```
# Improvements: read xlsx or csv , generate xlsx
# Allow to select file using Graphical interface
# As a application that opens up interface
# Use MapReduce
# Statistics , graph


# Input: Results of semester, ordered as
#          Multiple columns representing subject/subject code
#          Each row having Grades obtained by student, comma separated
#          Assuming grade as: S+, S, A, B, C, D, E, F, NE, NP, PP
#          Saved as a CSV, comma separated value file called result.csv
#
# To convert xlsx as csv:

# The Excel sheet should have columns that represent different subject, and
# each row content corresponding to grade obtained by the student for that subject

# Excel Sheet:
#         ________________________
#        |  Code1 | Code2 | Code3 |
#        |------------------------|
#        |  S1G1  | S1G2  | S1G3  |
#        |------------------------|
#        |  S2G1  | S2G2  | S2G3  |
#        |------------------------|
#        |  S3G1  | S3G2  | S3G3  |
#        |________|_______|_______|
#
# An option is available in Excel / Calc of Save as CSV
# Save the result.xlsx as result.csv
# Excel Sheet result.xlsx
#         ________________________
#        |  Code1 | Code2 | Code3 |
#        |------------------------|
#        |    S+  |   S   |   F   |
#        |------------------------|
#        |    S+  |   A   |   F   |
#        |------------------------|
#        |    F   |   S+  |   F   |
#        |________|_______|_______|

# The result.csv will now have:
#       Code1 , Code2, Code3
#       S+ , S , F
#       S+ , A , S
#       F  , F , F

# Output: Count of grades obtained in subject as csv
#        Example output file "analysis.csv"
#        CourseCode, S+, S, A, B, C, D, E, F, NE, NP, PP, TF, TP, PassPercent
#        SubjectCode1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 2, 66.66
#        SubjectCode2, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 100
#        SubjectCode3, 0, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 3, 0, 0
#
#        where TF is Total Failed
#        TP is Total Passed
#        PassPercent = TP/(TP+TF)

# If the analysis.csv file is opened in Excel / Calc then the same csv is
#   displayed in tabular format as

#  ____________________________________________________________________________
# |       | S+ | S | A | B | C | D | E | F | NE | NP | PP | TF | TP | PassPerc |
# |-------|----|---|---|---|---|---|---|---|----|----|----|----|----|----------|
# | Code1 |  2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0  | 0  | 0  | 1  | 2  |   66.66  |
# |-------|----|---|---|---|---|---|---|---|----|----|----|----|----|----------|
# | Code2 |  1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0  | 0  | 0  | 0  | 3  |    100   |
# |-------|----|---|---|---|---|---|---|---|----|----|----|----|----|----------|
# | Code2 |  0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0  | 0  | 0  | 3  | 0  |     0    |
# |_______|____|___|___|___|___|___|___|___|____|____|____|____|____|__________|




# initialize dictionary

# read result.csv

# For each student
#   For each subject grade
#     update dictionary
#
# write dictionaty to analysis.csv

import csv

with open('result.csv', 'r') as csvfile:
    results = csv.reader(csvfile, delimiter=',')

    subjectCode = next(results, None)

    print ( "Subject Codes = ", subjectCode )
    print ( "Total number of Subjects = ", len(subjectCode) )

    subject={}
    for code in subjectCode:
        subject[code] = { 'S+': 0, 'S': 0, 'A': 0, 'B': 0, 'C': 0, 'D': 0,
                          'E': 0, 'F': 0, 'NE': 0, 'NP': 0, 'PP': 0,
                          'TF':0, 'TP':0, }

    # consider PASS column of result

    totalNumberOfStudents = 0

    totalNumberOfPassStudents = 0

    totalNumberOfFailStudents = 0

    for row in results:

        totalNumberOfStudents = totalNumberOfStudents + 1

        fail = False

        for code, grade in zip( subjectCode, row ):

            if ( grade != '' ):

                subject[code][grade] = subject[code][grade] + 1
                # F is also incremented here

                if ( grade == 'F' ):
                    subject[code]['TF'] = subject[code]['TF'] + 1
                    fail = True
                elif ( grade == 'NE' ):
                    pass
                else:
                    subject[code]['TP'] = subject[code]['TP'] + 1

        if ( fail == True ):
            totalNumberOfFailStudents = totalNumberOfFailStudents + 1
        else:
            totalNumberOfPassStudents = totalNumberOfPassStudents + 1

with open('analysis.csv', 'w') as csvfile:
    writeAnalysis = csv.writer( csvfile , delimiter=',' )
    writeAnalysis.writerow( [ 'SubjectCode' , 'S+' , 'S' , 'A' , 'B' , 'C' ,
                              'D' , 'E' , 'F' , 'NE' , 'NP' , 'PP',
                              'TF', 'TP', 'PassPerc' ] )

    for code in subjectCode:
        if ( subject[code]['TF'] + subject[code]['TP'] ) == 0:
            passPercentage = 0
        else:
            passPercentage = ( 100.00 * subject[code]['TP'] ) / ( subject[code]
['TF'] + subject[code]['TP'] )
        writeAnalysis.writerow( [ code , subject[code]['S+'] ,
                                  subject[code]['S'] , subject[code]['A'] ,
                                  subject[code]['B'] , subject[code]['C'] ,
                                  subject[code]['D'] , subject[code]['E'] ,
                                  subject[code]['F'] , subject[code]['NE'] ,
                                  subject[code]['NP'] , subject[code]['PP'] ,
                                  subject[code]['TF'] , subject[code]['TP'],
                                  passPercentage ] )

    writeAnalysis.writerow( [ ] )
    writeAnalysis.writerow( [ ] )
    writeAnalysis.writerow( [ "OverallPerformance" ] )
    writeAnalysis.writerow( [ "TotalNumberOfStudents", totalNumberOfStudents ] )
    writeAnalysis.writerow( [ "TotalNumberOfFailStudents",
totalNumberOfFailStudents ] )
    writeAnalysis.writerow( [ "TotalNumberOfPassStudents",
totalNumberOfPassStudents ] )
    writeAnalysis.writerow( [ "PassPercentage",  ( 100.00 *
totalNumberOfPassStudents ) / ( totalNumberOfPassStudents +
totalNumberOfFailStudents ) ] )

    print("analysis.csv created")

# Save the result.csv and the Python script resultAnalysis.py is same folder
# Open Python Terminal and run this script as
# python resultAnalysis.py

# If input result.csv , hence result.xlsx format was correct, then
#   analysis.csv will be generated
# Now open analysis.csv in Excel / Calc

# Improvement: Use MapReduce?
```