Anil Neerukonda Institute of Technology & Sciences
Department of Computer Science & Engineering (AI & ML, DS)

CSE117 Problem Solving with C

Handout

Programs for Record

Exercise 1: **Miles to Kilometers**

Your summer surveying job requires you to study some maps that give distances in kilometers and some that use miles. You and your coworkers prefer to deal in metric measurements. Write a program that performs the necessary conversion.

Input: miles /*the distance in miles*/
Output : kms /*the distance in kilometers*/
Relevant formula : 1 mile = 1.609 kilometers

Design, algorithm, flow chart, program using the above data requirements for the given problem. Try the sample test cases given below.

Sample Test Cases	Input	Output
Test case 1	10	16.09
Test case 2	2	3.218

Exercise 2: **Supermarket Coin Processor**

You are drafting software for the machines placed at the front of supermarkets to convert change to personalized credit slips. In this draft, the user will manually enter the number of each kind of coin in the collection, but in the final version, these counts will be provided by the code that interfaces with the counting devices in the machine.

Problem input

```
char first, middle, last  /*customer's initials*/
int dollars;              /*number of dollars*/
int quarters;             /*number of quarters*/
int dimes;                /*number of dimes*/
int nickles;              /*number of nickles*/
int pennies;              /*number of pennies*/
```

Problem outputs

```
int total_dollars;        /*total dollar value*/
int change;               /*leftover change*/
```

Additional program variables
int total_cents; /*total value in cents*/

** Note: Pennies - 1 cent ; Nickels - 5 cents; Dimes - 10 cents; Quarters - 25 cents.; Dollar - 100 cents
Design, algorithm, flow chart, program using the above data requirements for the given problem. Try the sample test cases given below.

Testing tip: to test this program, try running it with a combination of coins that yield an exact dollar amount with no leftover change. For example, 1 dollar, 8 quarters, 0 dimes, 35 nickels, 25 pennies should yield a value of 5 dollars and 0 cents. Then increase and decrease the quantity of pennies by 1(26 and 24 pennies) to make sure that these cases are also handled properly.

Sample Test Cases	**Input**	**Output**
Test case 1	Your 3 initials : ABP Please enter your coin information. Number of dollars : 2 Number of quarters : 14 Number of dimes : 12 Number of nickels : 25 Number of pennies : 131	ABP coin credit Dollars : 9 Change : 26 cents
Test case 2	Your 3 initials : ABP Please enter your coin information. Number of dollars : 3 Number of quarters : 12 Number of dimes : 14 Number of nickels : 50 Number of pennies : 175	ABP coin credit Dollars : 11 Change : 65 cents

Exercise 3: **Water Bill Program**
Write a program that computes a customer's water bill. The bill includes a $35 water demand charge plus a consumption (use) charge of $1.10 for every thousand gallons used. Consumption is figured from meter readings (in thousands of gallons) taken recently and at the end of the previous quarter. If the customer's unpaid balance is greater than zero, a $2 late charge is also assessed.
Problem Constants
DEMAND_CHG 35.00 /* basic water demand charge */
PER_1000_CHG 1.10 /* charge per thousand gallons used*/
LATE_CHG 2.00 /* surcharge on an unpaid balance */

Problem Inputs
int previous /* meter reading from the previous quarter in thousands of gallons */
int current /* meter reading from current quarter */

double unpaid /* unpaid balance of previous bill */

Problem Outputs
double bill /* water bill */
doubleuse_charge /* charge for actual water use */
doublelate_charge /* charge for nonpayment of part of previous balance */

Relevant Formulas
water bill = demand charge + use charge + unpaid balance+ applicable late charge
Try the sample test cases given below:

