Contents

Problem Statements

1) There is a csv file called **employee.csv**.The file contains employee details with columns: Name, Age, Department, Salary, and Years_of_Experience.

   a) Write a function department_with_highest_experience() that calculates the average years of experience for each department and returns the department with the highest average experience. (Example Output: 'Engineering', 6.5)

   b) Write a function salary_by_age_group() that groups employees into age ranges (20-30, 31-40, etc.) and calculates the total salary for each age group.(Example Output: {'20-30': 148000, '31-40': 125000, '41-50': 0})

   c) Write a function employees_above_average_salary() that returns a list of employee names who are paid above the average salary for their respective department.

   d) Write a function give_raises that increases the salary of all employees with more than 5 years of experience by 10%. The updated data should be saved back into the employees.csv file.

2) A startup is conducting a feedback program of a product, stored in a text file named **feedback.txt**, where each line represents one piece of feedback. The company wants to analyze the feedback to understand customer sentiment around certain aspects, such as "service," "price," and "quality." Each feedback line could express either positive or negative sentiment. Positive feedback often contains keywords such as "excellent," "great," "satisfied," and "happy," while negative feedback includes words like "poor," "bad," "disappointed," and "unsatisfied." Write a Python program to read feedback.txt, analyze the following.
    a. Count keyword occurrences: Count how often each specified keyword appears in the feedback.
    b. Separate good and bad feedback: Save positive feedback lines in a file named good_feedback.txt and negative feedback lines in bad_feedback.txt.
    c. Summarize counts: Save the keyword counts and the overall counts of good and bad feedback lines in keyword_counts.csv, without overwriting any existing data in this file.

    Avoid overwriting- Ensure that no previous data is lost or overwritten in
    good_feedback.txt and bad_feedback.txt; if these files exist, append new
    data to them.

3) A restaurant chain wants to calculate the final cost of a meal, taking into account various discounts, service charges, and taxes. Write a program that has 3 functions that call each other to get the final price. Each step involves multiple conditions based on default parameters:
    a. apply_discount: This function reduces the base price by a default discount rate (e.g., 10%). Additionally, if the customer is a member, it applies an additional loyalty discount (default 5%). If it's a special promotion day, it applies a further discount (default 3%).
    b. add_service_charge: Calls apply_discount, then adds a service charge based on the type of meal (dine-in or takeout) with default rates. It adds a flat surcharge for dine-in orders, whereas for takeout, it calculates a percentage of the discounted price.
    c. calculate_final_price: Calls add_service_charge and applies a tax. For high-value orders (default threshold $100), an additional luxury tax (default 2%) is applied.

In the main program, call calculate_final_price with parameters to test different scenarios. Make sure to use default parameters wherever required.

4) You have a folder with 10 students output files and a main key output file. Each student output file and the key output file contain 10 lines of text. Your task is to compare the content of each student output file with the key file line by line and assign marks based on the correctness of each line. Each line is worth 2 marks, so the maximum score for a student is 20 marks. (files required are given.)

Department of CSE, PES University
UE24CS151A-Python for Computational Problem Solving
Laboratory Week 3

Problem Statements

1. Input the base (b) and height (h) of a triangle and calculate its area where
   Area of triangle =0.5* b * h

2. You are traveling to Kenya, if 1 Kenyan shilling = 0.65 Indian rupee. Calculate how many shillings you will have based on the input in Indian Rupees.

3. A grocery store offers a 10% discount on all purchases. Write a program that asks the user for the total amount of their purchase, applies the discount and prints the total.

4. You need to introduce yourself to your new classmates. What will you say as an ice breaker?
   Display it in the format: My name is _____ and a fun fact about me is ____

5. You go out for dinner with your friends and need to split the bill equally, how will you do so by taking input of the bill amount and number of friends?

6. You have a certain amount of total work(n) that needs to be shared among a fixed number of workers m. Write a program to calculate the amount of work for each worker and how much extra work is left. Take input for no. of work and no. of workers.
   eg:
   n = 100 m = 6
   work on each worker = 16 extra work left = 4
7. Tiling of a rectangular floor is performed. Write a program that calculates the number of tiles required and the total cost to tile the entire floor. Take the dimensions of the floor and the size and price of each tile as input from the user. Also consider labor charges.
8. Two persons A and B are conducting a Chemical experiment. In which A is user element D and E in ratio of 2:1 and B uses the ratio of 4:3, calculate the mass of the compound if each generates a 1 mole gram of substance given that the mass of D is 1 unit(amu) and E is 5 units(amu). Take the values from the user.
   Hint - Consider using split() method on the input string
   eg- "1:2".split(":") # [‘1’,’2’]
   a,b = "1:2".split(":") # a <- '1' ; b <- ‘2
   a,b = int(a), int(b) # type casting char/string to integers
9. Given a 4 digit integer no. , display the individual digit & compute the sum of digits.

