

# Modularization Design

## LAB 13 : SEGREGATION SORT PROGRAM

**Due Saturday at 5:00 PM MST**

This week, we will implement the segregation sort we designed two weeks ago and analyzed last week.

### Program Description

Implement in Python the segregation sort algorithm we designed two weeks ago and verified last week. This is part three in a three-part sequence:

1. [Lab 10 : Segregation Sort Design](#)
2. [Lab 12 : Segregation Sort Analysis and Verification](#)
3. [Lab 13 : Segregation Sort Program](#)

### Assignment

The following is needed to turn this in:

1. Your source code
2. A demonstration video

### Source Code

Submit your source code as a file attachment in I-Learn. It must be possible to open this file in Python and execute it without any additional work. You might want to double-check this before pressing [Submit]. In other words, open a fresh Python window and paste the code in to make sure it works as you expect.

At the top of your program, include a comment answering these five questions:

```
# 1. Name:
#     -your name-
# 2. Assignment Name:
#     Lab 13 : Segregation Sort Program
# 3. Assignment Description:
#     -describe what this program is meant to do-
# 4. What was the hardest part? Be as specific as possible.
#     -a paragraph or two about how the assignment went for you-
# 5. How long did it take for you to complete the assignment?
#     -total time in hours including reading the assignment and submitting the program-
```

### Demonstration Video

Record a short video demonstrating the execution of your program. The video must be very short. No video longer than two minutes will be accepted. This means you might need to practice once or twice before recording the video to make sure that you demonstrated everything that is necessary.

Your demonstration video must cover the test cases you identified last week. Make sure to briefly describe your test cases before you demonstrate them.

After the video is recorded, provide a voice-over mentioning what test case you are covering. Post your video on some streaming service (like YouTube) and provide a link in your assignment submission.

## Assessment

Your grade for this activity will be according to the following rubric:

	Exceptional 100%	Good 90%	Acceptable 70%	Developing 50%	Missing 0%
Code Quality 50%	Code quality is outstanding in every detail; there is no room for improvement	Code is efficient, maintainable, has strong cohesion and loose coupling	One or two minor code quality nit-picks	There are one or two major code quality issues	Little attention was spent on code quality
Functionality 50%	All the test cases execute perfectly and are demonstrated in the video	Everything works, but there are minor cosmetic defects	One test case fails to execute as expected	At least one test case works as expected	Code does not run, is missing, or does not resemble a working solution