE THE PACKAGE 'dominoes' for all of your files!**
- Use the Eclipse IDE
- Ensure that I, with only modest effort, can understand your code
- Ensure that the code is properly documented
- Ensure that the code is properly formatted
- **Test your code!** (What test cases can you think of?)
  - What are the "middle-of-the-road" (i.e., "vanilla") test cases?
  - What are the "corner" (i.e., "extreme") test cases?
- **Test your code some more!** (What other test cases can you think of?)
- Code that doesn't compile will not pass any tests and receive a score of 0.
- **Ensure that your files follow the naming convention under Deliverables**
- **WARNING: This specification may be misleading or incomplete! Part of the assignment is to read the assignment early, think about it, and ask any clarifying questions!**