Sample Test Cases	Input	Output
Test Case 1	This program figures a water bill based on the demand charge ($35.00) and a $1.10 per 1000 gallons use charge. A $2.00 surcharge is added to accounts with an unpaid balance. Enter unpaid balance and previous and current meter readings on separate lines after the prompts. Press or after typing each number. Enter unpaid balance> $71.50 Enter previous meter reading> 4198 Enter current meter reading> 4238	Bill includes $2.00 late charge on the unpaid balance of $71.50 Total due = $152.50
Test Case 2	This program figures a water bill based on the demand charge ($35.00) and a $1.10 per 1000 gallons use charge. A $2.00 surcharge is added to accounts with an unpaid balance. Enter unpaid balance and previous and current meter readings on separate lines after the prompts. Press or after typing each number. Enter unpaid balance> $51 Enter the previous meter reading> 4198 Enter current meter reading> 4137	Bill includes $2.00 late charge on the unpaid balance of $71.50 Total due = $102.00

Exercise 4: **Prime Sum**
Given a positive integer, n, calculate the sum of all prime numbers between 1 and n (inclusive).

Sample Test Cases	Input	Output
Test Case 1	5	10
Test Case 2	10	17

Exercise 5: **Bubble Sort**
Write a program that reads n integers from the user at the keyboard and stores them in an array. The program should then sort the array in ascending order using the bubble sort algorithm and print out the sorted list.

Sample Test Cases	Input	Output
Test Case 1	Enter the number of elements: 5 Enter elements in the array: 23 46 5 18 39	The sorted list is 5 18 23 39 46
Test Case 2	Enter the number of elements: -5	Invalid input. Please enter a positive number.

Exercise 6: **Text Editor**

Write a program to perform editing operations on a line of text. Your editor should be able to locate a target substring, delete a substring, and insert a substring at a specified location. The editor should expect source strings of less than 80 characters.

Sample Test Cases	Input	Output
Test Case 1	Enter the source string: Internet use is growing rapidly. Enter D(Delete), I(Insert), F(Find), or Q(Quit): d String to delete: growing	Internet use is rapidly.
Test Case 2	Enter D(Delete), I(Insert), F(Find), or Q(Quit): F String to find> u	'u' is found at position 6.

Exercise 7: Arithmetic with Common Fractions

Write a program to add, subtract, multiply, and divide several pairs of common fractions.

Sample Test Cases	Input	Output
Test Case 1	Enter a common fraction as two integers separated by a slash: 3/4 Enter a common fraction as two integers separated by a slash: 5/8 Enter an arithmetic operator (+,-,*, or /) : +	3/4 + 5/8 = 11/8
Test Case 2	Input invalid— denominator must be positive	Input invalid— denominator must be positive.

Exercise 8: Factorial using Recursion

Write a C program to calculate the factorial of a number using recursion.

Recursive method for calculating factorial.

$$\text{factorial}(n) = \begin{cases} 1 & \text{if } n = 0 \\ n * \text{factorial}(n-1) & \text{if } n > 0 \end{cases}$$

Sample Test Cases	Input	Output
Test Case 1	Enter the number: 5	Factorial: 120
Test Case 2	Enter the number : 3	Factorial: 6

Exercise 9: **Collecting Area for Solar Heated House**

PROBLEM STATEMENT : An architect needs a program that can estimate the appropriate size for the collecting area of a solar-heated house. Determining collecting area size requires consideration of several factors, including the average number of heating degree days for the coldest month of a year (the product of the average difference between inside and outside temperatures and the number of days in the month), the heating requirement per square foot of floor space, the floor space, and the efficiency of the collection method. The program will have access to two data files. File hdd.txt contains numbers representing the average heating degree days in the construction location for each of 12 months. File solar.txt contains the average solar insolation(rate in BTU/day at which solar radiation falls on one square foot of a given location)for each month. The first entry in each file represents data for January, the second, data for February, and so on.

