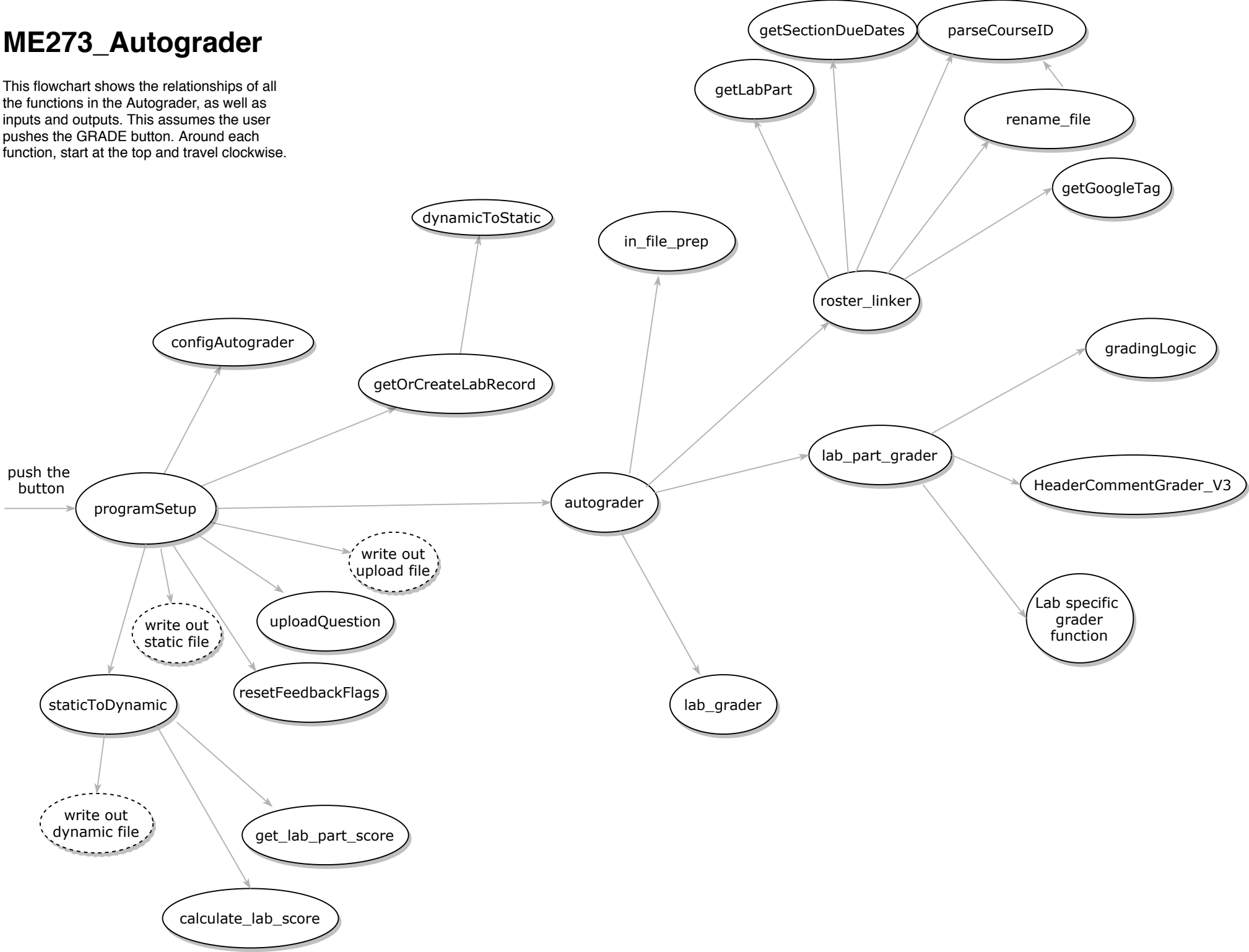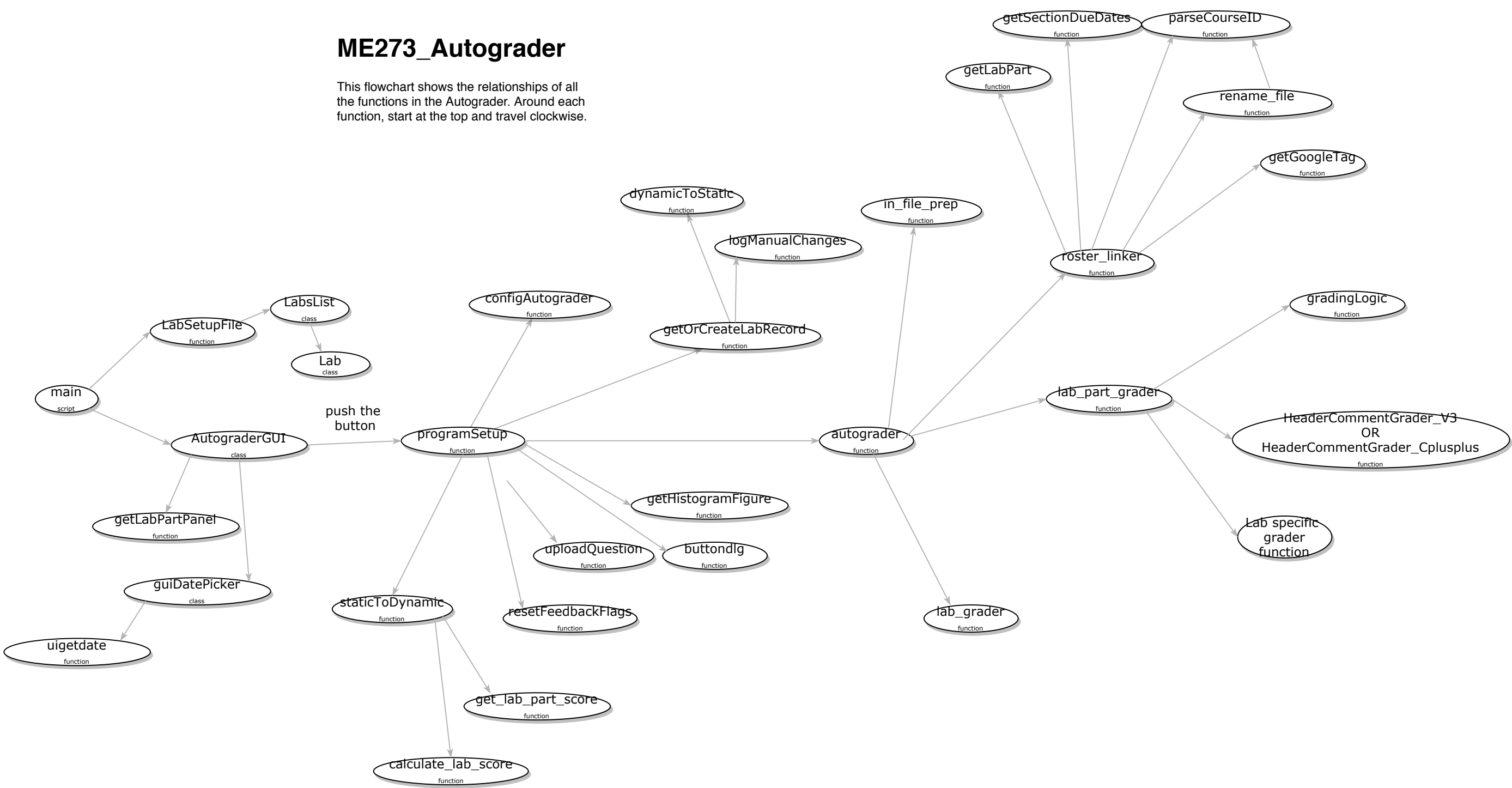# ME273_Autograder

