

**COMS 311: Homework 4**  
**Due: April 19<sup>th</sup>, 11:59pm**  
**Total Points: 50**

**Late submission policy.** Any assignment submission that is late by not more than two business days from the deadline will be accepted with 20% penalty for each business day. That is, if a homework is due on Friday at 11:59 PM, then a Monday submission gets 20% penalty and a Tuesday submission gets another 20% penalty. After Tuesday no late submissions are accepted.

**Submission format.** You are required to include, in your assignment submission, the source code for each of the classes. The template code segment is given in package `cs311hw4.zip`. Write your class so that its package name is `edu.iastate.cs311.hw4`. Your Java source file (.java file) will be placed in the directory `edu/iastate/cs311/hw4` (Mac or Linux) or `edu\iastate\cs311\hw4` (Windows), as defined by the package specified above. Be sure to put down your name after the `@author` tag in each class source file. Make sure to create a .zip file of the directory `edu`. Your submitted .zip file should be named `Firstname.Lastname.HW4.zip`. You may submit a draft version of your code early to see if you have any submission problem on Canvas. We will grade only your latest submission. Note that only a .zip file is accepted. Each student must submit their own .zip file. If you discussed the homework or solutions with others, a list of collaborators must be included with each submission. Each of the collaborators has to write their own code (copying is not allowed). You are not allowed to post your Java solution source code on Canvas or in any public code repository, such as GitHub. However, you are allowed to share JUnit test code on Canvas.

**General Requirements** This assignment gives you the opportunity to implement the Dijkstra algorithm for computing single-source shortest paths for every vertex in a directed graph with no negative weights. This algorithm is useful for solving optimization problems formulated in terms of directed graphs with no negative weights. The assignment will help you understand what it takes to develop a useful computer program. Your task in this project is to implement the algorithm described in lecture. Please see the attached Java template code for the specification of the function to be implemented. You are required to use Java in this assignment.