BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Work Integrated Learning Programs Division
MTech (Data Science and Engineering)

(S1-21_DSECCZG519)

(Data Structures and Algorithms Design)

Academic Year 2021-2022

Assignment 2 – PS05 - [Confectionery] - [Weightage 13%]

1. Problem Statement

A confectionery shop has a huge collection of chocolates and to arrange them in a certain order the shop keeper wants to know the increasing/decreasing order of their number. They require your assistance to design an efficient in-place algorithm that will help them to find out the names of the chocolates ordered (as specified by the shopkeeper) by the number in least possible time.

For the ease of implementation assume that number of chocolates is unique in type of chocolate i.e., no two types of chocolate have same number.

2. Requirements

- Formulate an efficient recursive algorithm using Divide and Conquer to perform the above task
- Also develop a non-recursive, linear-time algorithm for above problem.
- Analyse the time complexity for requirement 1.
- Implement the above problem statement using Python 3.7

3. Sample file formats

Sample Input:

Input should be taken in through a file called "**inputPS05.txt**" which has the fixed format mentioned below using the "/" as a field separator:
Chocolate: <Chocolate name>
Number: <count>
Order chocolate by number: <required display order by number>
Ex.

Chocolate: Nestle / Amul / Parle / M&M / Mars / Campco
Number: 123/ 231/ 89/ 156/ 99/ 278
Order chocolate by number: High to Low
…

Note that the input/output data shown here is only for understanding and testing, the actual file used for evaluation will be different.

Sample Output:

Divide & Conquer:
Chocolate: Campco / Amul / M&M / Nestle / Mars / Parle
Number: 278/ 231/ 156/ 123/ 99/ 89

Iterative Solution:
Chocolate: Campco / Amul / M&M / Nestle / Mars / Parle
Number: 278/ 231/ 156/ 123/ 99/ 89

Display the output in **outputPS05.txt**.

Note that the input/output data shown here is only for understanding and testing, the actual file used for evaluation will be different.

4. Deliverables

1. Word document **designPS05_<group id>.docx** detailing your design and time complexity of the algorithm and a contribution table as below:

Roll Number	Name	Contribution (%)

** Note: Each group should upload **one** submission only (Individual submissions are not valid)

2. **inputPS05.txt** file used for testing
3. **outputPS05.txt** file generated while testing
4. ***.ipynb (python notebook) file** containing the python code. Create a single notebook. Do not fragment your code into multiple files
5. Zip all the above files including the design document in a folder with the name:

   [Group id] _A2_PS05_Confectionery.zip and submit the zipped file.

Group Id should be given as **Gxxx** where xxx is your group number. For example, if your group is 26, then you will enter G026 as your group id.

5. Instructions

1. It is compulsory to make use of the data structure(s) / algorithms mentioned in the problem statement.
2. Ensure that all data structure insert and delete operations throw appropriate messages when their capacity is empty or full. Also ensure basic error handling is implemented.
3. For the purposes of testing, you may implement some functions to print the data structures or other test data. But all such functions must be commented before submission.
4. Make sure that your read, understand, and follow all the instructions
5. Ensure that the input, prompt, and output file guidelines are adhered to. Deviations from the mentioned formats will not be entertained.
6. The input, prompt and output samples shown here are only a representation of the syntax to be used. Actual files used to evaluate the submissions will be different. Hence, do not hard code any values into the code.
7. *Run time analysis is to be provided in asymptotic notations and not timestamp based runtimes in sec or milliseconds.*

6. Instructions for use of Python:

1. Implement the above problem statement using Python 3.7 or above
2. Use only native data types like lists and tuples in Python, do not use dictionaries provided in Python. Use of external libraries like graph, numpy, pandas library etc. is not allowed. The purpose of the assignment is for you to learn how these data structures are constructed and how they work internally.
3. Create a single notebook file for code. Do not fragment your code into multiple files.
4. Read the input file and create the output file in the root folder itself along with your notebook file. Do not create separate folders for input and output files.

7. Deadline

1. The strict deadline for submission of the assignment is **27th Feb 2022 11:55 PM**
2. The deadline has been set considering extra days from the regular duration to accommodate any challenges you might face. No further extensions will be entertained.

3. Late submissions will not be evaluated.

8. How to submit

1. This is a group assignment.
2. Each group must make one submission (only one, no resubmission) of solutions.
3. Each group should zip all the deliverables in one zip file and name the zipped file as mentioned above.
4. Assignments should be submitted via Canvas > Assignment section. Assignment submitted via other means like email etc. will not be graded.

9. Evaluation

1. The assignment carries 13 Marks.
2. Grading will depend on
   a. Fully executable code with all functionalities working as expected
   b. Well-structured and commented code
   c. Accuracy of the run time analysis and design document.
3. Every bug in the functionality will have negative marking.
4. Marks will be deducted if your program fails to read the input file used for evaluation due to change / deviation from the required syntax.
5. Use of only native data types and avoiding libraries like numpy, graph and pandas will get additional marks.
6. Plagiarism Policy:
   a. All the assignment submissions will go through plagiarism checks (both automated tools and manual checks).
   b. If any team's plagiarism score exceeds 75% and above, they would be awarded zero. (All the teams involved in the plagiarism will be awarded 0).
   c. If a team is booked under plagiarism and awarded 0, no further discussions around it will be entertained.
   d. With respect to the assignment, Plagiarism means:
      i. Copied partially/completely from other teams
      ii. Copied partially/completely from any internet sources [If there is a dire need, do cite the resources as references].
   e. Protecting your work is the responsibility of the team.

7. Source code files which contain compilation errors will get at most 25% of the value of that question.

10. Readings:

Textbook: Algorithms Design: Foundations, Analysis and Internet Examples Michael T. Goodrich, Roberto Tamassia, 2006, Wiley (Students Edition). Chapters: 5.2