This flowchart shows the relationships of all the functions in the Autograder, as well as inputs and outputs. This assumes the user pushes the GRADE button. Around each function, start at the top and travel clockwise.

getSectionDueDates

parseCourseID

getLabPart

rename_file

getGoogleTag

dynamicToStatic

in_file_prep

roster_linker

configAutograder

getOrCreateLabRecord

gradingLogic

lab_part_grader

push the button

programSetup

autograder

HeaderCommentGrader_V3

write out upload file

uploadQuestion

Lab specific grader function

write out static file

resetFeedbackFlags

staticToDynamic

lab_grader

write out dynamic file

get_lab_part_score

calculate_lab_score

ME273_Autograder

This flowchart shows the relationships of all the functions in the Autograder. Around each function, start at the top and travel clockwise.

getSectionDueDates
function

parseCourseID
function

getLabPart
function

rename_file
function

getGoogleTag
function

dynamicToStatic
function

logManualChanges
function

in_file_prep
function

roster_linker
function

configAutograder
function

getOrCreateLabRecord
function

gradingLogic
function

LabsList
class

LabSetupFile
function

Lab
class

lab_part_grader
function

main
script

AutograderGUI
class

push the button

programSetup
function

autograder
function

HeaderCommentGrader_V3
OR
HeaderCommentGrader_Cplusplus
function

getHistogramFigure
function

getLabPartPanel
function

uploadQuestion
function

buttonDlg
function

Lab specific grader function

guiDatePicker
class

staticToDynamic
function

resetFeedbackFlags
function

lab_grader
function

uigetdate
function

get_lab_part_score
function

calculate_lab_score
function

gradingLogic.m Flowchart

submitted a file?
- no → **old Score = 0**
  - no → copy over old scores / FF = old FF / gA = 2
  - yes → no submission / FF = old FF / gA = 4
- yes → **timestamp before deadline?**
  - no → copy over old scores / FF = old FF / gA = 2
  - yes → **finalGrading?**
    - yes → gA = 3 / FF = 1
    - no → **numSub < chances?**
      - no → copy over old scores / FF = old FF / gA = 2
      - yes → **oldFIleDate < newFileDate**
        - no → copy over old scores / FF = old FF / gA = 2
        - yes → feedback Grading / FF = old FF / numSub++ / gA = 1

FF - feedbackFlag
gA - gradingAction
numSub - number of previously graded student submissions

