```
#************************* CS421: Assignment 5 **************
#
#  There are four sections in this assignment. Each section is worth
2.5 points.
#  Skeleton code is already given.
#  You only need to add your code between BEGIN and END lines in each
section.
#
#  Use pythontutor.com to implement each section.
#  Save the complete implementation to a file called "a5_lists.py" and
submit the file to Google Classroom
# ************************************************************

#-------------------------------------------------------------------
-------
# A.5.1 --> Assume that some students registered twice for the same
class.
# You need to write a program to remove the duplicate registrations
from a list
#-------------------------------------------------------------------
-------

# define students list
students = ["abe", "barb", "chris", "abe", "dan", "chris", "ellie"]
print("All students --> ", students)

#=========================================
# BEGIN -- your code

students_temp = []
for x in students:
    if x not in students_temp:
        students_temp.append(x)

students = students_temp

# END -- your code
#=========================================

# print students list. This should NOT contain any duplicates
print("Unique students --> ", students)


#-------------------------------------------------------------------
-------
#A.5.2 --> Assume that some students registered both html and python
classes
# Find out the list of students who registered for both the classes.
#-------------------------------------------------------------------
-------


html = [ "barb", "dan", "ellie", "abe", "chris"]
python = ["henry", "chris", "dan", "ellie", "frank", "gavin"]
dupe_list = []
```

```
#=========================================
# BEGIN -- your code
# It is faster if start the for loop on a smaller list
# and to check whether those elements are in the longer list.
if len(html) < len(python):
    small_list = html
    large_list = python
else:
    small_list = python
    large_list = html

dupe_list = [ ]
for x in small_list:
    if x in large_list:
        dupe_list.append(x)

# END -- your code
#=========================================

print("html students --> ", html)
print("python students --> ", python)
print("duplicates --> ", dupe_list)


#-------------------------------------------------------------------
-------
#A.5.3 --> Assume that html class is overcrowded with too many
registrations.
# Since that class is too big, SILC decided to split the HTML class
# into two sections html_a and html_b
# All the students whose name starts with (a, b....,l, m) will be in
html_1
# And all the students whose name starts with (n,o,...., y,z) will be
in html_2
#
# You are given a big list called "html"
# Write python code to create two new lists "html_a" and "html_b" per
the above logic.
# Finally, print all three lists in alphabetical order
#-------------------------------------------------------------------
-------

html = [ "guy",
"madeline",
"parker",
"chris",
"tom",
"ursula",
"ramesh",
"lisa",
"staci",
"jordan",
"emmett",
"vinny",
"brian",
"zora",
```

```
"oliver",
"polly",
"kingston",
"olivia",
"xavier",
"fiona",
"zack",
"harmony",
"barb",
"samson",
"ariel",
"emma",
"yasmine",
"crystal",
"dan",
"xenia",
"irving",
"tiffany",
"noah",
"umesh",
"yates",
"victoria",
"desiree",
"quinn",
"wendy",
"frank",
"henry",
"mike",
"isabella",
"nora",
"julie",
"lincoln",
"alex",
"kim",
"raven",
"watson",
"ganga"
]
html_a = []
html_b = []

#=========================================
# BEGIN -- your code
for x in html:
    if x < "n":
        html_a.append(x)
    else:
        html_b.append(x)

# sort all the three lists
html.sort()
html_a.sort()
html_b.sort()
# END -- your code
#=========================================
```

```
# when printed, all the lists should be alphabetically sorted
print("html --> ", html)
print("html_a --> ", html_a)
print("html_b --> ", html_b)


#-------------------------------------------------------------------
-------
# A.5.4 --> Assume that python class has 10 students.
# Instructor is keeping track of their attendance on every saturday.
# by keeping the list of students present in another list.
# So, You are given an original list of 10 students.
# And for each saturday, another smaller list is given to you.
# You will write a program to provide attendance chart as follows
#
#    s1    A    P   P   P
#    s2    P    P   P   A
#    .....................
#    s10   A    A   A   A
#-------------------------------------------------------------------
-------

# define students list
python = ["abe", "barb", "chris", "dan", "ellie", "gabby", "henry",
"isabelle", "jack", "larry"]

# define the attendance list
week_1  = ["barb", "chris", "dan", "ellie", "henry", "isabelle",
"jack"]
week_2  = ["abe", "barb", "chris",  "ellie", "gabby", "henry",
"isabelle",  "larry"]
week_3  = ["abe", "barb",  "henry", "isabelle", "jack", "larry"]
week_4  = ["abe", "barb", "chris", "dan", "ellie", "gabby", "henry",
"isabelle", "jack"]

# defint the list to hold the attendance
attendance_report = [ ]

#=========================================
# BEGIN -- your code

for x in python:
    # Assume that the student is present for all the weeks
    x_record = [x, "P", "P", "P", "P"]

    # if the student is absent, then we will update that recrods
    if x not in week_1:
        x_record[1] = "A"
    if x not in week_2:
        x_record[2] = "A"
    if x not in week_3:
        x_record[3] = "A"
    if x not in week_4:
        x_record[4] = "A"

    # add the student's attendance record to the report
    attendance_report.append(x_record)
```

```
# END -- your code
#=========================================

# print students list. This should NOT contain any duplicates
for x in attendance_report:
    print(*x)
```