Problem Inputs
Average heating degree days file Average
solar insolation file
heat_deg_days /* average heating degree days for coldest month */
coldest_mon /* coldest month (number 1 .. 12) */
solar_insol /* average daily solar insolation (BTU/ft^2)for coldest month */
heating_req /* BTU/degree day ft^2 for planned type construction*/
efficiency /* % of solar insolation converted to usable heat */
floor_space /* square feet */
Program Variables
energy_resrc /* usable solar energy available in coldest month (BTUs obtained from 1 ft^2 of collecting area) */
Problem Outputs
heat_loss /* BTUs of heat lost by structure in coldest month */
collect_area /* approximate size (ft^2) of collecting area needed*/
The formula for approximating the desired collecting area (A) is:
A= heat loss / energy resource
Design algorithm , flow chart ,program using the above data requirements for the given problem Try

Sample Test Cases	Input	Output
Test Case 1	What is the approximate heating requirement (BTU / degree day ft^2) of this type of construction?	To replace heat loss of 11350800 BTU in the coldest month (month 12) with available solar insolation

	=>9 What percent of solar insolation will be converted to usable heat? => 60 What is the floor space (ft^2)? => 1200	of 500 BTU / ft^2 / day, and an efficiency of 60 percent, use a solar collecting area of 1221 ft^2.
Test Case 2	What is the approximate heating requirement (BTU / degree day ft^2) of this type of construction? =>10 What percent of solar insolation will be converted to usable heat? => 60 What is the floor space (ft^2)? => 1200	To replace heat loss of 12612000 BTU in the coldest month (month 12) with available solar insolation of 500 BTU / ft^2 / day, and an efficiency of 60 percent, use a solar collecting area of 1221 ft^2.

Exercise 10: **Universal Measurement Conversion**

Design a program that takes a measurement in one unit (e.g., 4.5 quarts) and converts it to another unit (e.g., liters). For example, this conversion request 450 km miles would result in this program output Attempting conversion of 450.0000 km to miles 450.0000km = 279.6247 miles . The program should produce an error message if a conversion between two units of different classes (e.g., liquid volume to distance) is requested. The program should take a database of conversion information from an input file before accepting conversion problems entered interactively by the user. The user should be able to specify units either by name (e.g., kilograms) or by the abbreviation (e.g., kg).

Structured Data Type

unit_tmembers :
name /* character string such as "milligrams" */
abbrev /* shorter character string such as "mg" */
class /* character string "liquid_volume", "distance", or "mass" */
standard /* number of standard units that are equivalent to this unit */

Problem Constants

NAME_LEN 30 /* storage allocated for a unit name */
ABBREV_LEN 15 /*
storage allocated for a unit abbreviation */ CLASS_LEN 20 /*
storage allocated for a measurement class */
MAX_UNITS 20 /* maximum number of different units handled */

Problem Inputs

unit_t units[MAX_UNITS] /* array representing unit conversionfactors database */
double quantity /* value to convert */
charold_units[NAME_LEN] /* name or abbreviation of units tobe converted */
charnew_units[NAME_LEN] /* name or abbreviation of units to convert to */

Problem Output

Message giving conversion.
Data file units.txt:

miles mi distance 1609.3
kilometers km distance 1000
ards yd distance 0.9144
meters m distance 1
quartsqtliquid_volume 0.94635
liters l liquid_volume 1
gallons gal liquid_volume 3.7854
milliliters ml liquid_volume 0.001
kilograms kg mass 1
grams g mass 0.001
slugsslugs mass 0.14594
poundslb mass 0.43592
Design algorithm , flow chart ,program using the above data requirements for the given problem
Try the sample test cases given below :

Sample Test Cases	**Input**	**Output**
Test Case 1	To convert 25 kilometers to miles, you would enter > 25 kilometers miles or> 25 km mi	450 km miles Attempting conversion of 450.0000 km to miles . . . 450.0000km = 279.6247 miles
Test Case 2	Enter a conversion problem or q to quit. > 2.5 qt l Attempting conversion of 2.5000 qt to l . . . 2.5000qt = 2.3659 l Enter a conversion problem or q to quit.	> 100 meters gallons Attempting conversion of 100.0000 meters to gallons . . . Cannot convert meters (distance) to gallons (liquid_volume)

***Textbook :** B. A. Forouzan and R. F. Gilberg —Cengage Learning , Computer Science: A Structured Programming Approach Using Cǁ Third Edition.