10. You are asked to color quarter of a circle with red, one third of the remaining with blue and rest in yellow, find the area covered by each color and perimeter of each color (include inner boundaries also). Take radius as input.
11. You've just celebrated your birthday, and you're curious about how many months, days, and hours you've been alive. Write a Python program that takes your age(in years) as input and prints your age in months and days.
12 Imagine you've just opened a savings account at your local bank. You're curious how much interest you'll earn after a few years. Write a Python program that calculates the simple interest based on the amount you've deposited, the bank's interest rate, and the number of years you plan to keep the money in the account.
13. You're planning a vacation to a different country, but their weather forecasts are in Celsius, and you're more familiar with Fahrenheit. Write a Python program to help you convert temperatures from Celsius to Fahrenheit so you can pack accordingly for your trip! (use the formula: F = C * 9/5 + 32)

14. You're baking a cake, but the recipe you're following is for 12 servings, and you need to make enough for a different number of people. Write a Python program that takes the number of servings you want to make and adjusts the ingredients accordingly. If the recipe calls for 2 cups of sugar for 12 servings, calculate how many cups of sugar you'll need for the number of servings entered by the user.

15. You're throwing a pizza party for your friends, and you want to make sure everyone gets an equal number of slices. Write a Python program that takes the number of pizzas, slices per pizza, and the number of people attending the party as input, and calculates how many slices each person gets. Also, calculate how many slices will be left over.

SSS

Department of CSE, PES University
UE24CS151A-Python for Computational Problem Solving
Laboratory Week 4

1. Write a Python program to Sum only even natural numbers up to n terms given by the user. Use control flow tokens.

2. Write a Python program to find the factorial of a number using loops. Take input from the user.

3. Create a Python program that counts down from 10 to 1 using a while loop. For each number, if the number is odd, print "Odd: <number>", and if it is even, print "Even: <number>". When it reaches 0, print "Blast off!".
   Expected output:
   Odd: 9
   Even: 8
   Odd: 7
   Even: 6
   Odd: 5
   Even: 4
   Odd: 3
   Even: 2
   Odd: 1
   Blast off!

4. Write a Python program that takes a number from the user and prints its multiplication table (from 1 to 10) using a for loop.

5. Create a Python program that prints the numbers from 1 to 30. For multiples of 3, print "Fizz" instead of the number, and for multiples of 5, print "Buzz". For numbers that are multiples of both 3 and 5, print "FizzBuzz".
   Expected output:
   1
   2
   Fizz
   4
   Buzz
   Fizz
   7
   8
   Fizz
   Buzz
   11
   Fizz
   13
   14
   FizzBuzz

6. Write a Python program that takes a number as input and checks whether the number is prime or not. A prime number is a number greater than 1 that has no divisors other than 1 and itself.

7. Write a Python program to print a right-angled triangle of numbers where each row contains the number repeated multiple times. The number of rows is provided by the user.

   For input 5
   1
   22
   333
   4444
   55555

8. Write a Python program to find the sum of the digits of a given number. For eg. , for input 123, the output should be 6 (1 + 2 + 3 = 6).

9. John is using a computer in which arithmetic operations (+, -, *, /, **) are not working. But as his assignments deadline are near he is looking for alternative methods. Help him out in solving the below questions using other operators.
   a) To calculate the power of 2 up to n terms.
      Eg. 2^0, 2^1, …… ,$2^{(n-1)}$
   b) Check whether a given number is even or odd.
      Hint - express the given num in the binary number system for analysis and check the least significant bit. (use pen and paper to analyze)

      Eg. 5 => 101
      10 => 1010

   c) Multiple a given number by 33.
      Hint - $(n * 2^5 + n)$
      - $(33 = 2^5 + 2^0)$
      Try using “bitwise or” and “bitwise shifts” operators instead of addition and multiplication operators.

10. Imagine you are a student calculating your final semester grade. You have 5 subjects, and each subject has a score between 1 and 100. Write a Python program that:

- Takes the score of each subject using a for loop.
- Calculates the average score.
- Assigns a grade based on the following scale:
  - 90 and above: Grade A
  - 80-89: Grade B
  - 70-79: Grade C
  - 60-69: Grade D
  - Below 60: Grade F

- Prints the average score and the corresponding grade.

Problem Statements

1. Write a python code to accept a list and print each item on a separate line.
2. Write a python code to check if a list is sorted, if not sort the list in ascending order.
3. Write a python program to split a List into Even and Odd Elements
4. Write a program that takes a list of integers and returns the count of the least frequently occurring element. Assume that only one element has the minimum frequency.

5. You are designing an attendance system. In which attendance of the day is maintained in the given format.

```
attendance_records = [("A", "P"), ("B", "A"), ("C", "P"), ("D", "A"), ("E", "P"), ("F", "P"), ("G", "A"), ("H", "P"), ("I", "P"), ("J", "P")]
```

   i. Calculate the total no of present count and absence count of the day.
   ii. It seems attendance of the student is wrong, check whether the attendance is actually wrong, if so change it in place. (take the name of the student and new statues as input.)
   iii. Suppose you are adding a new student to the list. Add using list methods.