ME273_Autograder

This flowchart shows the relationships of all the functions in the Autograder, as well as inputs and outputs. This assumes the user has pushed the GRADE button. Around each function, start at the top and travel clockwise.
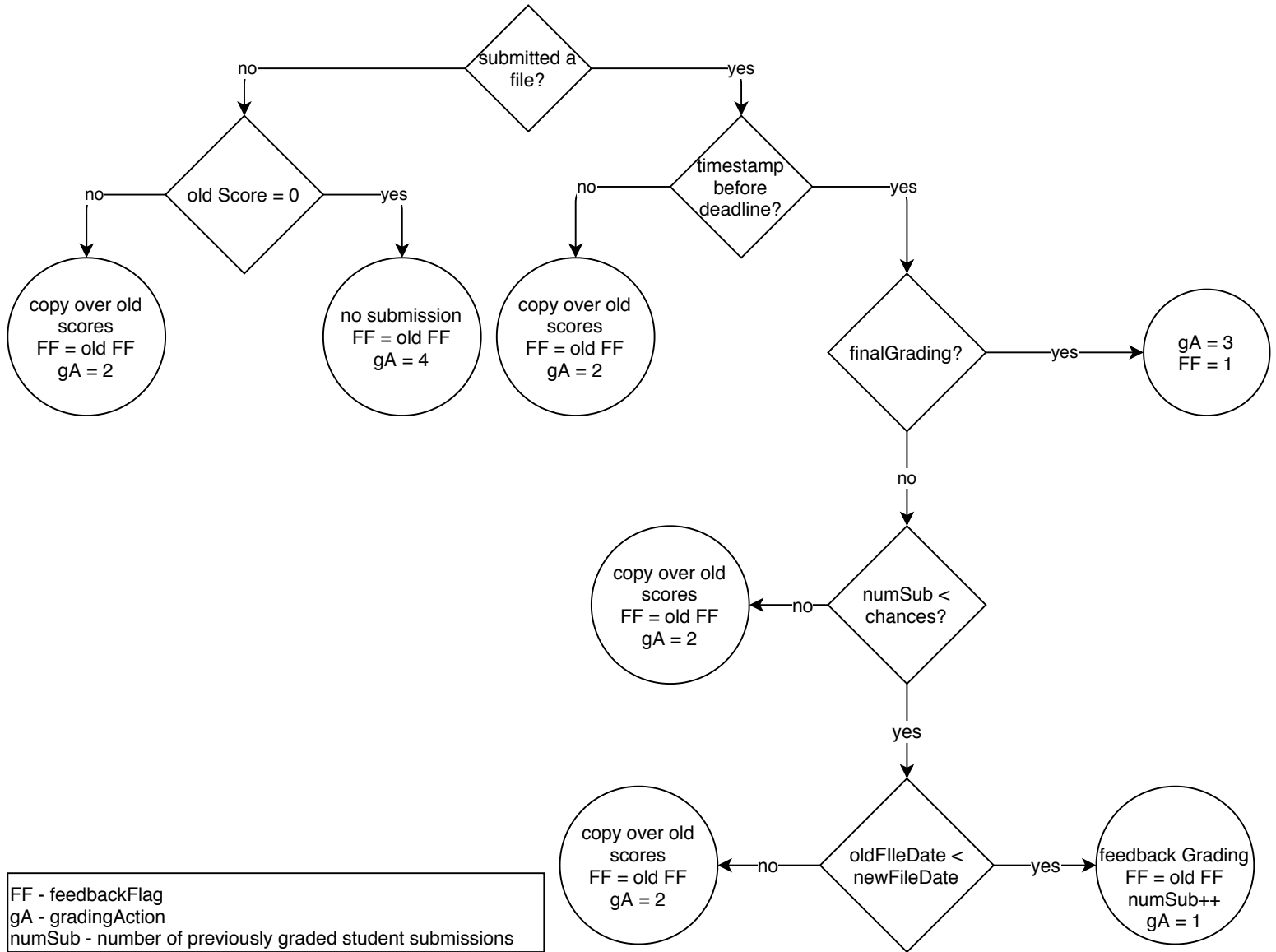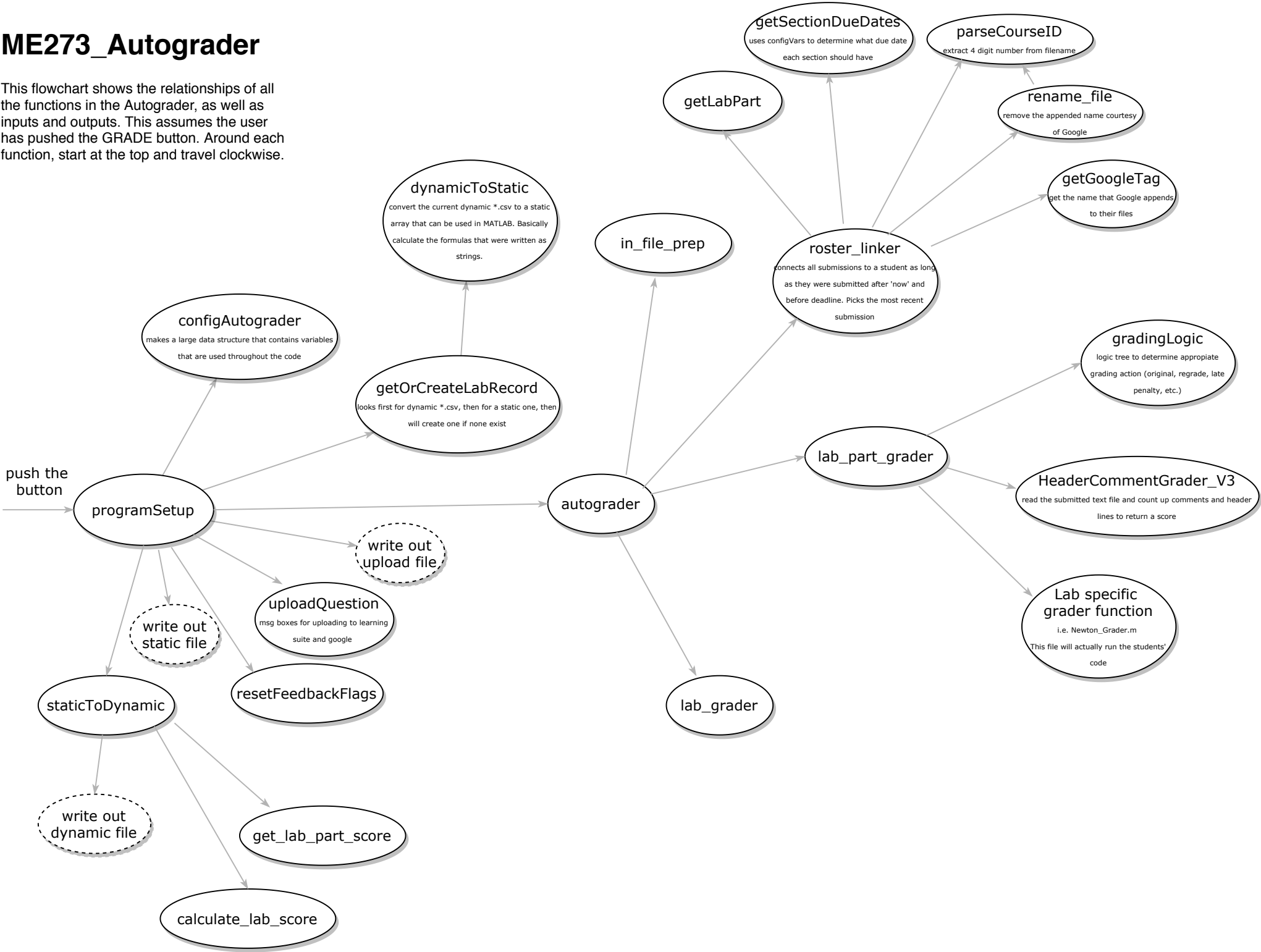
getSectionDueDates
uses configVars to determine what due date each section should have

