Problem Set – Loops & Functions

1. Enter destination city, miles travelled to get there and gallons of gasoline used for any number of trips entered at the keyboard (use ctl+z to stop). Use a function to compute miles per gallon. Display the destination city and miles per gallon for each trip entered. Sum the miles travelled and give a count of the number of trips made. Display these at the end of the program.

input	process	output
Destination city Miles traveled Gallons of gas used	Use a while loop to allow multiple entries (stop with Ctrl+Z). Call a function named compute_mpg that divides miles traveled by gallons used. Return miles per gallon and display it with the city name. Keep a running total of total miles traveled and number of trips entered. After all entries, display total miles and total number of trips.	City name Miles per gallon Total miles and total number of trips

2. Allow the employee to enter last name, job code and hours worked (use ctl+z to stop). Use a function to calculate pay. (Job code L is $25/hr, A is $30/hr and J is $50/hr). Give time and a half for overtime. Display last name and pay for each employee. Sum the pay for each employee as well as count the entries made. After all entries are made, compute and display the average pay and the number of entries made.

input	process	output
Employee last name Job code (L, A, or J) Hours worked	Use a while loop to allow many entries (stop with Ctrl+Z). Use a function named compute_pay to calculate pay. L = $25/hr, A = $30/hr, J = $50/hr. If hours > 40, give time-and-a-half for overtime. Add all pay values and count the number of employees.	Employee last name Pay for each employee Average pay and number of employees entered

	Display average pay at the end.	

3. Allow any number of students to enter their last name and the credits taken (use ctl+z to stop). Charge $250 per credit hour. Use a function to compute total tuition. Display student last name, credits taken and tuition owed. Sum tuition and give a count of the number of students who entered data.

input	process	output
Student last name Number of credit hours	Use a while loop (stop with Ctrl+Z). Use a function named compute_tuition. Tuition = credit hours × $250. Add all tuitions and count students entered.	Last name and tuition owed for each student Total tuition and total number of students

4. Any number of customers will enter a product code (W, C, G) and a quantity (ctl+z to stop). Use a function to determine unit price. Write another function to compute shipping. Then compute the total. Display the product code, unit price, shipping, extended price (quantity x unit price) and total for the order for each entry. Sum and display the total of all entries made.

Product Code	Unit Price	Shipping
W	$10.00	$2.00
C	$15.00	$5.00
G	$20.00	$7.00

input	process	output
Product code (W, C, or G) Quantity ordered	Use a while loop (stop with Ctrl+Z). Use two functions: get_price → returns the price for each product. get_shipping → returns the shipping cost.	Product code, price, shipping, extended total Overall total and number of orders

	Compute extended price (price × quantity). Compute total (extended + shipping). Add to overall totals and count orders.	

5. Allow students to enter the department and course code as noted below for any number of courses (ctl+z to stop). Use a function to determine the lab fee also in the table below. For each entry display the department, course code and lab fee. Give the total of all lab fees to collect. Compute and display the average lab fee.

Department	Course Code	Lab Fee
CIS	101	$50.00
CIS	121	$100.00
MAT	111	$25.00
MAT	112	$35.00
ENG	100	$55.00
All Others		$50.00

input	process	output
Department name (CIS,MAT,ENG, or others), course code, (Ctrl+Z to stop)	While not end of file (Ctrl+Z to stop) Function: find_lab_fee Input: dept, course_code If dept = CIS and code = 101 → fee = 50 If dept = CIS and code = 121 → fee = 100 If dept = MAT and code = 111 → fee = 25 If dept = MAT and code = 112 → fee = 35 If dept = ENG and code = 100 → fee = 55	Department,course code, lab fee, total fees, average fee

	Else → fee = 50 (All others) Return fee Display dept, course_code, fee Add fee to total_fees Increase count by 1 After loop ends: Compute average = total_fees / count	