6. **Music Playlist Management:** You are building a playlist management system. Write a python program using lists to:
   i. Merge two separate playlists into one.
   ii. Remove duplicate songs from the merged playlist.
   iii. Rotate the merged playlist by n positions.
   iv. Create two separate playlists from the merged one: one for songs starting with vowels and one for songs starting with consonants.

7. **Sales Analysis:** You are given a list of daily sales amounts for a store over a month. Write a program to:

   i. Remove any sales data that is a negative value.

   ii. Calculate the average sales for the month.

   iii. Find the day with the highest sales.

   iv. Find the day with the lowest sales.

8. Krishna is a teacher who wants to put up math tables in a cells format.

1	2	3	4	5	…n
2	4	6	8	10	
3	6	9	12	15	
4	8	12	16	20	
5	10	15	20	25	
.					
.					
.					
n	…	…	…	…	…n*n

So help him out to display the above by writing a python program to achieve the same up to n numbers.

9. Jone is running a small business where he lends out vehicles on rent. He needs to track vehicle's, borrower details, and rented details in a secured data structure.

Eg- Vehicle - Type - Car
- Reg_no - 2478c
Borrower details - Name - Alice
- Phone_no - 975318642
Rent Details - Borrow date - 15/10/2024 10.30.15 AM
- Expected return - 18/10/2024 10.30.15 AM
- rent - Rs. 1000 per day
- extra_rent - Rs 50 per hour
- deposit - Rs 1500
- in_hand - True

10. Write a program that works dynamically and stores the required details in a well structured data structure (use lists and tuples and combination of both). Add features to check rent details, add new rent, calculate rent the entry from the data structure and show all entries.(make in_hand to false on return). Make use of models such as “datetime” for date and time usage.

May use this template:

```
from datetime import datetime
main_ds = []
while True:
  print("--- Vehicle Rental System ---")
  print("1. Add Rental")
  print("2. Calculate and Return Vehicle")
  print("3. Show All Rentals")
  print("4. Exit")
  opt = int(input("Enter the Option: "))
  if opt == 1:
     pass
  elif opt == 2:
     pass
```

```
elif opt == 3:
    pass
elif opt == 4:
    pass
else:
    pass
```

Problem Statements

1. Event Attendance:

   A school has two events, and the attendees are stored in two sets. Create 2 sets with names of students. Write a program to:

- Find the students who attended both events.
- Find the students who attended only one of the events.
- Find all students who attended at least one event. (use '|' operator)

2. Student Marks Calculation:

   You are given a dictionary containing student names as keys and their marks as values.

   students = {

   "Rohan": 85,

   "Spoorthi": 90,

   "Aditi": 78,

   "Tanya": 92}

   Write a program to:

- Find the student with the highest marks.
- Calculate the average marks for the class.
- Add a new student and their marks to the dictionary.

3. Sentence Analysis:

Given a sentence, write a program to:

- Count the number of vowels and consonants.
- Find the longest word in the sentence.
- Reverse the sentence.

4. You are given a list of integers , sort the list based on the frequency of the number of occurrences of the elements. Take input of your choice.

5. You went shopping with your family at a local supermarket. Each family member picked up different items independently, creating their own shopping lists. To avoid buying duplicate items, you need to analyze these lists and calculate the final billing amount.

6. You work at a movie theater that keeps track of daily bookings. Each booking record contains the customer's name, selected movie, and seat number in a specific format. The theater needs to analyze these booking records to manage seating and prevent duplicate bookings. Take input from the user.

   Input format - "CustomerName-MovieName-SeatNumber"

   Hints:

   - Consider using string methods like split(), replace()
   - Dictionary and set data structures may be used for tracking duplicates

7. Write a Python program that takes two strings and checks if they are anagrams of each other. Ignore spaces and punctuation, and consider the comparison to be case-insensitive. For example, "Astronomer" and "Moon starer" should be identified as anagrams.

8. Write a Python program that takes a string as input and finds the first non-repeating character using a dictionary. If a non-repeating character is found, print it; else, print an appropriate message.

9. Password Validator

   Define a simple password validator. The password must:

- Be 8 characters long
- Must contain 1 Upper case character, 1 lower case character and 1 number
- If valid return valid password if not print invalid

10. Isomorphic Strings

Given two strings s and t, determine if they are isomorphic. Two strings s and t are isomorphic if the characters in s can be replaced to get t. All occurrences of a character must be replaced with another character while preserving the order of characters. No two characters may map to the same character, but a character may map to itself.

11. Anonymous Feedback Aggregator

    Scenario: Your company collects anonymous feedback from employees with a dictionary storing feedback themes as keys and a list of feedback messages as values.

    Use the dictionary feedback = {

    "Work Environment": ["Great work culture", "Need more team activities"],

```
    "Salary": ["Fair pay, but bonuses are inconsistent"],
    "Management": ["Leadership can improve", "Need more transparency"],
}
```

Write a program to:

- Add new feedback to the relevant theme.
- Identify the theme with the most feedback.