parseCourseID
extract 4 digit number from filename

getLabPart

rename_file
remove the appended name courtesy of Google

dynamicToStatic
convert the current dynamic *.csv to a static array that can be used in MATLAB. Basically calculate the formulas that were written as strings.

in_file_prep

roster_linker
connects all submissions to a student as long as they were submitted after 'now' and before deadline. Picks the most recent submission

getGoogleTag
get the name that Google appends to their files

configAutograder
makes a large data structure that contains variables that are used throughout the code

getOrCreateLabRecord
looks first for dynamic *.csv, then for a static one, then will create one if none exist

gradingLogic
logic tree to determine appropiate grading action (original, regrade, late penalty, etc.)

lab_part_grader

HeaderCommentGrader_V3
read the submitted text file and count up comments and header lines to return a score

push the button

programSetup

autograder

write out upload file

uploadQuestion
msg boxes for uploading to learning suite and google

write out static file

resetFeedbackFlags

Lab specific grader function
i.e. Newton_Grader.m
This file will actually run the students' code

staticToDynamic

lab_grader

write out dynamic file

get_lab_part_score

calculate_lab_score

ME273_Autograder

This flowchart shows the relationships of all the functions in the Autograder, as well as inputs and outputs. This assumes the user pushes the GRADE button. Around each function, start at the top and travel clockwise.

look for the '_' in the filename from right to left, instead of left to right

getSectionDueDates

parseCourseID

getLabPart

rename_file

variable late penalty needed when reading in the dynamic file

getGoogleTag

dynamicToStatic

in_file_prep

roster_linker

One more condition for late grading: Was this file submitted after the first deadline. This prevents students who got a bad score but didn't resubmit from getting graded.

added two fields for late penalty and learning suite points

configAutograder

getOrCreateLabRecord

gradingLogic

two extra inputs (configVars, FirstDeadline)

lab_part_grader

push the button

programSetup

autograder

HeaderCommentGrader_V3

write out upload file

after all the grading is complete, create a new array that has the score column multiplied by 25. This will only be used as the upload file.

uploadQuestion

For a particular lab part, if the late flag is set by gradingLogic, only allow the score to be a max of 0.75

write out static file

Lab specific grader function

resetFeedbackFlags

lab_grader

staticToDynamic

write out dynamic file

get_lab_part_score

variable late penalty added to the get_lab_part_score

calculate_lab_score